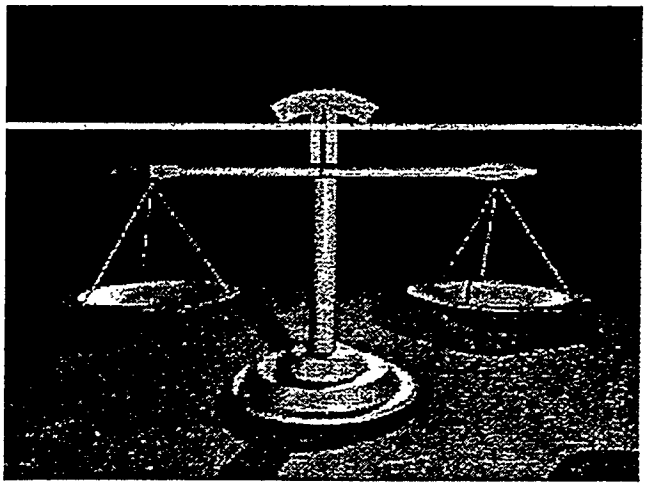
NUREG/CR-0200, Rev. 5
ORNL/NUREG/CSD-2/R5
Vol. 1, Part 1

# SCALE

## A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation



## Control Modules, Part 1

Oak Ridge National Laboratory

Prepared for
U.S. Nuclear Regulatory Commission

RECEIVED
MAY 07 1997
OSTI

RECE
MAY 07 199
OSTI

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

# DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, make any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

NUREG/CR–0200, Rev. 5
ORNL/NUREG/CSD-2/R5
Vol. 1, Part 1

# SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation

## Control Modules
## C4, C6

Manuscript Completed: December 1996
Date Published: March 1997

Oak Ridge National Laboratory
Managed by Lockheed Martin Energy Research Corp.

Oak Ridge National Laboratory
Oak Ridge, TN 37831

**Prepared for**
**Spent Fuel Project Office**
**Office of Nuclear Material Safety and Safeguards**
**U.S. Nuclear Regulatory Commission**
**Washington, DC 20555–0001**
**NRC Job Code B0009**

## AVAILABILITY NOTICE

Availability of Reference Materials Cited in NRC Publications

Most documents cited in NRC publications will be available from one of the following sources:

1. The NRC Public Document Room, 2120 L Street, NW., Lower Level, Washington, DC 20555-0001

2. The Superintendent of Documents, U.S. Government Printing Office, P. O. Box 37082, Washington, DC 20402-9328

3. The National Technical Information Service, Springfield, VA 22161-0002

Although the listing that follows represents the majority of documents cited in NRC publications, it is not intended to be exhaustive.

Referenced documents available for inspection and copying for a fee from the NRC Public Document Room include NRC correspondence and internal NRC memoranda; NRC bulletins, circulars, information notices, inspection and investigation notices; licensee event reports; vendor reports and correspondence; Commission papers; and applicant and licensee documents and correspondence.

The following documents in the NUREG series are available for purchase from the Government Printing Office: formal NRC staff and contractor reports, NRC-sponsored conference proceedings, international agreement reports, grantee reports, and NRC booklets and brochures. Also available are regulatory guides, NRC regulations in the *Code of Federal Regulations*, and *Nuclear Regulatory Commission Issuances*.

Documents available from the National Technical Information Service include NUREG-series reports and technical reports prepared by other Federal agencies and reports prepared by the Atomic Energy Commission, forerunner agency to the Nuclear Regulatory Commission.

Documents available from public and special technical libraries include all open literature items, such as books, journal articles, and transactions. *Federal Register* notices, Federal and State legislation, and congressional reports can usually be obtained from these libraries.

Documents such as theses, dissertations, foreign reports and translations, and non-NRC conference proceedings are available for purchase from the organization sponsoring the publication cited.

Single copies of NRC draft reports are available free, to the extent of supply, upon written request to the Office of Administration, Distribution and Mail Services Section, U.S. Nuclear Regulatory Commission, Washington, DC 20555-0001.

Copies of industry codes and standards used in a substantive manner in the NRC regulatory process are maintained at the NRC Library, Two White Flint North, 11545 Rockville Pike, Rockville, MD 20852-2738, for use by the public. Codes and standards are usually copyrighted and may be purchased from the originating organization or, if they are American National Standards, from the American National Standards Institute, 1430 Broadway, New York, NY 10018-3308.

## DISCLAIMER NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use, or the results of such use, of any information, apparatus, product, or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

## CONTENTS

PREFACE . . . . . vii

**Volume 1, Part 1: Control Modules**

C1* CSAS1: A ONE-DIMENSIONAL CRITICALITY SAFETY ANALYSIS MODULE

C2* CSAS2: A MULTIDIMENSIONAL CRITICALITY SAFETY ANALYSIS MODULE

C3*** CSAS3: AN OPTIMUM CONCENTRATION CRITICALITY SAFETY ANALYSIS MODULE

C4 CSAS: CONTROL MODULE FOR ENHANCED CRITICALITY SAFETY ANALYSIS SEQUENCES (N. F. Landers, L. M. Petrie)

C6 CSAS6: CONTROL MODULE FOR ENHANCED CRITICALITY SAFETY ANALYSIS WITH KENO VI (D. F. Hollenbach, L. M. Petrie)

**Volume 1, Part 2: Control Modules**

S1 SAS1: A ONE-DIMENSIONAL SHIELDING ANALYSIS MODULE (J. R. Knight, C. V. Parks, S. M. Bowman, L. M. Petrie, J. A. Bucholz)

S2 SAS2H: A COUPLED ONE-DIMENSIONAL DEPLETION AND SHIELDING ANALYSIS MODULE (O. W. Hermann, C. V. Parks)

S3 SAS3: AN AUTOMATED MONTE CARLO SHIELDING ANALYSIS MODULE (M. B. Emmett, J. T. West)

S4 SAS4: A MONTE CARLO CASK SHIELDING ANALYSIS MODULE USING AN AUTOMATED BIASING PROCEDURE (J. S. Tang)

S5 QADS: A MULTIDIMENSIONAL POINT-KERNEL ANALYSIS MODULE (B. L. Broadhead)

H1 HTAS1: A TWO-DIMENSIONAL HEAT TRANSFER ANALYSIS OF FUEL CASKS, VERSION 4.0 (G. E. Giles)

*Obsolete with SCALE-4.0 release.
**Not included in SCALE-4.3 release.
***Cancelled. Included in Section C4.

**Volume 2, Part 1: Functional Modules**

F1 BONAMI: RESONANCE SELF-SHIELDING BY THE BONDARENKO METHOD (N. M. Greene)

F2 NITAWL-II: SCALE SYSTEM MODULE FOR PERFORMING RESONANCE SHIELDING AND WORKING LIBRARY PRODUCTION (N. M. Greene, L. M. Petrie, R. M. Westfall)

F3 XSDRNPM: A ONE-DIMENSIONAL DISCRETE-ORDINATES CODE FOR TRANSPORT ANALYSIS (N. M. Greene, L. M. Petrie)

F4 XSDOSE: A MODULE FOR CALCULATING FLUXES AND DOSE RATES AT POINTS OUTSIDE A SHIELD (J. A. Bucholz)

F5* KENO IV/S: AN IMPROVED MONTE CARLO CRITICALITY PROGRAM

F6 COUPLE: SCALE SYSTEM MODULE TO PROCESS PROBLEM-DEPENDENT CROSS SECTIONS AND NEUTRON SPECTRAL DATA FOR ORIGEN-S DATA ANALYSES (O. W. Hermann)

F7 ORIGEN-S: SCALE SYSTEM MODULE TO CALCULATE FUEL DEPLETION, ACTINIDE TRANSMUTATION, FISSION PRODUCT BUILDUP AND DECAY, AND ASSOCIATED RADIATION SOURCE TERMS (O. W. Hermann, R. M. Westfall)

F8 ICE: MODULE TO MIX MULTIGROUP CROSS SECTIONS (N. M. Greene, L. M. Petrie, S. K. Fraley)

**Volume 2, Part 2: Functional Modules**

F9 MORSE-SGC FOR THE SCALE SYSTEM (J. T. West, T. J. Hoffman, M. B. Emmett)

F10 HEATING 7.2 USER'S MANUAL (K. W. Childs)

F11 KENO V.a: AN IMPROVED MONTE CARLO CRITICALITY PROGRAM WITH SUPERGROUPING (L. M. Petrie, N. F. Landers)

F12* JUNEBUG-II: A THREE-DIMENSIONAL GEOMETRY PLOTTING CODE

F13* HEATPLOT-S: A TEMPERATURE DISTRIBUTION PLOTTING PROGRAM FOR HEATING

---

*Obsolete with SCALE-4.0 release.

**Not included in SCALE-4.3 release.

**Volume 2, Part 3: Functional Modules**

F14* REGPLOT6: A PLOTTING PROGRAM TO VERIFY HEATING INPUT DATA

F15** PLORIGEN: A PLOTTING PROGRAM FOR ORIGEN-S OUTPUT (O. W. Hermann)

F16 OCULAR: A RADIATION EXCHANGE FACTOR COMPUTER PROGRAM (C. B. Bryan, G. E. Giles)

F17 KENO-VI: A GENERAL QUADRATIC VERSION OF THE KENO PROGRAM (D. F. Hollenbach, L. M. Petrie, N. F. Landers)

**Volume 3: Miscellaneous**

M1 SCALE SYSTEM DRIVER (L. M. Petrie)

M2 SCALE SUBROUTINE LIBRARY (L. M. Petrie)

M3 SCALE FREE-FORM READING ROUTINES (L. M. Petrie)

M4 SCALE CROSS-SECTION LIBRARIES (W. C. Jordan)

M5 THERMAL MATERIAL PROPERTIES LIBRARY (A. L. Edwards, P. T. Williams)

M6 ORIGEN-S DATA LIBRARIES (J. C. Ryman, O. W. Hermann)

M7 THE MATERIAL INFORMATION PROCESSOR FOR SCALE (N. F. Landers, L. M. Petrie, J. A. Bucholz)

M8 STANDARD COMPOSITION LIBRARY (J. A. Bucholz, J. R. Knight, C. V. Parks, L. M. Petrie, J. C. Turner, R. M. Westfall)

M9 MARS: A MULTIPLE ARRAY SYSTEM USING COMBINATORIAL GEOMETRY (J. T. West, M. B. Emmett)

M10 FIDO INPUT SYSTEM (L. M. Petrie)

M11* SCALE INTERACTIVE INPUT PROCESSOR

M12* CESAR: A CRITICALITY EXPERIMENT STORAGE AND RETRIEVAL PROGRAM

*Obsolete with SCALE-4.0 release.
**Not included in SCALE-4.3 release.

**Volume 3: Miscellaneous (continued)**

M13 PICTURE: A PRINTER PLOT PACKAGE FOR MAKING 2-D PICTURES OF MARS GEOMETRIES (M. B. Emmett)

M14 COMPOZ DATA GUIDE (J. R. Knight, L. M. Petrie)

M15 USER'S GUIDE FOR UTILITY MODULES (N. M. Greene)

M16 COMMENT DATA GUIDE (L. M. Petrie)

*Obsolete with SCALE-4.0 release.

# PREFACE

## Background

This Manual represents Revision 5 of the user documentation for the modular code system referred to as SCALE. The history of the SCALE code system dates back to 1969 when the current Computational Physics and Engineering Division at Oak Ridge National Laboratory (ORNL) began providing the transportation package certification staff at the U.S. Atomic Energy Commission with computational support in the use of the new KENO code for performing criticality safety assessments with the statistical Monte Carlo method. From 1969 to 1976 the certification staff relied on the ORNL staff to assist them in the correct use of codes and data for criticality, shielding, and heat transfer analyses of transportation packages. However, the certification staff learned that, with only occasional use of the codes, it was difficult to become proficient in performing the calculations often needed for an independent safety review. Thus, shortly after the move of the certification staff to the U.S. Nuclear Regulatory Commission (NRC), the NRC staff proposed the development of an easy-to-use analysis system that provided the technical capabilities of the individual modules with which they were familiar. With this proposal, the concept of the Standardized Computer Analyses for Licensing Evaluation (SCALE) code system was born.

The NRC staff provided ORNL with some general development criteria for SCALE: (1) focus on applications related to nuclear fuel facilities and package designs, (2) use well-established computer codes and data libraries, (3) design an input format for the occasional or novice user, (4) prepare "standard" analysis sequences (control modules) that will automate the use of multiple codes (functional modules) and data to perform a system analysis, and (5) provide complete documentation and public availability. With these criteria the ORNL staff laid out the framework for the SCALE system and began development efforts. The initial version (Version 0) of the SCALE Manual was published in July 1980. Then, as now, the Manual is divided into three volumes – Volume 1 for the control module documentation (Sections C4, C6, S1– S4, and H1), Volume 2 for the functional module documentation (Sections F1– F17), and Volume 3 for the documentation of the data libraries and subroutine libraries (Sections M1–M16).

## System Overview

The original concept of SCALE was to provide "standardized" sequences where the user had very few analysis options in addition to the geometry model and materials. Input for the control modules has been designed to be free-form with extensive use of keywords and engineering-type input requirements. The more flexible functional modules have a more difficult input logic and require the user to interface the data sets necessary to run the modules in a stand-alone fashion. As the system has grown in popularity over the years and additional options have been requested, the control modules have been improved to allow sophisticated users additional access to the numerous capabilities within the functional modules. However, the most important feature of the SCALE system remains the capability to simplify the user knowledge and effort required to prepare material mixtures and to perform adequate problem-dependent cross-section processing.

The modules available in Version 0 of SCALE were for criticality safety analysis sequences (CSAS) that provided automated material and cross-section processing prior to a one-dimensional (1-D) or multidimensional criticality analysis. Since that time the capabilities of the system have been

significantly expanded to provide additional CSAS capabilities, new shielding analysis sequences (SAS) that also include depletion/decay capabilities for spent fuel characterization, and a heat transfer analysis sequence (HTAS). At the center of the CSAS and SAS sequences is the library of subroutines referred to as the Material Information Processor or MIPLIB (see Section M7). The purpose of MIPLIB is to allow users to specify problem materials using easily remembered and easily recognizable keywords that are associated with mixtures, elements, and nuclides provided in the Standard Composition Library (see Section M8). MIPLIB also uses other keywords and simple geometry input specifications to prepare input for the modules that perform the problem-dependent cross-section processing: BONAMI, NITAWL-II, and XSDRNPM. A keyword supplied by the user selects the cross-section library from a standard set provided in SCALE (see Section M4) or designates the reference to a user-supplied library. Several utility modules from AMPX[1] have been included to provide users with the capability to edit the cross-section data and reformat user-supplied libraries for use in SCALE.

Over the history of the project several modules have been removed from the system because they are no longer supported by the development staff at ORNL. Tables 1 and 2 provide a summary of the major applications of each of the control modules and functional modules currently in the SCALE code system. The control modules were designed to provide the system analysis capability originally requested by the NRC staff. The CSAS module (sometimes denoted as the CSAS4 module and documented in Section C4) is the primary control module designed for the calculation of the neutron multiplication factor of a system. Eight sequences enable general analysis of a 1-D system model or a multidimensional system model, capabilities to search on geometry spacing, and problem-dependent cross-section processing for use in executing stand-alone functional modules. CSAS6 is a new criticality control module to provide automated problem-dependent cross-section processing and criticality calculations via the new KENO-VI functional module. The SAS1 and SAS3 modules (see Sections S1 and S3, respectively) provide general 1-D deterministic and 3-D Monte Carlo analysis capabilities. The SAS2 module (see Section S2) was originally developed to perform a depletion/decay calculation to obtain spent fuel radiation source terms that were subsequently input automatically to a 1-D, radial shielding analysis in a cylindrical geometry. Over time the depletion/decay portion of the SAS2 module has been significantly enhanced and interfacing to the other shielding modules has been provided. The SAS4 module (see Section S4) enables automated particle biasing for a Monte Carlo analysis of a transportation package-type geometry. The HTAS1 module (see Section H1) is the only heat transfer control module and uses the various capabilities of the HEATING code to perform different sequences of steady-state and transient analysis that enable the normal and accident conditions of a transportation package to be evaluated. Like SAS4, the HTAS1 module is limited to a package-type geometry.

The capability to perform a point-kernel shielding analysis within the SCALE system has been developed in the QADS control module which has been added in this release of the software package.

A 238-energy-group neutron cross-section library based on ENDF/B-V has recently been prepared for the SCALE system.[5] All the nuclides that are available in ENDF/B-V are in the library. A 44-group library has been collapsed from this 238-group library and validated against numerous critical measurements.[6] These libraries are available in this version of SCALE or as a separate data package from the software distribution centers.

## Major Enhancements in SCALE-4.3

### New ENDF/B-V libraries

SCALE-4.3 includes 238GROUPNDF5, a 238-group, AMPX master-format neutron cross-section library that contains data for all the nuclides available in ENDF/B-V. The library has Bondarenko shielding factors included for nuclides with unresolved resonances, nuclides with Adler-Adler resonance data (U-233 and Pu-241), and the nonresonance nuclides Li-7, F-19, Al-27, and Si-14. Bondarenko factors are used for the latter nuclides because they exhibit resonance structure in the point data. The initial development of this library included only s-wave resonance data in the resonance parameters that are passed to the Nordheim treatment in NITAWL. Although demonstrated sufficient for thermal systems, it was found that p-wave and perhaps d-wave resonances can have a significant effect on the results for shielding applications and for unmoderated, intermediate energy problems. Thus, p-wave and d-wave data were added to the library. Users, however, will not be able to readily access the p- and d-wave resonance data until SCALE-4.3 has been made available to RSIC in 1995.

Several spectra suitable for collapsing this fine-group library to a broad-group structure are included as part of the library. A 44-group derivative of the 238-group library, 44GROUPNDF5, collapsed using a spectrum for a pressurized-water-reactor fuel pin lattice, is also included in the package. This broad-group library has a similar group structure as that provided in the SCALE 27-group ENDF/B-IV library, except that additional groups were added to ensure more accurate treatment of the 0.3-eV resonance of $^{239}$Pu. Based on an analysis of a set of 93 thermal and fast critical experiments, the ENDF/B-V 44-group library has demonstrated markedly improved performance over similar analyses performed with the ENDF/B-IV 27-group library ("Validation of SCALE-4 Criticality Sequences Using ENDF/B-V Data," *Proceedings of the Topical Meeting on Physics and Methods in Criticality Safety*, September 19–23, 1993). In addition, by comparing SAS2H/ORIGEN-S results with measured isotopic data, the library has demonstrated improved results in the prediction of spent fuel isotopics (*Validation of the SCALE System for PWR Spent Fuel Isotopic Composition Analyses*, ORNL/TM-12667, March 1995).

### KENO-VI

KENO-VI is a new version of the KENO Monte Carlo Criticality Safety Code developed at Oak Ridge National Laboratory. It constructs and processes geometry data as sets of quadratic equations. A lengthy set of simple, easy-to-use geometric functions, similar to those provided in KENO-V.a, and the ability to build more complex geometric shapes (represented by sets of quadratic equations) are the heart of the geometry package in KENO-VI. The code's flexibility is increased by allowing the following features: intersecting geometry regions; hexagonal as well as cuboidal arrays; regions, holes, arrays, and units rotated to any angle and truncated to any position; and the use of an array boundary that intersects the array.

KENO-VI maintains all the flexibility and options of KENO-V.a plus a variety of new options. In KENO-VI, units can be constructed using both the simple geometric shapes provided and the tailored geometric shapes constructed using quadratic equations. It includes the new 2-D color plotting capability that has been added to KENO-V.a (see "Improvements to KENO-V.a Include 2-D

Color Plot Capability"). A new SCALE Criticality Safety Analysis Sequence, CSAS6, has been added for running KENO-VI.

Users should be aware that the added geometry features in KENO-VI can result in significantly longer run times than KENO-V.a. A KENO-VI problem that can also be modeled with KENO-V.a will typically run twice as long as the same problem using KENO-V.a. Thus, the new version VI is not a replacement for the existing version V.a, but an additional version for complex geometries that could not be modeled previously.

**New criticality search on material concentrations**

A new search feature has been added to the criticality search sequences in SCALE. Searches may now be done on the concentration of standard compositions in one or more mixtures in the problem. During a concentration search, if the search convergence criteria have not been met, the atom densities of the specified standard compositions are updated, problem-dependent resonance processing is performed using BONAMI, NITAWL, and applicable sections of the material information processor, and a new $k_{eff}$ is calculated using KENO-V.a. Concentration searches cannot be performed on materials contained in cell-weighted mixtures.

In all search sequences (both geometry and concentration), problem-dependent resonance processing of the cross sections is now performed for each iteration prior to the execution of KENO. Heterogeneous effects, Dancoff factors, and self-shielding are now accounted for between iterations. Prior to the addition of concentration searches, a search sequence would update only the specified geometry, but not the cross sections, prior to calculating a new $k_{eff}$ using KENO-V.a. Previously, to ensure that the converged result was acceptable, the problem needed to be rerun with the converged geometry data. The recent changes to the search sequences eliminate the need to rerun a converged problem to account for changes in cross sections.

**Enhanced graphical capabilities**

The graphical display of KENO-V.a and KENO-VI geometry models has been dramatically improved by the new capability to produce high-resolution 2-D color plot files that can be viewed on-screen using appropriate viewer software or printed on a color printer. This new option is activated by using the parameter scr=yes in the KENO plot data. A plot file using up to 256 colors is generated. The plot data parameter "clr=" can be used to override the default color assignments. These plots can be generated only on UNIX workstations and MS-DOS-based personal computers (PC).

In addition to the 2-D color plots, fractional values are now valid for the lpi= parameter in the plot data, which will make it easier to print circles on laser printers.

**KENO-V.a enhancements**

Enhancements to KENO-V.a fall into one of three categories: removed limitations, editing changes, or added capabilities.

Two independent limitations to KENO-V.a were removed. The first limitation involved HOLE boundaries touching other HOLE or adjacent geometry region boundaries. Significant errors

in $k_{eff}$ (as much as 1%) occurred when HOLE boundaries touched with no indication of a problem. HOLE boundaries may now touch other HOLE or adjacent geometry region boundaries.

The second limitation involved the number of scattering angles allowed. Although the parameter sct= (number of scattering angles) was limited to 13, no check was made to avoid overstoring data. Any number of scattering angles are now allowed.

The mixing table edit has been changed to sort mixture data by mixture number. The mixture number and density make up the first line of the data for each mixture. The following lines, which represent the nuclides in the mixture, are sorted by nuclide number. Each line contains a nuclide number, atom density, weight fraction, ZA number, atomic weight, and nuclide title. This provides the user a concise table of the components that make up the mixtures used in the problem.

Three new capabilities were also added to KENO-V.a which provide the user with useful information for analyzing and checking the problem.

The energy corresponding to the average lethargy causing fission is now calculated and the printing of this information (i.e., NUB=YES) is now a default parameter. This is useful for determining trends in $k_{eff}$ as a function of neutron energy to look for problem-type biases.

System total fissions and absorptions by region are now calculated. A parameter to suppress printing fissions and absorptions by region for each group has been added. Set GAS=NO to print only the system totals by group. These values have several uses, such as helping to verify proper problem sampling and determining region importance.

Finally, the total mass of each mixture in the problem is now calculated and printed. These values can be used to help verify the geometry and material masses of the problem.

**New cross-section processing features**

The cross-section processing portion of the SCALE system has been upgraded to add major improvements to the NITAWL-II code. NITAWL-II has been modified to perform the resonance calculation in double precision. The mesh generation for a resonance has been modified to better span higher energy resonances and to reduce excessive numbers of mesh points.

In addition, this new version of NITAWL-II is able to interpolate the available thermal-scattering matrices to obtain data for the temperature requested by user input (via direct input to NITAWL-II or via the SCALE Standard Composition data). Previously, NITAWL-II had no such capability and selected the thermal-scattering data that was closest to the requested temperature. Thermal-scattering data for moderator materials, such as hydrogen and graphite, are available at many more temperatures in the new ENDF/B-V SCALE libraries than in the older ENDF/B-IV SCALE libraries. However, even with the new ENDF/B-V libraries, the interpolation should better determine the effect that a change in moderator temperature will have on the system multiplication factor.

The CSAS and SAS1–SAS4 modules also have been modified via MIPLIB to provide two new parameters in the optional MORE DATA input:

MLV= maximum L value for which resonance calculation will be done (L = 0 is s-wave, L = 1 includes p-wave, L = 2 includes d-wave). The default value is L = 2.

MSH= maximum number of mesh points per resonance. The default value is 2001.

Note that if NITAWL-II is run stand-alone, the default values for MLV and MSH are 0 and 501, respectively.

## New KENO-V.a sample problems

The KENO-V.a sample problems have been updated to improve the range of functionality testing and provide more flexibility in using other cross-section libraries. The SCALE-4.2 sample problems used the 27-group ENDF/B-IV cross-section library. The cross sections were processed by NITAWL-II prior to the execution of KENO-V.a. The new sample problems use BONAMI, which is not necessary for the 27-group library, as well as NITAWL-II for resonance processing of cross sections so that the sample problems can easily be run with other cross-section libraries that require the use of BONAMI. The NITAWL-II input data were updated to specify 2001 mesh points per resonance and the highest resonance L-value of 2. The previous input data used the default values of 501 mesh points and an L-value of 0.

Eight new problems were added to the existing 25 sample problems. Sample problems 26 through 28 are variations of sample problem 1, modeled as hemicylinders with chords and origins, and are extensions of sample problems 23 through 25, which are modeled using chords without origins. Sample problem 29 is a bare critical sphere, and sample problems 30 through 32 are the same critical system modeled using hemispheres with origins. Sample problem 33 is a critical triangular pitched array of partially flooded cylindrical annuli of low-enriched-uranium metal.

## Standard Composition Library updated for ENDF/B-VI compatibility

The Standard Composition Library has been expanded for compatibility with ENDF/B-VI cross-section libraries. In addition, the Material Information Processor Library (MIPLIB) has also been updated so that all the control modules can use ENDF/B-VI cross-section libraries via the new expanded Standard Composition Library. The changes were necessary because the cross sections for the natural-element identifiers such as **FE** for iron are no longer available in the ENDF/B-VI data. Most problems set up to use other cross-section libraries should now run without any changes using ENDF/B-VI libraries.

The changes to the Standard Composition Library were very extensive and fall into several categories. Additions were made so that nearly all the elements are given by both their full name and by their chemical symbol. The full-name elements are all given as naturally occurring abundance nuclides, while a few of the chemical symbol nuclides are given as the principal isotope. For instance, **HYDROGEN** would include trace amounts of deuterium, while **H** would be 100% $^{1}H$.

The isotopic distribution table was changed to include most of the elements. Arbitrary materials may now contain more than one multiple-isotope nuclide. In addition, the specially weighted elements in stainless steel and in Inconel were added, specifying the natural abundance of each.

Atomic masses for isotopes were taken from ENDF/B-VI, if available. Atomic masses for naturally occurring elements were taken from ENDF/B-V, if available. Remaining atomic masses were taken from *GE Chart of the Nuclides* or the *CRC Handbook of Chemistry and Physics.*

## QAD-CGGP and QADS

The 3-D point-kernel shielding code, QAD-CGGP, and its associated SCALE control module have been added. QADS performs multidimensional point-kernel estimation of gamma transport through practically any type of shielding materials using a simplified input scheme that follows the

general input philosophy of the SCALE shielding sequences. QADS was designed to capture the flexibility and power of the QAD technique for problems amenable to point-kernel solution while allowing for an easy-to-use input interface. The QADS module is called via the SCALE driver and preprocesses the actual input for the QAD-CGGP code. The driver then calls and executes QAD-CGGP automatically. The SCALE standard composition library is used to allow simplified input of materials. The problem geometry is coded using the well-known MORSE combinatorial geometry package. Finally, the dose portion of the input follows the XSDOSE input data format in SAS1. The combined use of the SCALE standard compositions and free-form input with the multidimensional geometry capabilities and generally short running time of point-kernel techniques produces a very powerful procedure for shielding analysis of a wide variety of problems.

**SCALE-4.3 minor modifications**

In addition to the major enhancements noted above, SCALE-4.3 contains many minor modifications, including corrections to errors in SCALE-4.2 and changes to improve portability to other computing platforms.

**SCALE driver**: The driver was modified to improve portability to other types of workstations. Modifications to accept upper or lower case input files were also added.

**KENO-V.a:** (1) Corrected error that sometimes occurred when the boundary of a HOLE touched another HOLE or adjacent geometry region. The error can produce erroneous results, either conservative or nonconservative, without triggering warning or error messages. The effect on $k_{eff}$ can be very significant in some cases (~ 1%). (2) Fixed to allow more than 13 scattering angles. Although the parameter "sct=" was limited to 13, no check was made to avoid overstoring data. (3) Corrected error causing core dump on some workstations when a global unit wasn't explicitly defined. (4) Corrected error that permitted a case with a negative radius to run. (5) Added calculation of the energy corresponding to the average lethargy causing fission and made the printing of this information (i.e., NUB=YES) a default parameter. (6) Corrected errors in the starting points edit obtained when PSP=YES. (7) Corrected an error that occurred when a history encountered an edge where differential albedos were used. Other minor fixes were made for (8) printing densities of mixtures that have other mixtures added in and (9) printing plot coordinates for problems ending in a bare array. (10) Output was enhanced to print information on start and plot input parameters that were previously omitted.

**NITAWL**: (1) Corrected the grayness correction to self-shielding in annular regions. (2) Corrected an error passing a variable of the wrong type that resulted in incorrect self-shielding of thermal groups below 0.5 eV.

**All Control Modules (MIPLIB)**: (1) Added checks to avoid the use of negative number densities when solutions with physically unrealistic parameters are entered. (2) Corrected an error that caused CSAS to loop if the collapse option was specified and an input nuclide ID was not found in the cross-section library. (3) Modified MIPLIB to check for resonance data on the cross-section library used to determine whether to perform resonance corrections for a nuclide. Previously, a flag in the Standard Composition Library was used to make this determination. (4) Corrected calculated thickness ("lbar") of cylindrical and spherical annuli for input to BONAMI. (5) Modified reading of Dancoff factor keywords in MORE DATA to detect and prevent erroneous keyword input. (6) Changed the label "lbar" printed by CSAS in PARM=CHECK output to "abar." (7) Replaced the 123GROUP and 218GROUP cross-section-library keywords with 44GROUP and

238GROUP. The former libraries, which were the least used of the standard SCALE libraries, were removed as defaults so the new ENDF/B-V libraries, which give more accurate results, could be accessed directly via keyword input.

**CSAS/MODIFY**: Added an option in the CSAS4 and CSAS4X sequences to perform a criticality search on nuclide concentration. Modified the search sequences to update unit cell data or number densities (depending on the type of search) and execute BONAMI and NITAWL prior to every KENO-V.a calculation.

**SAS2:** (1) Modified for compatibility with recent MIPLIB changes. (2) Corrected the soluble boron concentration to change linearly when NLIB/CYC>1. (3) Modified to allow use of the SYMMSLABCELL geometry in MIPLIB and more than one fuel zone. (4) Automated model setup when burnable poison rods occupy some but not all guide tubes in a fuel assembly. (5) Added THERM, RES, FAST, and the two energy boundaries for converting cross sections to the total flux normalization. (6) Modified the AMPX cross-section library defaults to match MIPLIB, including the use of unit 70 for unrecognized libraries. (7) Corrected an error that occurred if light element nuclides are in both the ORIGEN-S light element and fission product libraries and are input as individual nuclides in the MIPLIB input.

**ORIGEN-S** : (1) Replaced call to INQOPN with call to OPNFIL for standard input (unit 5) to improve portability to other computers. (2) Initialize arrays only within the array boundaries for better portability. (3) Added EXTERNAL statements for BLOCK DATA files to improve portability. (4) Improved coding for the integral option using the actinide library. (5) Program was corrected to execute a continuation subcase when using the integral option. (6) Corrected call to subroutine STOP.

**ORIGEN-S Libraries**: (1) Corrected Cf-249 data in lowest energy group in Master Photon Library. It was high by factor of 100. (2) Corrected the half-life of Se-79. It was low by factor of 10. (3) Corrected Br-86m decay data that was erroneously changed in ENDF/B-VI update.

**COUPLE**: (1) Added check for correct ordering of fission product nuclides by atomic weight in input library. (2) Added EXTERNAL statement for BLOCK DATA files to improve portability to other computers. (3) Modified output format for THERM, RES, and FAST from floating point (F) to exponential (E). (4) Corrected an error that occurred when only fission product yield data were being updated. Note that this error did not occur when using COUPLE in the SAS2 sequence.

**Standard Composition Library**: (1) Revised the resonance energy scattering and/or total cross sections for 197 materials based on the 44-group ENDF/B-V cross-section library. Consistent values were specified for both the natural element and individual isotopes B, Zr, Eu, Hf, and W. (2) Added new composition ZIRC2 for Zircaloy-2 when using ENDF/B-V cross-section libraries. (3) Replaced $\sigma_s$ and $\sigma_t$ values in Nuclide Information Table with average values calculated from ENDF/B-V or ENDF/B-VI libraries. (4) Density for ZIRCALLOY and ZIRC2 was corrected from 6.506 to 6.56 g/cc.

**All Cross-Section Libraries:** The atomic masses of nuclides were modified to all be in terms of neutron mass for consistency.

**XSDRNPM**: (1) Modified to scale the weighting flux so that minimum value is 1E-20. This modification was made to reduce the likelihood that an underflow would occur when creating a weighted cross-section library for problems where there are large variations in the flux (many orders of magnitude). (2) Corrected an error that caused broad-group-flux edits to print bad values. (3) Corrected an error that would occur when creating a gamma-only broad-group library.

**XSDRNPM/SAS4**: Adjoint calculation in XSDRNPM for SAS4 was corrected. Only affects fixed-source calculations with upscatter on machines whose single- and double-precision formats are different (includes IBM workstations). Any such calculations should be rerun to determine impact of correction.

**SAS4**: (1) Increased the number of mesh intervals for both neutrons and gammas in the XSDRN adjoint calculation to improve the likelihood of convergence. Made additional minor changes to improve portability. (2) Updated for consistency with new nuclide ID numbering scheme in MIPLIB.

**MORSE**: Corrected a problem caused by the use of an incorrect normalization factor on standard deviations at the end of each batch. This occurred when NMOST was equal to NSTRT.

**BONAMI**: (1) Restored Bondarenko factor printout that was inadvertently deleted in July 1990 modification. Results were not impacted. (2) Corrected so that calculations with more than one nuclide specified with a zero number density will run.

**SAS1**: (1) Modified to force the pointer for the double-precision flux array to start on a double word boundary. This change improves performance significantly on some workstations other than IBM. (2) Corrected setting of termination flag so that SCALE driver can run other modules when cases are stacked in an input file.

**AIM**: Corrected error that created a bad magic word in a binary library if a magic word position occurred as the last position of a transfer array in the BCD data.

**HEATING**: A coding flaw was found and corrected in HEATING that prevented the proper application of the unit system conversion factors for the material properties database. This option allows the user to select a material from the database and convert the property values to the user's unit system. Due to the flaw, the material property values were not converted and values were used in the calculations that were not correct for the unit system. This flaw would affect only HEATING or HTAS1 problems that specify unit conversion factors for the material properties database.

**HTAS1**: A coding flaw was found and corrected in HTAS1 that prevented the building and the use of enclosure radiation connectors for models that replace the material in the neutron shield with a void (no material). A normal (and the default) analysis would replace the material in the neutron shield with air. Any other material or a void can be placed in the neutron shield volume. As long as any type of material (not a void) is in the neutron shield then the thermal radiation connectors are built in OCULAR and the analysis procedures correctly. However, if a void volume is modeled in HTAS1 then the flaw prevented the building and use of the connectors. Thus, no heat would be transferred across the neutron shield, which could significantly and potentially nonconservatively alter the thermal history of the shipping cask.

**All modules (UNIXLIB)**: Modified subroutine DATIM to read the date and time from the file "datetime.std" if it exists. This option will be useful for comparing sample problem output files.

**KENO-V.a, SAS2 (SUBLIB)**: Modified subroutine YREAD to correctly read an array of numbers separated by commas.

## Portability

Version 4.3 of the SCALE system has been developed to ensure portability among various computing platforms. The system is maintained and enhanced at ORNL under quality assurance and configuration management plans. The system has been routinely tested on IBM and DEC

workstations. In addition, the system has been applied by ORNL on SUN and HP workstations. Information needed to install SCALE on each of these systems is included with the software package distributed by the code centers. A separate SCALE software package, designated SCALE-PC, is available for the execution of the SCALE system, except heat transfer modules, on a MS-DOS personal computer.

**Related developments**

The definition of "easy-to-use" has changed considerably since the late 1970s. As funding has allowed, the ORNL development staff has sought to develop user interfaces that provide a distinct aid to novice or occasional users of the system. These full-screen input processors were developed to work on a MS-DOS personal computer and provide interactive help to the user in preparing accurate input for a SCALE module. Currently, input processors are available only for the criticality control sequences[2] and the ORIGEN-S functional module. [3]The SCALE-PC version includes an updated version of the ORIGEN-S processor coupled with the ARP code, which interpolates on standard LWR ORIGEN-S binary libraries, in the ORIGEN-ARP system (Section F7.A).

**Availability**

The SCALE code system and the other software designated under Related Developments have been packaged by the Radiation Shielding Information Center (RSIC). The SCALE system and the related software may be obtained by contacting

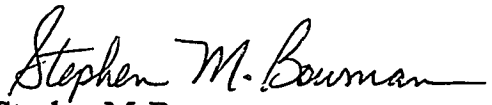
Radiation Shielding Information Center
Oak Ridge National Laboratory
P.O. Box 2008
Oak Ridge, TN 37831-6362
Telephone: (615) 574-6176
FAX: (615) 574-6182
E-mail: pdc@ornl.gov

**Acknowledgments**

The SCALE system is maintained at ORNL and enhanced to keep pace with normal technical advancements in the analysis areas of interest. Although the NRC continues its role as the controlling sponsor of the SCALE system, the U.S. Department of Energy (DOE) began assisting in the maintenance of the SCALE system in 1987. Over the years numerous individuals within these sponsoring organizations have played key roles in ensuring that the SCALE system remained a readily available, reliable system for the analysis of nuclear fuel facilities and packages. The individuals who have worked with the ORNL staff to coordinate maintenance and development activities include R. H. Odegaarden (NRC, ret.), G. H. Bidinger (NRC, ret.), C. Mauck (DOE, ret.), E. P. Easton (NRC), W. H. Lake (NRC and DOE), M. E. Wangler (DOE), and M. G. Bailey (NRC).

As demonstrated by this Manual, there are also numerous individuals from the ORNL staff who have contributed significantly to the development and enhancement of the SCALE system. Most

are credited by their authorship of the sections in this Manual that correspond to their work. A few individuals have been essential to the development and maintenance of SCALE but are not credited by authorship. These individuals include: S. K. Martin, who is responsible for implementing and controlling software system changes; C. H. Shappert, who provided the editorial review of this Manual; and L. F. Norris, who prepared the entire manuscript. Special acknowledgement is also due to R. M. Westfall and G. E. Whitesides who, together with R. H. Odegaarden of the NRC, developed the concept and long-range goals of the SCALE system in the late 1970s. Finally, this Project Leader will always be grateful to C. V. Parks, who has served as the SCALE Project Leader for the past 15 years, and L. M. Petrie, who for nearly 20 years has consistently provided consultation and advice on the technical direction that should be taken in development of nearly every module and cross-section library that are in the present system.



Stephen M. Bowman
SCALE Project Leader
January 1997

**References**


1. Greene, N. M., Ford, W. E., III, Petrie, L. M., and Arwood, J. W., *AMPX-77: A Modular Code System for Generating Coupled Multigroup Neutron-Gamma Cross-Section Libraries from ENDF/B-IV and/or ENDF/B-V*, ORNL/CSD/TM-283, Martin Marietta Energy Systems, Inc., Oak Ridge Natl. Lab., October 1992.

2. S. M. Bowman *OFFSCALE: A PC Input Processor for the SCALE Code System – The CSASIN Processor for the Criticality Sequences*, NUREG/CR-6182, Vol. 1 (ORNL/TM-12263/V1), U.S. Nuclear Regulatory Commission, November 1994.

3. S. M. Bowman *OFFSCALE: A PC Input Processor for the SCALE Code System – The ORIGNATE Processor for ORIGEN-S*, NUREG/CR-6182, Vol. 2 (ORNL/TM-12263/V2), U.S. Nuclear Regulatory Commission, November 1994.

4. Broadhead, B.L., *QADS: A Multidimensional Point Kernel Analysis Module*, NUREG/CR-5468 (ORNL/CSD/TM-270), U.S. Nuclear Regulatory Commission, May 1990.

5. N. M. Greene, J. W. Arwood, R. Q. Wright, C. V. Parks, *The LAW Library–A Multigroup Cross-Section Library for Use in Radioactive Waste Analysis Calculations*, ORNL/TM-12370, Martin Marietta Energy Systems, Inc., Oak Ridge Natl. Lab., August 1994.

6. M. D. DeHart and S. M. Bowman, *Validation of the SCALE Broad Structure 44-Group ENDF/B-V Cross-Section Library for Use in Criticality Safety Analyses*, NUREG/CR-6102 (ORNL/TM-12460), U.S. Nuclear Regulatory Commission, September 1994.

Table 1 Analysis capabilities summary of the SCALE control modules

| Control module | Analysis function(s) | Functional modules executed | Section reference |
|---|---|---|---|
| CSAS | 1-D deterministic calculation of neutron multiplication<br>3-D Monte Carlo calculation of neutron multiplication<br>Problem-dependent cross-section processing<br>Multiplication search or spacing | BONAMI<br>NITAWL-II<br>XSDRNPM<br>KENO V.a<br>ICE | C4 |
| CSAS6 | 3-D Monte Carlo calculation of neutron multiplication | BONAMI<br>NITAWL-II<br>XSDRNPM<br>KENO-VI | C6 |
| SAS1 | 1-D deterministic calculation of radiation transport through shield and dose evaluation at a point<br>Calculation of dose at detector based on leakage from critical volume | BONAMI<br>NITAWL-II<br>XSDRNPM<br>XSDOSE | S1 |
| SAS2 | Point depletion/decay of nuclear fuel<br>1-D radial shielding analysis in cylindrical geometry | BONAMI<br>NITAWL-II<br>XSDRNPM<br>COUPLE<br>ORIGEN-S<br>XSDOSE | S2 |
| SAS3 | Dose evaluation using MORSE Monte Carlo code | BONAMI<br>NITAWL-II<br>XSDRNPM<br>MORSE-SGC | S3 |
| SAS4 | Calculation of dose outside of transportation package using MORSE code and automated biasing techniques | BONAMI<br>NITAWL-II<br>XSDRNPM<br>MORSE-SGC | S4 |
| QADS | 3-D point-kernel gamma-ray shielding analysis | QAD-CGGP | S5 |
| HTAS1 | R-Z steady-state and transient analyses of a transportation package | OCULAR<br>HEATING | H1 |

Table 2 Analysis capabilities summary of the SCALE functional modules

| Module | Function | Section reference |
|---|---|---|
| BONAMI | Resonance self-shielding of cross sections with Bondarenko factors | F1 |
| NITAWL-II | Resonance self-shielding of cross sections with resolved resonance data | F2 |
| XSDRNPM | General 1-D, discrete-ordinates code for:<br>• zone-weighting of cross sections<br>• eigenvalue calculations for neutron multiplication<br>• fixed-source calculation for shielding analysis<br>• adjoint calculation for determining importance functions | F3 |
| XSDOSE | Module for calculation of dose at a point based on the 1-D leakage flux from a finite shield | F4 |
| COUPLE | Interface module for preparation of cross-section and spectral data for ORIGEN-S | F6 |
| ORIGEN-S | General-purpose point-depletion and decay code to calculate isotopic, decay heat, radiation source terms, and curie levels | F7 |
| ICE | Cross-section utility module for mixing cross sections | F8 |
| MORSE-SGC | Monte Carlo code with combinatorial and array geometry features used to perform radiation shielding analysis | F9 |
| HEATING7.2 | Finite-volume, multidimensional code for conduction and radiation heat transfer | F10 |
| KENO V.a | Monte Carlo code for calculation of neutron multiplication factors | F11 |
| OCULAR | Calculation of radiation exchange factors | F16 |
| KENO-VI | Monte Carlo code for calculation of neutron multiplication factors for complex geometries | F17 |

NUREG/CR-0200
Revision 5
Volume 1, Section C4
ORNL/NUREG/CSD-2/V2/R5

Computational Physics and Engineering Division

# CSAS: CONTROL MODULE FOR ENHANCED CRITICALITY SAFETY ANALYSIS SEQUENCES

N. F. Landers*
L. M. Petrie

Date Published: March 1997

Prepared for the
Office of Nuclear Material Safety and Safeguards
U. S. Nuclear Regulatory Commission
Washington, DC 20555
Under Interagency Agreement DOE 1886-C000-9B
NRC JCN No. B0009

Prepared by the
OAK RIDGE NATIONAL LABORATORY
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORP.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-96OR22464

*Formerly with Oak Ridge National Laboratory.

## ABSTRACT

The Criticality Safety Analysis Sequences (CSAS) were developed within the SCALE code system to provide automated, problem-dependent, cross-section processing followed by calculation of the neutron multiplication factor for the system being modeled. This document describes the program that contains the available sequences.

These control sequences activate the cross-section processing codes BONAMI and NITAWL-II to provide resonance-corrected cross sections. Control sequences ending with "X" utilize XSDRNPM to provide a cell-weighted mixture cross section and/or the neutron multiplication factor (k-effective) for a one-dimensional system model. ICE is used by CSASI and CSASIX to provide a Monte Carlo-formatted mixed cross-section library. KENO V.a uses the processed cross sections and calculates the k-effective of three-dimensional system models. The geometric modeling capabilities available in KENO V.a coupled with the automated cross-section processing within the control sequences allow complex, three-dimensional systems to be easily analyzed. A search capability is achieved by repeatedly activating the control module MODIFY to alter the pitch or other dimensions and the functional module KENO V.a to calculate the k-effective for the new pitch or dimensions.

The control sequences (sometimes referred to as modules) described within this document were developed simultaneously and are closely related. These sequences are listed in the following table with the modules they invoke:

| Control module | Search function | Functional modules executed by the control module | | | | |
|---|---|---|---|---|---|---|
| CSASI | No search | BONAMI | NITAWL-II | | ICE | |
| CSASIX | No search | BONAMI | NITAWL-II | XSDRNPM | ICE | |
| CSASN | No search | BONAMI | NITAWL-II | | | |
| CSAS1X | No search | BONAMI | NITAWL-II | XSDRNPM | | |
| CSAS25 | No search | BONAMI | NITAWL-II | | KENO V.a | |
| CSAS2X | No search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | |
| CSAS4 | Search | BONAMI | NITAWL-II | | KENO V.a | MODIFY[a] |
| CSAS4X | Search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | MODIFY[a] |

[a]MODIFY is a control module.

## CONTENTS


Page

ABSTRACT ..... C4.iii
LIST OF FIGURES ..... C4.viii
LIST OF TABLES ..... C4.x
ACKNOWLEDGMENTS ..... C4.xi

C4.1 INTRODUCTION ..... C4.1.1

C4.2 TECHNIQUES ..... C4.2.1
C4.2.1 OPTIMUM SEARCH ..... C4.2.2
C4.2.2 CRITICAL SEARCH ..... C4.2.3
C4.2.3 CSAS LIMITATIONS ..... C4.2.3
C4.2.4 REFERENCES ..... C4.2.5

C4.3 LOGICAL PROGRAM FLOW ..... C4.3.1
C4.3.1 INITIATION OF CSAS (CRITICALITY SAFETY ANALYSIS SEQUENCE NO. 4) ..... C4.3.1
C4.3.2 OVERALL FLOW DIAGRAM ..... C4.3.5
C4.3.3 THE MATERIAL INFORMATION PROCESSOR ..... C4.3.10
C4.3.4 KENO V.a DATA PROCESSOR ..... C4.3.10
C4.3.4.1 Control of the Data Input for KENO V.a ..... C4.3.13
C4.3.4.2 Read KENO V.a Array Data ..... C4.3.16
C4.3.4.3 Read KENO V.a Geometry Data ..... C4.3.17
C4.3.4.4 Read KENO V.a Mixing Table Data ..... C4.3.18
C4.3.4.5 Read KENO V.a Plot Data ..... C4.3.19
C4.3.4.6 Read KENO V.a Biasing Data ..... C4.3.20
C4.3.4.7 Search Type Specification ..... C4.3.21
C4.3.4.8 Generate Remaining KENO V.a Data ..... C4.3.23
C4.3.4.9 Process the KENO V.a Geometry Data ..... C4.3.25
C4.3.4.10 Load KENO V.a Data from the Direct-Access File ..... C4.3.26
C4.3.4.11 Check the KENO V.a Geometry Data ..... C4.3.27
C4.3.4.12 Determine Distances ..... C4.3.29
C4.3.4.13 Calculate KENO V.a Geometry Volumes ..... C4.3.31
C4.3.4.14 Generate Printer Plot ..... C4.3.32
C4.3.4.15 Write KENO V.a Input Data on Restart File ..... C4.3.33
C4.3.5 THE SEARCH DATA PROCESSOR ..... C4.3.35
C4.3.6 MODIFY, THE SEARCH MODULE ..... C4.3.37
C4.3.7 REFERENCE ..... C4.3.40

C4.4 INPUT DATA GUIDE ..... C4.4.1
C4.4.1 INPUT DATA SUMMARY FOR EXPERIENCED USERS ..... C4.4.3
C4.4.2 INPUT DATA REQUIREMENTS AND DATA CHECKING ..... C4.4.23
C4.4.3 MATERIAL INFORMATION PROCESSOR DATA ..... C4.4.26
C4.4.4 STANDARD COMPOSITIONS SPECIFICATION DATA ..... C4.4.28

C4.4.5 OPTIONAL UNIT CELL SPECIFICATION FOR INFINITE HOMOGENEOUS PROBLEMS ..... C4.4.35
C4.4.6 UNIT CELL SPECIFICATION FOR LATTICECELL PROBLEMS ..... C4.4.37
C4.4.7 UNIT CELL SPECIFICATION FOR MULTIREGION PROBLEMS ..... C4.4.42
C4.4.8 OPTIONAL PARAMETER DATA ..... C4.4.46
C4.4.9 SUMMARY OF KENO V.a DATA ..... C4.4.51
C4.4.10 SEARCH DATA ..... C4.4.51
C4.4.10.1 Search Type Specification ..... C4.4.62
C4.4.10.2 Auxiliary Search Commands and Constraints ..... C4.4.64

C4.5 NOTES FOR CSAS USERS ..... C4.5.1
C4.5.1 CSAS MODULES ..... C4.5.1
C4.5.2 AUTOMATIC CROSS-SECTION PROCESSING FOR CRITICALITY CALCULATIONS ..... C4.5.5
C4.5.2.1 Run KENO V.a using CSAS25 or CSAS4 ..... C4.5.5
C4.5.2.2 Run KENO V.a using CSAS2X or CSAS4X ..... C4.5.9
C4.5.2.3 Run XSDRNPM using CSAS1X ..... C4.5.11
C4.5.3 AUTOMATED CROSS-SECTION PREPARATION FOR USE BY SUBSEQUENT CODES ..... C4.5.12
C4.5.3.1 Create a Microscopic, AMPX Working Format, Cross-Section Library for Use by KENO V.a ..... C4.5.12
C4.5.3.2 Create a Microscopic, Cell-Weighted AMPX Working Format, Cross-Section Library ..... C4.5.14
C4.5.3.3 Create a Macroscopic, AMPX Working Format, Cross-Section Library and a Macroscopic, MORSE/KENO Cross-Section Library .... C4.5.19
C4.5.3.4 Create a Macroscopic, Cell-Weighted, AMPX Working Format, Library and a Macroscopic, Cell-Weighted, MORSE/KENO Library ... C4.5.22
C4.5.4 CREATE AND USE TWO CELL-WEIGHTED MIXTURES IN A PROBLEM ..... C4.5.24
C4.5.5 CREATE A CELL-WEIGHTED MIXTURE REPRESENTING A NONHOMOGENEOUS FUEL ASSEMBLY ..... C4.5.28
C4.5.6 SEARCH DATA ..... C4.5.30
C4.5.6.1. Search Type Specification ..... C4.5.30
C4.5.6.2 Auxiliary Search Commands ..... C4.5.32
C4.5.7 SEARCH CONSIDERATIONS ..... C4.5.41
C4.5.8 OPTIMUM PITCH SEARCH ..... C4.5.42
C4.5.9 OPTIMUM DIMENSION SEARCH ..... C4.5.46
C4.5.10 CRITICAL PITCH SEARCH ..... C4.5.52
C4.5.11 CRITICAL DIMENSION SEARCH ..... C4.5.57
C4.5.12 CONCENTRATION SEARCH CONSIDERATIONS ..... C4.5.66
C4.5.13 OPTIMUM CONCENTRATION SEARCH ..... C4.5.67
C4.5.14 CRITICAL CONCENTRATION SEARCH ..... C4.5.68

C4.6 DESCRIPTION OF OUTPUT ..... C4.6.1
C4.6.1 CONTROL MODULE HEADER PAGE ..... C4.6.1
C4.6.2 CONTROL MODULE PROGRAM VERIFICATION TABLE ..... C4.6.2
C4.6.3 MATERIAL INFORMATION PROCESSOR OUTPUT ..... C4.6.3

C4.6.4 KENO V.a OUTPUT ..... C4.6.4
C4.6.5 PROGRAM VERIFICATION TABLE FOR THE CONTROL MODULE MODIFY ..... C4.6.5
C4.6.6 TABLE OF SEARCH DATA ..... C4.6.6
C4.6.7 SEARCH PASS ITERATION DATA ..... C4.6.7
C4.6.8 SEARCH CONVERGENCE INFORMATION ..... C4.6.8
C4.6.9 EXTRAPOLATION OF CONVERGED DATA ..... C4.6.9

C4.7 WARNING AND ERROR MESSAGES ..... C4.7.1
C4.7.1 MATERIAL INFORMATION PROCESSOR MESSAGES ..... C4.7.1
C4.7.2 MESSAGES FROM CRITICALITY SAFETY SEQUENCE NO. 4 ..... C4.7.19
C4.7.3 STOP CODES ..... C4.7.25
C4.7.4 MESSAGES ASSOCIATED WITH STOP 20 IN CSAS ..... C4.7.27
C4.7.5 UNNUMBERED MESSAGES ..... C4.7.28

C4.A ALPHABETICAL INDEX OF SUBROUTINES ..... C4.A.1
C4.B SAMPLE PROBLEMS ..... C4.B.1
C4.B.1 REFERENCES ..... C4.B.6
C4.C SAMPLE OUTPUT ..... C4.C.1

## LIST OF FIGURES


Page

C4.3.1 Control modules available in Criticality Safety Analysis Sequence No. 4 ..... C4.3.4
C4.3.2 Overall flow of Criticality Safety Analysis Sequence No. 4 ..... C4.3.6
C4.3.3 Flowchart of KENO V.a data processor ..... C4.3.11
C4.3.4 Flowchart of KENO V.a data reading routines ..... C4.3.13
C4.3.5 Flowchart for reading KENO V.a array data ..... C4.3.16
C4.3.6 Flowchart of routines to read the KENO V.a geometry data ..... C4.3.17
C4.3.7 Flowchart for reading KENO V.a mixing table data ..... C4.3.19
C4.3.8 Flowchart for reading KENO V.a plot data ..... C4.3.20
C4.3.9 Flowchart for reading KENO V.a biasing data ..... C4.3.20
C4.3.10 Flowchart for reading KENO V.a restart data ..... C4.3.22
C4.3.11 Flowchart for generating remaining KENO V.a data ..... C4.3.23
C4.3.12 Flowchart for processing KENO V.a geometry data ..... C4.3.25
C4.3.13 Flowchart for loading data from direct access ..... C4.3.26
C4.3.14 Flowchart of geometry-checking procedure ..... C4.3.28
C4.3.15 Flowchart for distance to intersection ..... C4.3.30
C4.3.16 Flowchart for calculating KENO V.a geometry volumes ..... C4.3.31
C4.3.17 Flowchart for printer plots ..... C4.3.32
C4.3.18 Flowchart for writing data on the restart data file ..... C4.3.34
C4.3.19 Flowchart of the Search Data Processor ..... C4.3.35
C4.3.20 Flowchart of the search module, MODIFY ..... C4.3.38
C4.4.1 Outline of CSAS input data ..... C4.4.4
C4.4.2 Arrangement of materials in a SQUAREPITCH and SPHSQUAREP unit cell ..... C4.4.40
C4.4.3 Arrangement of materials in a TRIANGPITCH and SPHTRIANGP unit cell ..... C4.4.40
C4.4.4 Arrangement of materials in a SYMMSLABCELL unit cell having reflected left and right boundary conditions ..... C4.4.40
C4.4.5 Arrangement of materials in an ASQUAREPITCH or ASPHSQUAREP unit cell ..... C4.4.41
C4.4.6 Arrangement of materials in an ATRIANGPITCH or ASPHTRIANGP unit cell ..... C4.4.41
C4.4.7 Arrangement of materials in an ASYMSLABCELL unit cell ..... C4.4.41
C4.5.1 Mixing table from CSASN with PARM=CHECK ..... C4.5.13
C4.5.2 Mixing table from CSASN edited to KENO V.a format ..... C4.5.13

C4.5.3 Mixing table from CSASN edited to KENO V.a format ..... C4.5.14
C4.5.4 Second mixing table from CSASIX ..... C4.5.15
C4.5.5 Mixing table from CSASIX edited to KENO V.a format ..... C4.5.16
C4.5.6 Mixing tables from CSASIX ..... C4.5.17
C4.5.7 KENO V.a mixing table using all CSASIX entries ..... C4.5.18
C4.5.8 KENO V.a mixing table using only mixtures 4, 5 and 6 from CSASIX ..... C4.5.18
C4.5.9 Condensed KENO V.a mixing table using only mixtures 4, 5, and 6 from CSASIX ..... C4.5.19
C4.5.10 1.98% enriched $UO_2$ fuel assembly ..... C4.5.28
C4.6.1 Sample control module header page ..... C4.6.1
C4.6.2 Sample control module program verification table ..... C4.6.2
C4.6.3 Part 1 of Material Information Processor printout ..... C4.6.3
C4.6.4 Part 2 of Material Information Processor printout ..... C4.6.4
C4.6.5 Summary of KENO V.a I/O usage by CSAS control module ..... C4.6.5
C4.6.6 Program verification table for MODIFY ..... C4.6.6
C4.6.7 Table of search data ..... C4.6.7
C4.6.8 Search pass iteration data ..... C4.6.8
C4.6.9 Search pass iteration data ..... C4.6.8
C4.6.10 Best estimate of the parameter and geometry ..... C4.6.9
C4.B.1 Critical assembly of four solution units and four metal units ..... C4.B.2
C4.B.2 Critical assembly of eight highly enriched uranium metal cylinders ..... C4.B.5

## LIST OF TABLES


| | | Page |
|---|---|---|
| C4.2.1 | CSAS and related criticality sequences | C4.2.1 |
| C4.3.1 | Description of units utilized by the Criticality Safety Analysis Sequence No. 4 control module | C4.3.2 |
| C4.3.2 | Description of units used by the functional modules of CSAS | C4.3.3 |
| C4.4.1 | Functional modules executed by CSAS control modules | C4.4.1 |
| C4.4.2 | I/O units on which cross-section libraries are written | C4.4.2 |
| C4.4.3 | I/O units generated by CSAS and associated modules | C4.4.2 |
| C4.4.4 | Input data requirements | C4.4.5 |
| C4.4.5 | Summary of KENO V.a input data | C4.4.14 |
| C4.4.6 | Summary of SEARCH input data | C4.4.24 |
| C4.4.7 | Summary of data input requirements | C4.4.26 |
| C4.4.8 | Outline of Material Information Processor data | C4.4.27 |
| C4.4.9 | Outline of standard composition specification data | C4.4.29 |
| C4.4.10 | Unit cell specification for INFHOMMEDIUM problems | C4.4.36 |
| C4.4.11 | Unit cell specification for latticecell problems | C4.4.38 |
| C4.4.12 | Geometry specification for multiregion problems | C4.4.43 |
| C4.4.13 | Optional parameter data | C4.4.47 |
| C4.4.14 | Outline of KENO V.a data | C4.4.52 |
| C4.4.15 | Summary of KENO V.a data | C4.4.53 |
| C4.4.16 | Outline of search type specification | C4.4.62 |
| C4.4.17 | Outline of auxiliary search commands and constraints | C4.4.65 |
| C4.5.1 | Description of units utilized by the Criticality Safety Analysis Sequence No. 4 control module | C4.5.2 |
| C4.5.2 | Description of units used by the functional modules of CSAS | C4.5.3 |
| C4.6.1 | Location of functional output | C4.6.1 |
| C4.A.1 | Alphabetical index of CSAS routines | C4.A.2 |
| C4.A.2 | Alphabetical index of MODIFY routines | C4.A.32 |
| C4.A.3 | Commons used in CSAS | C4.A.36 |
| C4.A.4 | Commons used in MODIFY | C4.A.47 |

## ACKNOWLEDGMENTS

CSAS and its related Criticality Safety Analysis Sequences are based on the old CSAS2 control module, which was not written in VS FORTRAN and, therefore, is not included in this version of SCALE, and the KENO V.a functional module described in Sect. F11. Therefore, special acknowledgment is made to J. A. Bucholz, R. M. Westfall, and J. R. Knight who developed CSAS2. G. E. Whitesides is acknowledged for his contributions through early versions of KENO. Appreciation is expressed to C. V. Parks for his guidance in developing CSAS.

## C4.1 INTRODUCTION

Within the life cycle of the SCALE system, the CSAS program has superseded earlier programs that contained criticality safety analysis sequences. Embedded within the CSAS program are numerous sequences that enable automated cross-section processing and criticality analyses with and without the search option. All the criticality analysis sequences (sometimes referred to as modules) within SCALE are contained within the CSAS program.

Criticality Safety Analysis Sequence No. 4 (CSAS4) was developed to provide a search capability for three-dimensional (3-D) configurations in the SCALE system. Because it encompassed all the options and functional modules available in earlier SCALE criticality sequences, the program was initially called by the same name, CSAS4. In order to distinguish the program from the numerous sequences that it contained (including CSAS4), the program name was later changed to CSAS. The reader should be aware that the documentation has not been completely revised, and the program is still referred to as CSAS4 in much of the documentation that follows.

The standardized automated procedures process SCALE cross sections using the Bondarenko method (via BONAMI) and the Nordheim integral method (via NITAWL-II) to provide a resonance-corrected cross-section library based on the physical characteristics of the problem being analyzed. This cross-section library can be utilized by KENO V.a, a 3-D multigroup Monte Carlo criticality program, or XSDRNPM, a one-dimensional (1-D) discrete-ordinates code for transport analysis. The search capability utilizes KENO V.a and is performed by activating CSAS4 or CSAS4X. The type of search is defined by the input data, and a parameter search is performed on k-effective as a function of pitch or other dimensions. The two basic search options offered are (1) an optimum search seeking a maximum or minimum and (2) a critical search seeking a fixed value of k-effective.

The user may select the CSAS4 or the CSAS4X sequence to perform a search sequence. An X in the module name specifies that XSDRNPM will be used to provide cell-weighted cross sections. CSAS4 should be used for 3-D searches when the fuel pins are modeled individually in the KENO V.a geometry. CSAS4X should be used for 3-D searches when cell-weighted cross sections are used to provide a homogeneous representation of the fuel pins in the KENO V.a geometry. Cell-weighting is accomplished by describing a unit cell, as in a fuel assembly, for which a 1-D eigenvalue calculation of the unit cell is made to determine the spatially dependent flux spectrum. This flux spectrum is used to cell-weight the microscopic cross-section data. The nuclide number densities in the unit cell are homogenized, and the resultant homogenized cell-weighted cross sections are identified as mixture 500 and utilized in the KENO V.a geometry. Note that only one mixture 500 can be created per problem.

The codes utilized in Criticality Safety Analysis Sequence No. 4 start with an AMPX master format cross-section library and generate a self-shielded, group-averaged library applicable to the specific problem configuration. These cross sections are then used in the Monte Carlo determination of the effective neutron multiplication factor. The codes and their functions are given below.

1. BONAMI — performs resonance self-shielding calculations for nuclides that have Bondarenko data associated with their cross sections.

2. NITAWL-II — applies a Nordheim resonance self-shielding correction to nuclides having resonance parameters.

3. XSDRNPM — provides cell-weighted cross sections based on the specified unit cell and, in addition, can calculate the effective neutron multiplication (k-effective) for a 1-D system.

4. ICE — creates a Monte Carlo formatted mixed cross-section library for use by KENO V.a.

5. KENO V.a — calculates k-effective of a 3-D system.

6. MODIFY — alters the pitch and redefines the geometry data used by KENO V.a. The optimum pitch search is achieved by repeatedly changing the geometry data and executing KENO V.a until the optimum pitch is determined or the calculation is terminated for other reasons.

All the control sequences in the CSAS program are listed in the following table with the modules they invoke. The first seven sequences are subsets of the CSAS4X sequence. This section of the SCALE manual describes all eight sequences.

| Control module | Search function | Functional modules executed by the control module | | | | | |
|---|---|---|---|---|---|---|---|
| CSASI | No search | BONAMI | NITAWL-II | | ICE | | |
| CSASIX | No search | BONAMI | NITAWL-II | XSDRNPM | ICE | | |
| CSASN | No search | BONAMI | NITAWL-II | | | | |
| CSAS1X | No search | BONAMI | NITAWL-II | XSDRNPM | | | |
| CSAS25 | No search | BONAMI | NITAWL-II | | KENO V.a | | |
| CSAS2X | No search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | | |
| CSAS4 | Search | BONAMI | NITAWL-II | | KENO V.a | MODIFY* | |
| CSAS4X | SYarch | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | MODIFY* | |

*MODIFY is a control module.

## C4.2 TECHNIQUES

Criticality Safety Analysis Sequence No. 4 (CSAS) is designed to perform a search utilizing the Monte Carlo Criticality code KENO V.a. Related sequences (see Table C4.2.1) can bypass the search and provide k-effective for the system being analyzed or simply prepare a cross-section library for subsequent use. In order to minimize human error, the SCALE data handling is automated as much as possible. CSAS and many other SCALE sequences apply a standardized procedure to provide appropriate cross sections for the calculation. This procedure is carried out by the Material Information Processor that generates number densities and related information, prepares geometry data for resonance self-shielding and flux-weighting cell calculations, and creates data input files for the cross-section processing codes. Sequences that execute KENO V.a include a KENO V.a Data Processor to read and check the KENO V.a data. Sequences that execute a search use a Search Data Processor to read and check the search data. When the data checking has been completed, the control sequence executes BONAMI and NITAWL-II to prepare a resonance-corrected microscopic cross-section library in the AMPX working library format. If cell-weighted cross sections are requested, XSDRNPM performs the necessary calculations and produces a cell-weighted microscopic cross-section library in the AMPX working library format. ICE may be executed to produce mixture libraries (macroscopic cross-section libraries) in the ICE Monte Carlo format and the AMPX working library format. KENO V.a may be executed to calculate the k-effective or neutron multiplication factor using the cross-section library that was prepared by the control sequence. MODIFY may be invoked to perform a search* by repeatedly altering the KENO V.a data and executing KENO V.a.

Table C4.2.1 CSAS and related criticality sequences

| Control module | Search function | Functional modules executed by the control module | | | | |
|---|---|---|---|---|---|---|
| CSASI | No search | BONAMI | NITAWL-II | | ICE | |
| CSASIX | No search | BONAMI | NITAWL-II | XSDRNPM | ICE | |
| CSASN | No search | BONAMI | NITAWL-II | | | |
| CSAS1X | No search | BONAMI | NITAWL-II | XSDRNPM | | |
| CSAS25 | No search | BONAMI | NITAWL-II | | KENO V.a | |
| CSAS2X | No search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | |
| CSAS4 | Search | BONAMI | NITAWL-II | | KENO V.a | MODIFY[a] |
| CSAS4X | Search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | MODIFY |

[a]MODIFY is a control module.

The Material Information Processor is described in Sect. M7. It is responsible for reading the standard composition data and other engineering-type specifications, including volume fraction or percent theoretical density, temperature, and isotopic distribution. The techniques used in the Material Information Processor and their applications and limitations are discussed in Sect. M7.2. The input data for the Material Information Processor is the same for all eight analytical sequences available through CSAS. For convenience, the Material Information Processor input data guide is included in Sect. C4.4.

*This search technique is discussed in detail in the report by M. J. Lorek, *Improved Criticality Search Techniques for Low- and High-Enriched Systems*, NUREG/CR-2122, ORNL/NUREG/CSD/TM-13, NUREG/CR-2122, 1981.

The Material Information Processor checks the input data pertaining to cross-section preparation and prepares binary input files for the applicable functional modules. Depending on the specified sequence, data is prepared for BONAMI, NITAWL-II, XSDRNPM, and ICE as appropriate.

The Bondarenko method is discussed in the Material Information Processor document, Sect. M7.2.5.1, and BONAMI is described in Sect. F1. The Nordheim method and NITAWL-II are described in Sects. M7.2.5.2 and F2 respectively. The Dancoff treatment is discussed in Sect. M7.2.5.3. The automatic mesh generator, the automatic quadrature generator, and the convergence criteria for control sequences that execute XSDRNPM are described in Sects. M7.2.5.6, M7.2.5.7, and M7.2.5.8 respectively. XSDRNPM is described in Sect. F3. ICE is described in Sect. F8, and KENO V.a is described in Sect. F11. See Sect. M7A for an explanation of many of the techniques used in processing the cross sections.

The search capability is implemented by the control module MODIFY. It performs operations according to the specified search data to determine (1) the maximum (or minimum) value of k-effective as a function of pitch or dimensions or (2) the pitch or dimensions corresponding to a specified value of k-effective. An iterative procedure is used, making use of all previous information to modify the pitch or dimensions to achieve the desired result. The procedures for conducting optimum searches and critical searches are summarized below.

### C4.2.1 OPTIMUM SEARCH

Because only an initial value of k-effective and a set of boundary constraints are available, four initial points are generated spanning the range defined by the constraints. The search package identifies the type of cubic equation [i.e., a cubic with no local extrema (type A) or a cubic with two local extrema (type B)] and utilizes this knowledge in determining the pitch or dimensions corresponding to the maximum (or minimum) value of k-effective. The optimum search procedure is summarized as follows:

1. Calculate k-effective for the specified problem.
2. Calculate k-effective for the minimum constraint.
3. Calculate k-effective for the maximum constraint.
4. Calculate k-effective for a fourth point that lies approximately equidistant between the initial guess and the constraint that is farthest from it.
5. Utilize a weighted least-squares fit to a cubic polynomial on the data points.
6. Determine the type of cubic. For a type A cubic, go to step 11.
7. Take the first derivative of the least-squares cubic.
8. Solve the quadratic for its roots.
9. Take the second derivative of the least-squares cubic to determine which root is the maximum (or minimum), and if it falls within the constraints, use this root as the next guess. Otherwise, convergence has been defined as occurring at the constraint with the maximum (or minimum) k-effective.

10. Calculate the k-effective corresponding to the next guess. Go to step 5. Repeat this procedure until convergence is achieved.

11. If the cubic equation is a type A cubic, the optimum lies on one of the boundaries. If the fit shows that the cubic is actually a type B cubic, go to step 7 and continue.

Convergence is defined as occurring when a k-effective has been calculated for a point on the curve where the value of the curve is within epsilon of the maximum (or minimum) of the curve. Additionally, the calculated k-effective must be within two standard deviations of the value of the curve at that point. The search is terminated when convergence is achieved or when the code determines there is no local maximum within the constraints.

**NOTE:** The entire search is performed using the problem-dependent cross sections that were created using data derived from the initial cell description. Therefore, if the search converges with a configuration that is significantly different from the initial cell, the search should be rerun using the converged cell configuration to ensure that the cross-section treatment was adequate.

### C4.2.2 CRITICAL SEARCH

The critical search option searches for the pitch or dimensions corresponding to a specified value of k-effective. If the calculated value of k-effective is within the specified search tolerance (EPS) of the desired k-effective, the search is considered to be complete. The critical search procedure is summarized as follows:

1. Calculate k-effective for the specified problem. If it is within EPS of the specified k-effective, convergence has been achieved.

2. Calculate k-effective for one of the constraints. If the specified problem coincided with one of the constraints, calculate k-effective at a point halfway to the other constraint. If the calculated k-effective is within EPS of the specified k-effective, convergence has been achieved.

3. Calculate k-effective for a point chosen from a linear fit of the two existing points closest to the specified k-effective. If the calculated k-effective is within EPS of the specified k-effective, convergence has been achieved.

4. Repeat step 3.

5. Calculate k-effective for a point determined from fitting the previous points to a cubic and solving the cubic for the point closest to the desired k-effective. If all roots lie outside the constraints, the problem is terminated and an appropriate message is written. If the calculated k-effective is within EPS of the specified k-effective, convergence has been achieved. Repeat step 5 until the problem converges, exceeds the specified number of passes, or determines that a feasible solution does not exist.

It should be noted that the entire search is performed using the problem-dependent cross sections that were created using data derived from the initial cell description. Therefore, if the search converges with a configuration that is significantly different from the initial cell, the search should be rerun using the converged cell configuration to ensure that the cross-section treatment was adequate.

## C4.2.3 CSAS LIMITATIONS

The CSAS control module was developed to use simple input data and prepare problem-dependent cross sections for use in calculating the effective neutron multiplication factor of a 1-D or 3-D system using XSDRNPM or KENO V.a. An attempt was made to make the system as general as possible within the constraints of the standardized methods chosen to be used in SCALE. Standardized methods of data input were adopted to allow easy data entry and for quality assurance purposes. Some of the limitations of the CSAS sequences are due to assumptions in the Nordheim integral treatment[1] as implemented in CSAS. Inherent limitations in the NITAWL-II Nordheim integral treatment of resonance nuclides are as follows:

1. The treatment assumes no resonance overlap from other resonances or other material regions.

2. The treatment of an external moderator assumes an asymptotic flux present at the absorber-moderator interface.

3. The treatment of spatial transport uses the first-flight escape probability for the absorber, the two-region reciprocity theorem, and Dancoff[2] factors.

In many cases, the Nordheim integral treatment is an excellent method for processing resonance cross sections for use in the analysis of light-water reactor (LWR) fuel pin cells and for many other applications commonly used in neutronics analysis. However, caution should be used in applications involving several isotopes with large resonances or complicated absorber-moderator geometries. Another area of probable inadequacy involves the analysis of machinery for dissolving spent fuel elements. Such a system usually requires the simultaneous consideration of resonance absorbers located in both the solid fuel pellets and the acid solution, thereby resulting in an inadequate treatment by NITAWL-II.

The major limitations of the analytical model used by the Nordheim integral treatment in CSAS are as follows:

1. A lattice system whose fuel or moderator contains an absorber that has rapidly varying cross sections across the resonance region may be inadequately treated. Examples include gadolinium, silver, indium, and hafnium.

2. The presence of more than one resonance isotope in a material can lead to an effect called resonance overlap, which can result in enhanced or reduced group-averaged cross sections, depending on the predominance of resonance scattering or absorption and the relative locations of the resonances within the energy groups. The resonance overlap effect is discussed in more detail in Sect. M7.A.7.

3. Resonance interference between two different media is not correctly treated.

4. Fissile lumps in fissile solutions are not correctly treated.

5. The determination of parameters used in processing the cross sections are derived from the geometry of the cell description. CSAS does not allow multiple cell descriptions.

6. It is strongly recommended that CSAS calculations be made of benchmark experiments similar to the problem being calculated in order to demonstrate the validity of the code/cross-section combination for that type of problem.

CSAS uses cross sections created using data derived from the cell description specified in the Material Information Processor data. The cell description is not updated, and the cross sections are not reprocessed even when convergence is achieved. Therefore, it is strongly recommended that once convergence is achieved, the materials information processor cell be updated to use the converged configuration and the search be rerun to ensure that the cross sections were adequate. The new search should converge immediately if the initial cell was adequate for the cross-section treatment. Otherwise, the search will continue until convergence is achieved.

## C4.2.4 REFERENCES


1. L. W. Nordheim, "The Theory of Resonance Absorption," *Proceedings of Symposia in Applied Mathematics*, Vol XI, p. 58, G. Birkhoff and E. P. Wigner, Eds., Am. Math. Soc. (1961).

2. S. M. Dancoff and M. Ginsburg, *Surface Resonance Absorption in a Close-Packed Lattice*, CP-2157, Clinton Plant, U.S. Atomic Energy Commission (September 1944).

## C4.3 LOGICAL PROGRAM FLOW

The general flow of Criticality Safety Analysis Sequence No. 4 is given in this section. The program flow is divided into several sections. An abbreviated representation, rather than a formal flowchart, is used to outline the program flow of each section. A flowchart may show a subroutine or library routine only once, even though it may be called multiple times in that portion of the program. The corresponding text usually states when multiple calls are made to a routine. Accompanying each abbreviated flowchart is an explanation of the purpose of that segment of the program and text describing each of the referenced subroutines.

Criticality Safety Analysis Sequence No. 4 has been written in a general form to allow maximum flexibility. Data are transferred from one module to another via sequential data files and direct-access data files. Table C4.3.1 lists the data files utilized by this program. CSAS is used to represent the control module defining the available sequences. Data files utilized by the individual functional modules are included in Table C4.3.2.

The flowchart representation for Criticality Safety Analysis Sequence No. 4 is defined as follows:

> The SCALE DRIVER is contained in a diamond, subroutines initiating control modules are contained in circles, functional modules are contained in hexagons, major program segments are contained in rectangularized ovals, subroutine names are contained in boxes, and library routines are represented as bare names. An arrow in the flowchart indicates that the subroutine or module associated with the arrow will be discussed later in the section.

### C4.3.1 INITIATION OF CSAS (CRITICALITY SAFETY ANALYSIS SEQUENCE NO. 4)

This portion of the program is the initiation of the criticality safety analysis sequence. The SCALE DRIVER, described in Sect. M1, obtains the name of the module to be accessed from the first card (module specification card) in the user input data file. The control modules available in Criticality Safety Analysis Sequence No. 4 are shown in Fig. C4.3.1. These control modules, with the exception of MODIFY, use the Material Information Processor data as described in Sect. M7.3.

SCALE DRIVER allows implementation of a modular code system which includes Criticality Safety Analysis Sequence No. 4. See Sect. M1 for additional information about the driver.

CSAS4 performs a search for three-dimensional (3-D) problems that do not need a cell-weighted mixture cross section. The search is conducted by sequentially activating the functional modules BONAMI, NITAWL-II, and KENO V.a to process the required cross sections and calculate the k-effective. The control module MODIFY then performs operations to alter the pitch or other geometric dimensions repeatedly and to activate KENO V.a until the search requirements are met or the job is terminated for other reasons.

CSAS4X performs a search for 3-D problems that require a cell-weighted mixture cross section. The search is achieved by sequentially activating the functional modules BONAMI, NITAWL-II, XSDRNPM-S, and KENO V.a to process the required cross sections and calculate the k-effective. The control module MODIFY then performs operations to alter the pitch or other geometric dimensions repeatedly and to activate KENO V.a until the search requirements are met or the job is terminated for other reasons.

Table C4.3.1 Description of units utilized by the Criticality Safety Analysis Sequence No. 4 control module

| Unit No. | Type of data | Creating module | User module | Type of file |
|---|---|---|---|---|
| 5 | Input | DRIVER | CSAS | Sequential |
| 8 | Scratch file | CSAS | CSAS | Direct or random-access data |
| 11 | Short master cross-section library | CSAS | BONAMI | Sequential |
| 16 | Scratch file | CSAS | CSAS | Sequential |
| 17 | Scratch file | CSAS | CSAS | Sequential |
| 18 | Scratch file | CSAS | CSAS | Sequential |
| 19 | Scratch file | CSAS | CSAS | Sequential |
| 70 | User-supplied cross sections | | CSAS | AMPX master format library |
| 81 | 16-group cross sections | | CSAS | AMPX master format library |
| 82 | 27-group cross sections | | CSAS | AMPX master format library |
| 83 | 44-group cross sections | | CSAS | AMPX master format library |
| 84 | 238-group cross sections | | CSAS | AMPX master format library |
| 85 | 22 neutron-18 gamma cross sections | | CSAS | AMPX master format library |
| 86 | 18-gamma cross sections | | CSAS | AMPX master format library |
| 87 | 27-group cross sections for burnup | | CSAS | AMPX master format library |
| 88 | 27 neutron-18 gamma cross sections | | CSAS | AMPX master format library |
| 89 | Standard Composition Library | | CSAS | Direct or random-access data file |
| 90 | Search data | CSAS | MODIFY | Direct or random-access data |
| 92 | ICE input data | CSAS | ICE | Sequential |
| 95 | KENO V.a restart data | CSAS MODIFY KENO V.a | KENO V.a MODIFY | Sequential |
| 96 | BONAMI input data | CSAS | BONAMI | Sequential |
| 97 | NITAWL-II input data | CSAS | NITAWL-II | Sequential |
| 98 | XSDRN input data | CSAS | XSDRNPM | Sequential |
| 99 | Printed output | CSAS MODIFY | CSAS MODIFY | Sequential |

Table C4.3.2 Description of units used by the functional modules of CSAS

| Unit No. | Module | Action | Type of file |
|---|---|---|---|
| 1 | BONAMI | WRITE | Microscopic master working format cross sections |
| | NITAWL-II | READ | Microscopic master working format cross sections |
| 2 | ICE | WRITE | Macroscopic working format cross sections |
| 3 | XSDRNPM | WRITE | Microscopic cell-weighted working format cross sections |
| | ICE | READ | Microscopic cell-weighted working format cross sections |
| | KENO V.a | READ | Microscopic cell-weighted working format cross sections |
| 4 | NITAWL-II | WRITE | Microscopic working format cross sections |
| | ICE | READ | Microscopic working format cross sections |
| | KENO V.a | READ | Microscopic working format cross sections |
| | XSDRNPM | READ | Microscopic working format cross sections |
| 6 | BONAMI | WRITE | Printed output |
| | ICE | WRITE | Printed output |
| | KENO V.a | WRITE | Printed output |
| | NITAWL-II | WRITE | Printed output |
| | XSDRNPM | WRITE | Printed output |
| 8 | BONAMI | WRITE/READ | Direct or random-access scratch files |
| | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 9 | BONAMI | WRITE/READ | Direct or random-access scratch files |
| | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct or random-access scratch files |
| | NITAWL-II | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 10 | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 11 | BONAMI | READ | Short AMPX master format cross sections |
| 14 | ICE | WRITE | Monte Carlo-formatted macroscopic cross sections |
| | KENO V.a | READ/WRITE | Monte Carlo-formatted macroscopic cross sections |
| 16 | KENO V.a | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Angular flux file |
| 17 | XSDRNPM | WRITE/READ | Sequential scratch file |
| 18 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 19 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 79 | KENO V.a | READ | KENO V.a albedo file |
| 80 | KENO V.a | READ | KENO V.a weighting file |
| 92 | ICE | READ | Binary input file |
| 95 | KENO V.a | READ | Binary input file |
| 96 | BONAMI | READ | Binary input file |
| 97 | NITAWL-II | READ | Binary input file |
| 98 | XSDRNPM | READ | Binary input file |

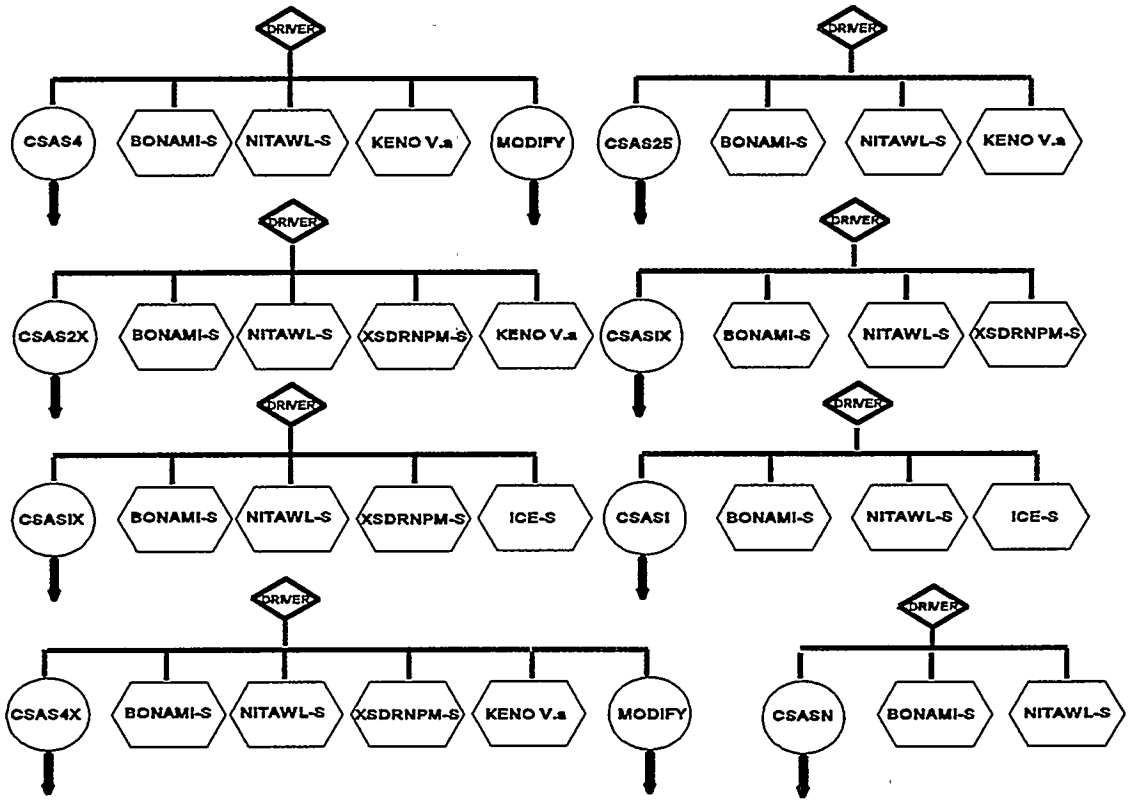



Figure C4.3.1 Control modules available in Criticality Safety Analysis Sequence No. 4

CSAS25 calculates the k-effective for 3-D problems. This module sequentially activates the functional modules BONAMI, NITAWL-II, and KENO V.a to process the required cross sections and calculate the k-effective. Note that a cell-weighted mixture cross section cannot be used.

CSAS2X calculates the k-effective for 3-D problems that require a cell-weighted mixture. It is the same as CSAS25 except a cell-weighted mixture cross section is always created. This control module sequentially activates the functional modules BONAMI, NITAWL-II, XSDRNPM-S, and KENO V.a to process the required cross sections and calculate the k-effective.

CSAS1X creates a microscopic cell-weighted library in the AMPX working library format. This control module sequentially activates the functional modules BONAMI and NITAWL-II to process the necessary cross sections and calculates the k-effective of the one-dimensional (1-D) problem using the discrete-ordinates code XSDRNPM-S.

CSASN creates a resonance-corrected microscopic cross-section library in the AMPX working library format by sequentially activating the functional modules BONAMI and NITAWL-II.

CSASI provides a resonance-corrected macroscopic mixture cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, and ICE.

CSASIX provides a resonance-corrected, cell-weighted, macroscopic mixture cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, XSDRNPM-S, and ICE. The discrete ordinates code XSDRNPM-S also computes the k-effective for the 1-D problem.

BONAMI is a functional module in the SCALE system that performs resonance self-shielding calculations for nuclides that contain Bondarenko data in their cross sections. See Sect. F1 for detailed information concerning BONAMI.

NITAWL-II is a functional module in the SCALE system. It applies a Nordheim resonance self-shielding correction to nuclides having resonance parameters associated with the cross-section data. See Sect. F2 for detailed information.

XSDRNPM-S is a functional module in the SCALE system. It provides cell-weighted cross sections based on the specified unit cell. It can also calculate the k-effective for 1-D problems. See Sect. F3 for additional information.

ICE is a functional module in the SCALE system. It creates a Monte Carlo-formatted mixed cross-section library for use by KENO V.a or MORSE. See Sect. F8 for additional details.

KENO V.a is a functional module in the SCALE system. It calculates the effective neutron multiplication of a 3-D problem. See Sect. F11 for a detailed description of KENO V.a.

MODIFY is a control module in the SCALE system. It is responsible for conducting a search by repeatedly altering the geometry specifications and executing KENO V.a. Section C4.3.6 contains more detail about MODIFY.

### C4.3.2 OVERALL FLOW DIAGRAM

Figure C4.3.2 illustrates the overall flow of Criticality Safety Analysis Sequence No. 4. The control modules CSAS4, CSAS4X, CSAS25, CSAS2X, and CSAS1X communicate with the driver and dictate the order in which other modules are to be executed.

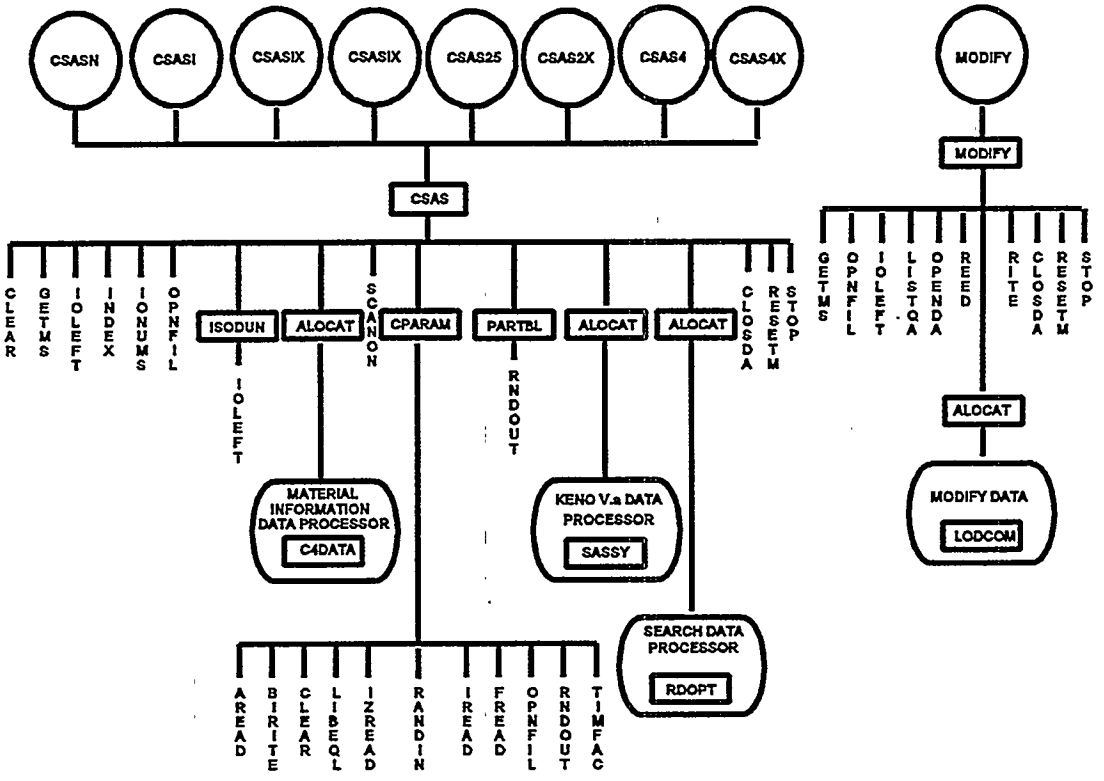



Figure C4.3.2 Overall flow of Criticality Safety Analysis Sequence No. 4

CSAS4 - This control module is responsible for conducting a search for 3-D problems that do not need a cell-weighted cross section.

CSAS4X - This control module is responsible for conducting a search for 3-D problems that require a cell-weighted cross section.

CSAS25 - This control module calculates k-effective for a 3-D system, using KENO V.a, when a cell-weighted mixture cross section is not needed.

CSAS2X - This control module calculates k-effective for a 3-D system, using KENO V.a, when a cell-weighted mixture cross section is required.

CSAS1X - This control module is responsible for performing a 1-D criticality calculation and providing a resonance-corrected, cell-weighted, microscopic cross-section library in the AMPX working library format.

CSASI - This control module generates both a resonance-corrected, microscopic, cross-section library in the AMPX working library format and a resonance-corrected, macroscopic mixture

cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, and ICE.

CSASIX - This control module generates both a resonance-corrected, cell-weighted, microscopic cross-section library in the AMPX working library format and a resonance-corrected, cell-weighted, macroscopic mixture cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, XSDRNPM-S, and ICE. The discrete-ordinates code XSDRNPM-S also provides the k-effective for the 1-D problem.

CSASN - This control module generates a resonance-corrected, microscopic cross-section library in the AMPX working library format. This is accomplished by sequentially activating the functional modules BONAMI and NITAWL-II.

CSAS - This is the main program for CSAS and its associated control modules. GETMS is called to read the analytical sequence specification and the parameters associated with it. The sequence specification is used to set flags to cause the desired functional modules to be executed in the proper order. IOLEFT is called to initialize the counter that keeps track of the number of I/Os that are done. INDEX is a FORTRAN-supplied intrinsic function used to determine if a variable matches a specified character string. IONUMS is called to set the input and output units for the free-form reading routines. OPNFIL is called to open the input and output units that are used by the control module. IOSDUN is called several times to determine the number of I/Os that have been used in various operations. C4DATA is called from ALOCAT with the requested storage allocation. This accesses the MATERIAL INFORMATION PROCESSOR. SCANON is called to activate the feature that allows scanning the input data for the word END. If KENO V.a is to be executed, CPARAM is called to read the KENO V.a title and to read and initialize the KENO V.a parameter data. SASSY is called from ALOCAT to read the KENO V.a data and write the information in a form that can be utilized by the CSAS module. If a search is to be made, RDOPT is called from ALOCAT to read and check the search data and transmit it to the CSAS module. If errors are encountered during reading, preparing, and checking the data, execution is terminated without activating any of the functional modules. The direct-access files are closed out when the problem is completed or terminated. This program makes use of error flags in an effort to continue reading and checking input data even after errors are found. However, some errors are severe enough to cause immediate termination.

CLEAR - This library routine is called to initialize the logical unit numbers used by CSAS.

GETMS - This assembly language library routine is called to read the analytical sequence specification and any associated parameter data. It establishes communication with the SCALE driver and enters data obtained from the SCALE driver into common. An example of an analytical sequence specification is given on the following page. This particular example specifies that the control module CSAS is to be executed. The associated parameters are entered following the word "PARM=," starting beyond column 10. In this example the parameters specify that the data are to be checked without executing the problem, and a maximum of 200,000 words is requested to complete this task.

```
=CSAS4    PARM= (CHECK,SIZE=200000).
```

IOLEFT - This library routine is called to store the initial I/O count in common for determining the number of I/Os used in certain parts of the program.

IONUMS - This library routine sets the I/O units or devices for the free-form data reading routines.

OPNFIL - This library routine is called to open the I/O units or devices used by the specified CSAS control module and all of the functional modules it invokes.

IOSDUN - This subroutine is called several times to indicate the number of I/Os used in various operations.

ALOCAT - This assembly language library routine is called with three arguments. The first argument is a subroutine name, the second argument is the maximum number of words of core storage to be allocated, and the third argument is the statement number to return to if the available core storage is less than that requested by the second argument. ALOCAT is used to call C4DATA, SASSY and, if a search is to be made, RDOPT. In each call an array name, the length of the array, and a statement number in a nonstandard return are passed to the subroutine.

MATERIAL INFORMATION PROCESSOR - This segment of the program is accessed by calling subroutine C4DATA via ALOCAT. It is responsible for reading the standard compositions specification data and associated engineering-type specifications used to generate the mixing table and in performing the cell-weighting calculations. It checks the data for obvious errors and generates information necessary for processing the cross sections. It continues checking and preparing data for use in the functional modules activated by the sequence. The MATERIAL INFORMATION PROCESSOR is described in more detail in Sect. M7.3.

SCANON - This library routine is called to activate the feature that allows scanning for the word END during data reading.

CPARAM - This subroutine is called only if KENO V.a is to be executed. It is responsible for reading the KENO V.a title, initializing certain default values for the KENO V.a parameter data, reading the KENO V.a parameter data, and writing the parameter data on the restart data file to communicate it to the KENO V.a functional module. AREAD, IREAD, FREAD and ZREAD are used to read the parameter data. BIRITE is used to write the parameter data on the restart data file, which is used by the functional module KENO V.a to define the problem. CLEAR is called to zero a common. LIBEQL determines the unit from which cross sections are to be read. RNDIN is called to load the random number if it is entered as parameter data. OPNFIL opens the I/O units that will be used, RNDOUT stores the current random number, and TIMFAC adjusts the time estimate.

AREAD - This library routine may be called many times from subroutine PARAM. It is used to read parameter names and alphanumeric parameter data.

BIRITE - This library routine is used to write the binary data used by the functional modules.

CLEAR - This library routine is called to zero common DIMEN.

LIBEQL - This subroutine is called to determine which unit or device will be used to read the resonance-corrected microscopic library.

ZREAD - This library routine can be called from subroutine PARAM to read a hexadecimal random number that will be used as a kernel for the random number package.

RNDIN - This library routine is called from subroutine PARAM to transfer the random number read by ZREAD to the random number package. It is called only if a random number is entered as parameter data.

IREAD - This library routine can be called many times from subroutine PARAM. It is used to read integer parameter data.

FREAD - This library routine can be called many times from subroutine PARAM. It is used to read floating-point parameter data.

OPNFIL - This library routine opens the I/O units or devices that will be utilized in the problem.

RNDOUT - This library routine is used to preserve the current random number so it can be written on the restart data file.

TIMFAC - This library routine provides the proper adjustment factor by which the allotted time is multiplied to adjust the execution time for execution for different computers.

PARTBL - This subroutine is called only if KENO V.a is to be executed. It is responsible for printing the KENO V.a numeric and logical parameter tables. RNDOUT is called to preserve the current random number so that it can be printed in the numeric parameter table.

KENO V.a DATA PROCESSOR - This segment of the program is accessed by calling subroutine SASSY via ALOCAT. It reads and checks the remainder of the KENO V.a data used by CSAS. It writes the data on the restart data file to be used by the functional module KENO V.a. This segment of the program is described in detail in Sect. C4.3.4.

SEARCH DATA PROCESSOR - This segment of the program is accessed by calling subroutine RDOPT via ALOCAT. It reads the search data and writes it on a data set for use by the CSAS control module. It is described in more detail in Sect. C4.3.5.

CLOSDA - This library routine is called to close out each direct-access file when the problem is completed in a normal manner.

RESETM - This library routine is an entry point in the assembly language library routine GETMS. It transmits control data back to the SCALE driver.

STOP - This library routine is called several times to write error messages. It can be instructed to provide a traceback and can terminate execution or allow execution to continue, depending on the arguments provided to it.

MODIFY - This is the MAIN program for the search control module. GETMS establishes communication between the SCALE driver and the search control module, MODIFY. OPNFIL opens the input and output units for the search. IOLEFT is used to monitor the I/Os used by the code. LISTQA is called to print a program verification information table for quality assurance purposes. OPENDA opens the direct-access units; REED is used to load data from direct access. LODCOM is called via ALOCAT to process the search data and alter the appropriate data accordingly. RITE writes data on the direct-access unit. CLOSDA is called to close out the direct-access file. RESETM transmits control data back to the SCALE driver. STOP is called if additional core storage is required. If the problem has converged or become unsolvable, an appropriate message is written, and flags are set to terminate execution. Otherwise, a flag is set so that the driver will execute KENO V.a using the modified data.

ALOCAT - This assembly language library routine is used to call LODCOM with an array name, the length of the array, and a statement number for a nonstandard return.

MODIFY DATA - This segment of the program alters the data for the optimum pitch search. It is accessed by calling subroutine LODCOM. Section C4.3.6 contains a detailed description of this segment of the program.

### C4.3.3 THE MATERIAL INFORMATION PROCESSOR

The Material Information Processor is responsible for reading data that specifies the cross-section library, defines the materials to be used in the problem, and provides data used to apply geometric and resonance corrections to the cross sections. After the data are read and checked, calculations are performed to create a mixing table and provide data to the functional modules that are invoked by the control module to perform cross-section processing to prepare a problem-dependent cross-section library. Section M7.3 contains the program flow for the Material Information Processor.

### C4.3.4 KENO V.a DATA PROCESSOR

This portion of the program, as shown in Fig. C4.3.3, is responsible for processing the KENO V.a input data. All the KENO V.a data, except the title and parameters, are read in this portion of the program. The data are carefully checked for errors and are written on the restart data file to be used by the functional module KENO V.a.

SASSY - This subroutine controls the processing of the KENO V.a data. It calls CLEAR to zero the direct-access pointer and length arrays. IOSDUN is called to indicate the number of I/Os used in the different operations involved in processing the KENO V.a data. Then RITE is used to load the direct-access pointer and length arrays on the direct-access data file. Various subroutines are called to read and check the KENO V.a data. If an error is recognized in the data, STOP is called to write a message, and control is returned to the MAIN program. If the data are found to be error free, a restart data file is written by subroutine WRTRST before returning to MAIN.

IOSDUN - This subroutine is called several times to indicate the number of I/Os used for various operations.

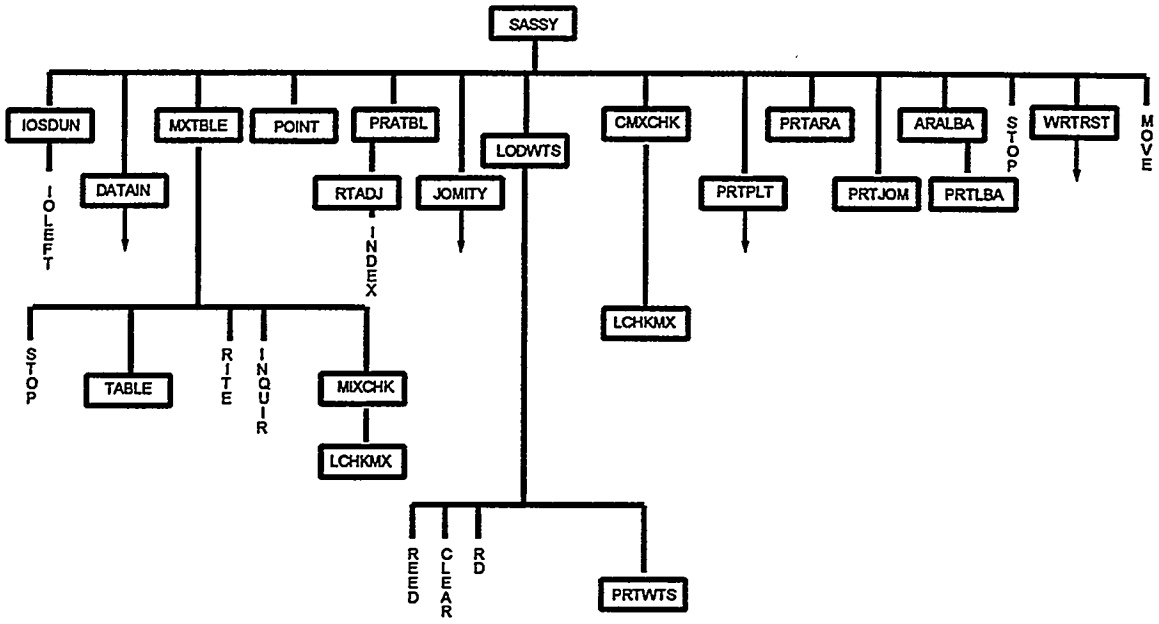



Figure C4.3.3 Flowchart of KENO V.a data processor

IOLEFT - This library routine stores the I/O count in common.

DATAIN - This subroutine controls the reading of all the KENO V.a input data except the title card and parameters. It is explained in more detail in Sect. C4.3.4.1.

MXTBLE - This subroutine initializes the threshold for printing cross-section processing error messages for invalid moments if it was not entered as data, sets up pointers for mixing table data for KENO V.a, calls STOP if more storage is required, and calls TABLE to read the mixing table data. RITE is used to write the mixing table data to a random-access file, and INQUIR is called to set the next block for a direct or random-access device. MIXCHK is called to ensure that the mixtures used in the geometry data were also specified in the Material Information Processor data.

TABLE - This subroutine is responsible for reading the mixing table data created by the Material Information Processor and storing it in the proper arrays for KENO V.a.

MIXCHK - This subroutine reads the KENO V.a mixing table data and writes an error message if a mixture utilized in the geometry data was not created by the Material Information Processor.

LCHKMX - This logical function is called by MIXCHK to return a value of true if a mixture specified on a KENO V.a geometry card is not specified in the mixing table and a value of false if it is.

POINT - This subroutine calculates pointers for storing and accessing KENO V.a data in memory.

PRATBL - This subroutine prints the KENO V.a table of additional data.

RTADJ - This subroutine right-adjusts the albedo boundary condition names that are read in as left-adjusted data. RTADJ utilizes the FORTRAN-supplied intrinsic function INDEX to locate the first blank character.

JOMITY - This subroutine is primarily responsible for generating additional geometry data, checking the geometry data, writing appropriate geometry error messages, and printing the geometry that is used in the problem. See Sect. C4.3.4.9 for additional information.

LODWTS - This subroutine is called only if PWT=YES is specified in the KENO V.a parameter data. LODWTS prints the bias IDs versus the material I/Os used in the problem. It also loads and prints the biasing or weighting data. The library routine REED is used to load the bias IDs from the direct-access device. CLEAR is used to zero the weighting arrays. RD is used to load the biasing data, and subroutine PRTWTS is called to print it.

PRTWTS - This subroutine prints the group-dependent weight average array for each biasing region in a compact fashion.

CMXCHK - This subroutine is used to ensure that a component of the cell-weighted mixture is not used in the KENO V.a geometry data. If it is, an error message is written, and an error flag is set.

LCHKMX - This logical function is called by CMXCHK to return a value of true if a mixture specified on a KENO V.a geometry card is not a component of the cell-weighted mixture and a value of false if it is.

PRTPLT - This subroutine is called to generate and print 2-D printer plots representing slices through the geometrical representation of the physical problem. See Sect. C4.3.4.14 for details.

PRTARA - This subroutine prints a table listing the array number, the number of units in the x, y, and z directions, and the nesting level for each array utilized in the problem.

PRTJOM - If a geometry error is recognized, this subroutine is called to print the KENO V.a geometry data that is used in the problem.

ARALBA - This subroutine calls PRTLBA for each unit orientation array used in the problem. Subroutine PRTLBA checks each array for geometrical consistency and prints the array data.

STOP - This library routine is called to print an error message and terminate if errors were found in the CSAS or KENO V.a data.

PRTLBA - If a geometry error is recognized, and more than one unit or box type is entered in the geometry data, this subroutine is called to print the KENO V.a unit orientation array data. STOP is called to write an error message if too many units are used in the KENO V.a geometry description.

WRTRST - This subroutine controls the writing of all the KENO V.a data on the restart data file for use by the KENO V.a functional module. MOVE is called to load the required common information into COMMON/REC1/, a common that is used by the analytical sequence. The array that contains the IDs of the 1-D cross sections is written on the restart data file. REED and IO are then used to write the geometry data, unit orientation data, biasing data, and start data on the restart data file.

MOVE - This library routine is called to load data into common.

### C4.3.4.1 Control of the Data Input for KENO V.a

This section of the program, shown in Fig. C4.3.4, is responsible for reading the input data that will be used by KENO V.a, with the exception of the title and parameter data.

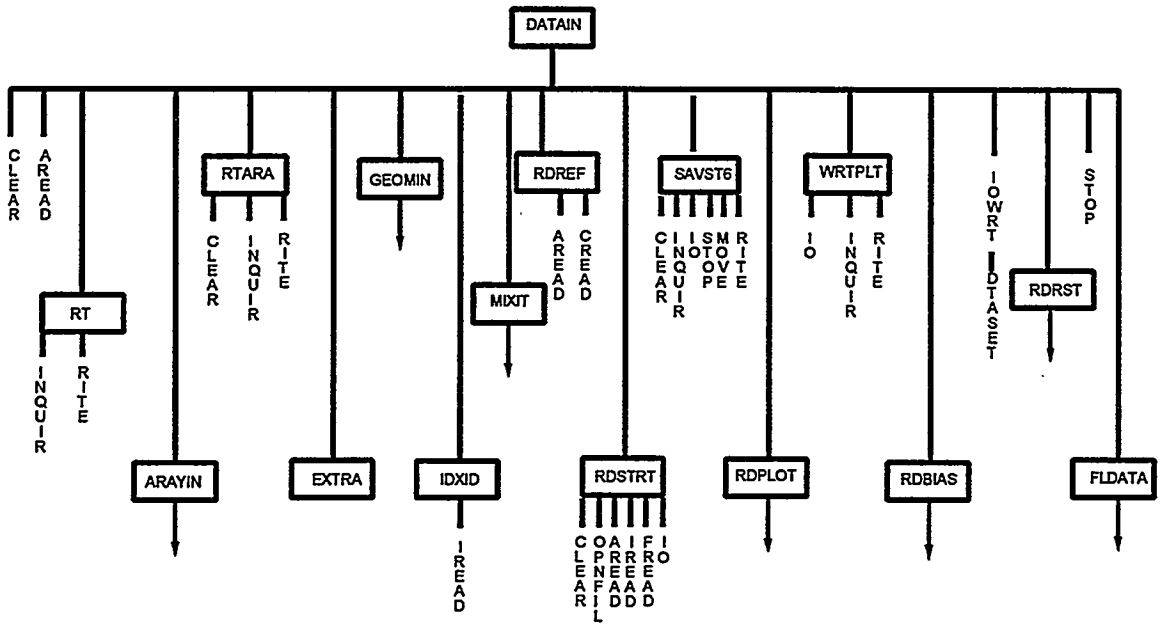



Figure C4.3.4 Flowchart of KENO V.a data reading routines

DATAIN - This subroutine controls the reading of all the KENO V.a data except the title card and parameter data. It initializes COMMON/STDATA/, the common that contains the start data, and initializes the array that contains the IDs of the 1-D cross sections that are to be utilized in the problem. The data reading is accomplished by reading blocks of data. Each block of data is preceded by a keyword that indicates the type of data being entered. After reading the keyword, the appropriate subroutine is called to read the accompanying data. After the data has been read, it is written on the direct-access data file. IOWRT is called several times to generate a table that lists the unit

numbers used by KENO V.a, their names, data set names, and the volume containing the data set. This table can be valuable for quality assurance applications. The library routine STOP is called to write error messages. Subroutine FLDATA is then called to supply information that was not entered as data.

CLEAR - This library routine is called to initialize COMMON/STDATA/. If biasing data are entered, it is also called to clear the space that will contain the biasing data. CLEAR is called with two arguments, a beginning location and a length. It initializes all included locations to zero.

AREAD - This library routine is used to read the READ flag and the keyword defining the type of associated data as well as the END flag and the associated keyword. It can be called many times from DATAIN.

RT - This subroutine may be called many times to write data on the direct-access data file. It passes information between DATAIN and RITE. INQUIR is called to return the value of the next direct-access record after the geometry data and/or the extra 1-D cross sections are written on the direct-access device.

RITE - This library routine is called from RT to write an array of data on the direct-access data file.

ARAYIN - This subroutine is called to read data defining the array size. It also reads the unit orientation data if it is entered. ARAYIN is not called for a single unit problem. See Sect. C4.3.4.2 for a more complete description.

RTARA - This subroutine is called only if ARAYIN is called. CLEAR is called to zero the array before RTARA reads the array data from the scratch unit and writes them on the direct-access device. INQUIR is called to return the value of the next direct-access record after the array data are written on the direct-access device.

EXTRA - This dummy subroutine is provided to allow the user to input extra data that is not normally processed by KENO V.a. The user must provide the programming to read and utilize the data.

GEOMIN - This subroutine is called to read the geometry region data. See Sect. C4.3.4.3 for additional information. The geometry region data are described in detail in Sect. F11.4.4.

IDX1D - This subroutine is called if the number of extra 1-D cross sections is greater than zero and extra 1-D data are entered. It reads the extra 1-D IDs and loads them into the MT array. The data reading is accomplished using the library routine IREAD.

MIXIT - This subroutine is called to read the mixing table data block that defines the mixtures that are to be created. Section C4.3.4.4 explains the mixing procedure in more detail.

RDREF - This subroutine is called to read the boundary conditions (or albedo options) that are to be applied at the outer boundaries of the system described by the KENO V.a geometry data and unit orientation data. The boundary condition data are read using the library routine AREAD. The library routine CREAD is used to read the albedo names. Some preliminary data checks are made

to detect invalid face code names and incompatible boundary conditions. Section F11.4.6 describes the input data for defining the KENO V.a boundary conditions.

RDSTRT - This subroutine is called to read start data for KENO V.a. This data are used to define the spatial distribution of the initial generation. The library routine CLEAR is called to zero the array that will contain the start data. The library routine AREAD is used to read the keywords associated with the start data. IREAD and FREAD are used to read the integer and floating-point start data, respectively. The library routine IO is used to write the start data associated with start type 6 on the scratch data file. The library routine INQUIR returns the value of the next direct-access record after writing the start data on the direct-access device. Data input for defining the initial source distribution is described in Sect. F11.4.8.

SAVST6 - This subroutine is called to save the data associated with start type 6. SAVST6 is called only if start type 6 is specified in the start type data. See Sect. F11.5.8 for information concerning the specification of start data. The library routine CLEAR is called to initialize the arrays that will contain the start data. The library routine IO is used to read the start data array from the scratch data file and load it into memory. The library routine MOVE is used to move the start data array into the neutron bank. The library routine RITE is called to write the neutron bank data on the direct-access data file. The library routine INQUIR returns the value of the next direct-access record after writing the start type 6 data on the direct-access device.

RDPLOT - This subroutine is called to read the plot data block that is used to generate printer plots. Section C4.3.4.5 explains the processing of the plot data in more detail. Section F11.4.11 describes the plot input data in detail.

WRTPLT - This subroutine reads the plot data block from the scratch data file and loads it on direct access. IO reads the data from the scratch data file and loads it in memory. RITE writes the data on the direct-access data file. INQUIR returns the value of the next direct-access record.

RDBIAS - This subroutine is called to read the biasing or weighting data to be used in the problem. See Sect. C4.3.4.6 for more information. The biasing input data block is described in detail in Sect. F11.4.7.

IOWRT - This subroutine is called with five arguments. In sequence they are (1) the output unit, (2) a four-character hollerith name representing a unit name, (3) the unit number represented by the second argument, (4) the number of words of hollerith information contained in the fifth argument, and (5) hollerith information to be printed. IOWRT is called several times to generate a table of the unit numbers, their names, the data set names, and the volumes on which each resides.

DTASET - This library routine is called from IOWRT to provide the data set name of the requested I/O unit and the volume on which it resides.

RDRST - After the data reading is complete, this subroutine is called if a unit containing data for restarting the problem has been defined. It loads data from the restart data file as described in Sect. C4.3.4.7.

FLDATA - This subroutine is called to supply default data for arrays that were not entered as input. Section C4.3.4.8 contains a more detailed account of the exact procedure.

### C4.3.4.2. Read KENO V.a Array Data

This section explains the procedure involved in reading the KENO V.a array data utilized by the analytical sequence, as shown in Fig. C4.3.5.

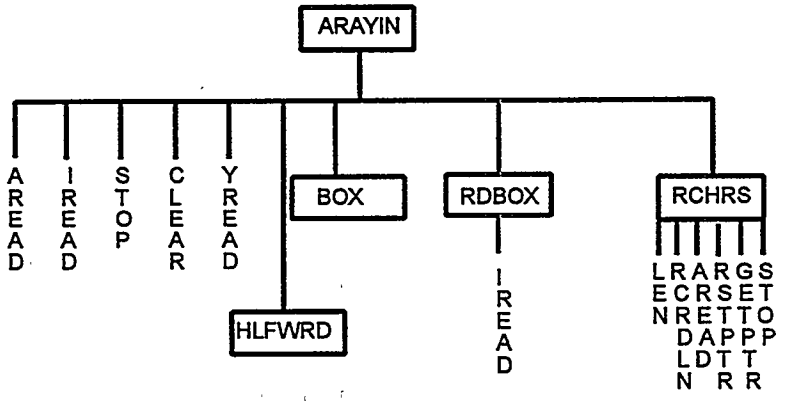



Figure C4.3.5 Flowchart for reading KENO V.a array data

ARAYIN - This subroutine is called from DATAIN when the words READ ARRA are encountered. It is responsible for reading the data that define the size of each unit orientation array. It reads the unit orientation data for each array using subroutine HLFWRD for the FILL option and subroutine RDBOX for the loop option. BOX then writes the array data on the scratch data file. CLEAR is called to zero the unit orientation array. The data are read using the library routines AREAD, IREAD, and YREAD. The data read by this subroutine is described in Sect. F11.4.5.

AREAD - This library routine is used to read the keywords associated with the array data.

IREAD - This library routine is used to read the integer data associated with the array data.

STOP - This library routine is called to write an error message and stop if insufficient memory is available to accommodate the unit orientation array.

CLEAR - This library routine is used to zero arrays.

YREAD - This library routine is called to read the unit orientation data for the FILL option.

HLFWRD - This subroutine is called only if the FILL option is used for entering the unit orientation data. The purpose of HLFWRD is to convert the unit orientation array data from full-word integers to half-word integers.

BOX - This subroutine is called to write the array data on the scratch data file.

RDBOX - This subroutine is called only if the LOOP option is used for entering the unit orientation data. It uses the library routine IREAD to read the unit orientation data.
Some data consistency checks are made, and appropriate error messages are written if errors are encountered. If the input geometry is to be printed, RDBOX prints the unit orientation for each array.

RCHRS - This subroutine is used to read the comment associated with an array. LEN is a FORTRAN-supplied function that sets the length of the data string. GETPTR is used to return the current pointer in the input buffer. RSTPTR resets the pointer. AREAD is used to read the input data, and RCRDLN sets the length of the input buffer. STOP is called to write an error message and stop if the array comment is too long (i.e., the ending delimiter is missing).

### C4.3.4.3. Read KENO V.a Geometry Data

This section of the program reads the KENO V.a geometry data required for the problem as shown in Fig. C4.3.6.

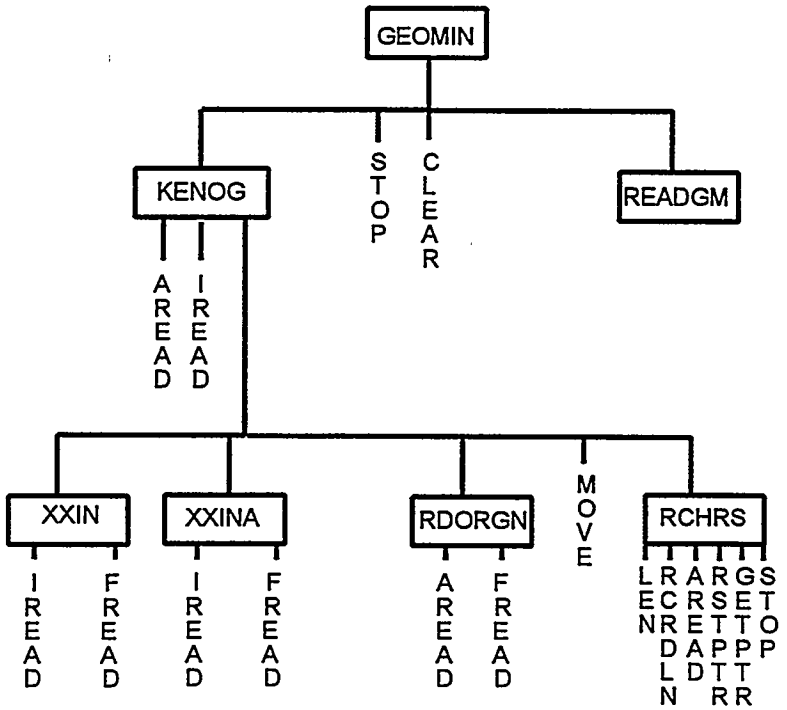



Figure C4.3.6 Flowchart of routines to read the KENO V.a geometry data

GEOMIN - This subroutine controls the reading of the KENO V.a geometry data. KENOG is called to read the data and write it on the scratch data file. Pointers are then calculated. STOP is called to print a message and terminate if more computer storage is required. CLEAR is called to zero the arrays that hold the geometry data. READGM is called to read the data from the scratch data file and load it into the appropriate arrays. STOP is called to write an error message if more storage is needed. Section F11.4.4 describes the KENO V.a geometry input data in detail.

KENOG - This subroutine uses AREAD to read the geometry word. IREAD is called to read the unit number for a unit or box type. It is also used to read the number of reflector regions on a reflector card. XXIN is called to read the mixture number, the bias ID number, and the geometry dimensions. XXINA is an entry point in XXIN. It reads the mixture number and geometry dimensions. The origin specification, if any, associated with a sphere, hemisphere, cylinder, or hemicylinder is read by RDORGN. The chord specification, if any, associated with a hemisphere or hemicylinder is also read by RDORGN. The necessary geometry data block is written on the scratch data unit. If the input geometry data block is to be printed, KENOG does this as the data are read.

XXIN - This subroutine is called to read the mixture number, the bias ID number, and the geometry dimensions. IREAD is used to read the mixture number and bias ID number. FREAD is used to read the geometry region dimensions.

XXINA - XXINA is an entry point in XXIN. It is called when the geometry word ARRA or ARRAY is encountered. It reads the mixture number and the geometry dimensions. IREAD is used to read the mixture number. FREAD is used to read the geometry region dimensions.

RDORGN - This subroutine uses AREAD to read a word. If the word is ORIGIN, it uses FREAD to read the points defining the origin. If the word is CHORD, it uses FREAD to read the offset of the plane with respect to the origin. If the word is not ORIGIN or CHORD, a flag is set to prevent KENOG from attempting to read another geometry word.

RCHRS - This subroutine is used to read the comment associated with a unit in the geometry region data. LEN is a FORTRAN-supplied function that sets the length of the data string. GETPTR is used to return the current pointer in the input buffer. RSTPTR resets the pointer. AREAD is used to read the input data, and RCRDLN sets the length of the input buffer. STOP is called to write a message and terminate the problem if the comment is too long.

READGM - This subroutine reads the geometry data from the scratch data unit and loads them into the proper arrays.

### C4.3.4.4. Read KENO V.a Mixing Table Data

This section deals with reading the mixing table data, as shown in Fig. C4.3.7.

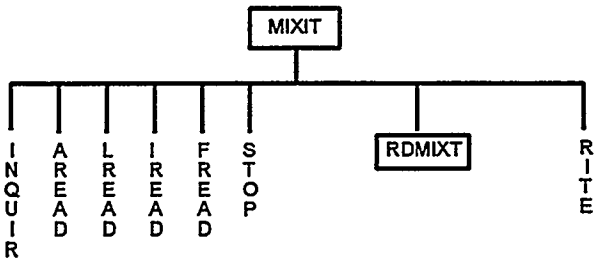



Figure C4.3.7 Flowchart for reading KENO V.a mixing table data

MIXIT - This subroutine uses AREAD to read the mixture keywords and the scattering keyword. IREAD is used to read the mixture numbers and the number of scattering angles as well as the nuclide IDs. FREAD reads the number densities. LREAD is used to determine if the next item is a keyword rather than a number. The necessary data arrays are written on the scratch data file. Pointers are calculated for the necessary storage arrays, and RDMIXT is called to load the data from the scratch file into the storage arrays.

The library routine RITE is called to write the mixing table information on the direct-access device, and INQUIR is called to return the value of the next direct-access record.

LREAD - This library routine returns a value of "true" if the next character is a numeric digit. Otherwise, a value of "false" is returned.

STOP - This library routine is called from MIXIT if the storage space is insufficient to hold the mixing table arrays.

RDMIXT - This subroutine reads the mixing table data arrays from the scratch data file and loads them into the storage arrays.

### C4.3.4.5. Read KENO V.a Plot Data

This section of the program, shown in Fig. C4.3.8, reads the plot or picture data used to generate printer plot maps of the mixtures, units, and/or bias IDs used in the problem. The plot input data block is described in Sect. F11.4.11.

RDPLOT - This subroutine uses CLEAR to initialize the data arrays. RCHRS is used to read the plot title and the character string of symbols to be used in the plot. AREAD, IREAD, and FREAD are used to read the plot or picture input data. SQRT is used to determine the normalization factor for the direction cosines, and IO is called to load the plot data on the scratch data file.

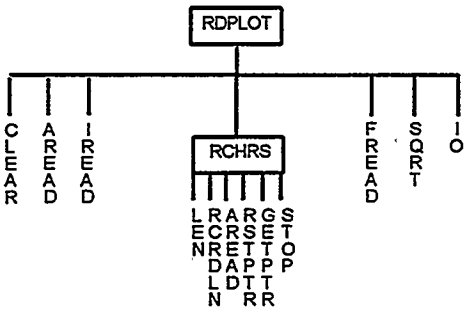



Figure C4.3.8 Flowchart for reading KENO V.a plot data

RCHRS - This subroutine is used to read the plot title and the character string that defines the symbols to be used in the plot. LEN is a FORTRAN-supplied routine that sets the length of the data string. GETPTR is used to return the current pointer in the input buffer. RSTPTR resets the pointer. AREAD is used to read the input data, and RCRDLN sets the length of the input buffer. STOP is called to write an error message if the plot title is too long and terminate the problem.

### C4.3.4.6. Read KENO V.a Biasing Data

This section of the program reads the biasing data used in the problem. The biasing input data block is described in Sect. F11.4.7 and illustrated in Fig. C4.3.9.

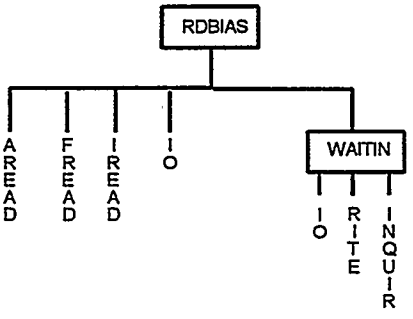



Figure C4.3.9 Flowchart for reading KENO V.a biasing data

RDBIAS - This subroutine is responsible for reading the biasing data block and writing it on the scratch data file. AREAD is used to read the keywords used in the biasing data and a title for the biasing material if the energy- and space-dependent values of the biasing function are entered from cards. IREAD and FREAD are used to read the numerical data. Pointers for the storage arrays needed

to process the biasing data are determined, and WAITIN is called to load the data from the scratch data file into the storage arrays and write them on the direct-access data file.

WAITIN - This subroutine reads the biasing data block from the scratch data file and loads it into the storage arrays. IO is used to load the energy- and space-dependent biasing function (*wtavg*) into the storage arrays. RITE is used to write the biasing data on the direct-access data file. The library routine INQUIR is called to set the next block on a direct- or random-access device.

### C4.3.4.7. Read KENO V.a Restart Data

This section of the program reads restart information from the restart data file. The flowchart for this procedure is given in Fig. C4.3.10.

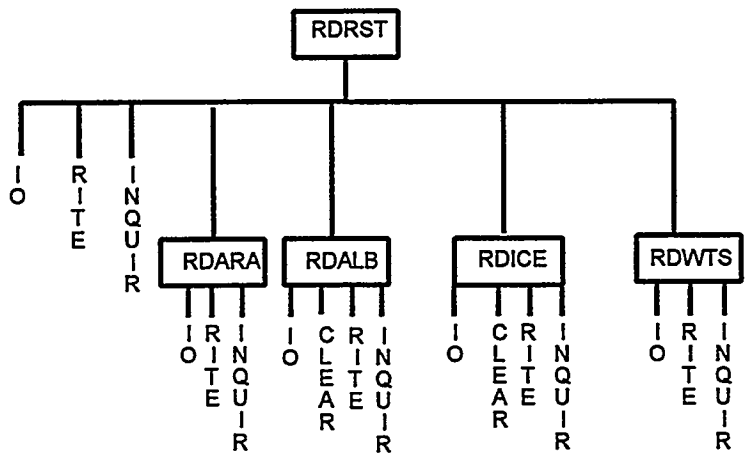



Figure C4.3.10 Flowchart for reading KENO V.a restart data

RDRST - This subroutine is called only if the problem is to use data from the restart data file. The program recognizes that restart data will be read if the restart unit is defined as a number greater than zero. IO is used to load the array that contains the 1-D IDs from the restart data file. Each type of data is loaded from the restart data file using IO and is written on the direct-access data file by RITE. All restart data except the mixed cross-section data, the differential albedo data, the array data, and the biasing data are processed directly in RDRST. RDICE is called to load the cross-section data on the direct-access data file, RDALB is called to load the differential albedo data on the direct-access data file, RDARA is called to load the array data on the direct-access data file, and RDWTS is called to load the biasing data on the direct-access data file. The library routine INQUIR is called to return the value of the next direct-access record.

RDALB - This subroutine is called from RDRST to read the albedo data block from the restart data file and write it on the direct-access data file. IO is used to load the albedo pointer and length arrays into

memory from the restart data file. CLEAR is called to zero the albedo pointer and length arrays, and RITE writes them on the direct-access data file. INQUIR stores the next direct-access block number. Each record of albedo data is read from the restart data file, loaded into memory using IO, and written on the direct-access data file using RITE. When all the records of albedo data have been processed, the updated pointer and length arrays are rewritten over the initial ones using RITE.

RDICE - This subroutine is called from RDRST to read the cross-section data block from the restart data file and write it on the direct-access data file. CLEAR is called to zero the pointer and length arrays. The length array is then read from the restart data file. RITE is used to write the pointer array and the length array on the direct-access data file. INQUIR stores the next direct-access block number. Then IO and RITE are used to load the cross-section data from the restart data file and write them on the direct-access data file. This procedure is repeated for every record of each mixture. The updated pointer and length arrays are then rewritten over the initial ones using RITE.

RDARA - This subroutine is called from RDRST to read the array data block from the restart data file and write it on the direct-access data file. IO is used to load the data from the restart data file, and RITE is used to write them on the direct-access data file. INQUIR stores the next direct-access block number.

RDWTS - This subroutine is called from RDRST to read the biasing data block from the restart data file and write it on the direct-access data file. IO is used to load the data from the restart data file, and RITE is used to write them on the direct-access data file. INQUIR stores the next direct-access block number.

### C4.3.4.8. Generate Remaining KENO V.a Data

This section of the program, shown in Fig. C4.3.11, is responsible for generating data that are required for a KENO V.a problem but are not entered directly as data.

FLDATA - The library routine REED is used to load data arrays from the beginning of the mixtures used in the geometry through the geometry data. The library routine STOP may be called to print an error message. SORTA is called to determine which arrays and holes are used as well as the array and hole nesting levels. RITE is called to write the geometry data on the direct-access data file. RGUSED is called to determine which geometry regions are used in the problem. SORTR is called to generate the mixture correspondence array. It is called again to generate the bias region correspondence array. These correspondence arrays are used to avoid storing mixture cross sections and biasing data that are entered as data but are not actually used in the problem. If boundary conditions specify differential albedo data, and they are not available from the restart data file, DIFALB is called to read the albedo data block and load the requested data on the direct-access file. If the requested biasing data were unavailable from the restart data file, WATES is called to load the energy and position-dependent biasing function (*wtavg* array) on the direct-access data file.

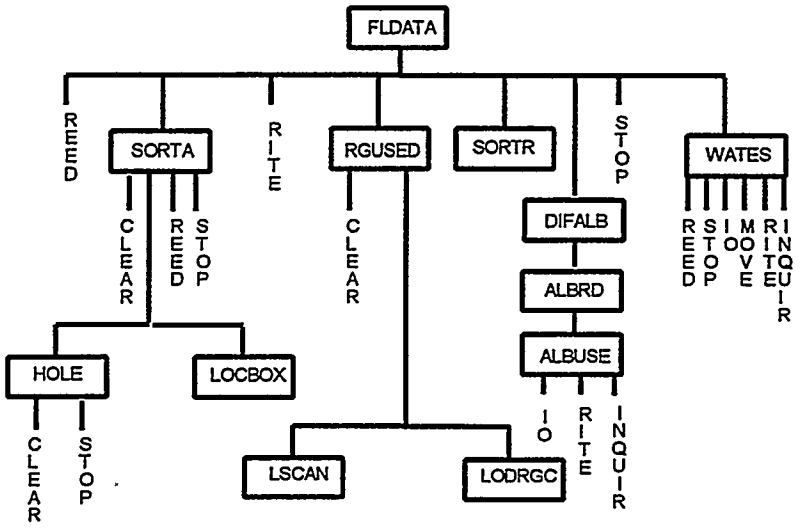



Figure C4.3.11 Flowchart for generating remaining KENO V.a data

SORTA - This subroutine checks to see that the global array is properly defined. It determines the array correspondence array and the nesting levels for holes and arrays. SORTA uses CLEAR to initialize arrays. REED is used to load the array data, and STOP is used to write an error message and terminate if more computer storage is needed for the problem.

HOLE - This subroutine is called from SORTA to determine what holes occur at the next nesting level and to adjust the array nesting level for arrays that occur in holes. It also checks to ensure holes are not recursively nested.

LOCBOX - This function subprogram is called from SORTA to determine the unit or box type at a given position in the unit orientation array.

RGUSED - This subroutine determines which geometry regions are used in the problem. The library routine CLEAR is called to zero the space for the region correspondence array. LSCAN determines if a particular unit or box type has been used in the unit orientation array, and if it has, LODRGC is called to load the region number into the region correspondence array.

LSCAN - This is a logical function that returns a value of true if the specified unit or box type is used in the unit orientation array. A value of false is returned if the unit or box type was not used in the unit orientation array.

LODRGC - This subroutine loads the region number in the region correspondence array.

SORTR - This subroutine is called twice from FLDATA to create a mixture correspondence array and a biasing correspondence array. These correspondence arrays are used to avoid storing mixture

cross sections and biasing information that are defined in the input data but were not referenced in the geometry data utilized in the problem. They are also used throughout the code for accessing the proper mixture cross sections and biasing information.

DIFALB - This subroutine is called if differential albedos are specified as a boundary condition but are not available from the restart data file. It rewinds the albedo data file, reads the header record, and calculates pointers. ALBRD is called to load the albedo data.

ALBRD - This subroutine searches through the albedo data file to locate the requested albedo name or boundary condition. If it is not found, an error message is written. If it is found, the number of different differential albedos that were requested are tabulated and ALBUSE is called.

ALBUSE - This subroutine writes the albedo pointer array on the direct-access data file. IO is used to load data from the albedo data file, and RITE is used to write the data on the direct-access data file. Then the pointer and length arrays are rewritten on the direct-access data file. INQUIR stores the next direct-access block number.

WATES - This subroutine reads the biasing input data block from the direct-access data file and reads the KENO V.a weights library. STOP is called if the computer storage space is too small to contain the energy- and position-dependent biasing function (*wtavg*). IO is used to load the biasing function into a temporary storage array. If a specific biasing function is to be used, MOVE is called to load it into the *wtavg* array. If biasing or weighting data are entered from cards, REED is used to load the data into a temporary storage array. If a specific biasing function that was loaded from cards is to be used in the problem, MOVE is called to load it into the *wtavg* array. When all the data have been processed, RITE is called to load the biasing input data on the direct-access data file. RITE is called again to load the *wtavg* array on the direct-access data file. INQUIR is used to store the next direct-access block number.

### C4.3.4.9. Process the KENO V.a Geometry Data

This portion of the program, shown in Fig. C4.3.12, is primarily responsible for loading the geometry data, generating additional geometry data, checking the geometry for consistency, writing error messages related to the geometry, and printing the geometry that is used in the problem.

JOMITY - This subroutine is responsible for generating additional geometry data, checking the geometry data, writing geometry error messages, and printing the geometry.

LOADIT - This subroutine loads the geometry data and the nonsupergrouped portion of the albedo data. Section C4.3.4.10 contains a more detailed description of the procedure.

IOSDUN - This subroutine returns the numbers of I/Os used. The library routine IOLEFT returns the numbers of I/O requests remaining before the system cancels the job.

CORSIZ - This subroutine sends the appropriate lattice or array information to ARASIZ for each lattice that is used in the problem. Using this information, it calculates the overall positive dimensions of the global array. The library routine, SQRT, is utilized to calculate the maximum chord length of an unreflected array, a reflected array, or a single unit problem.

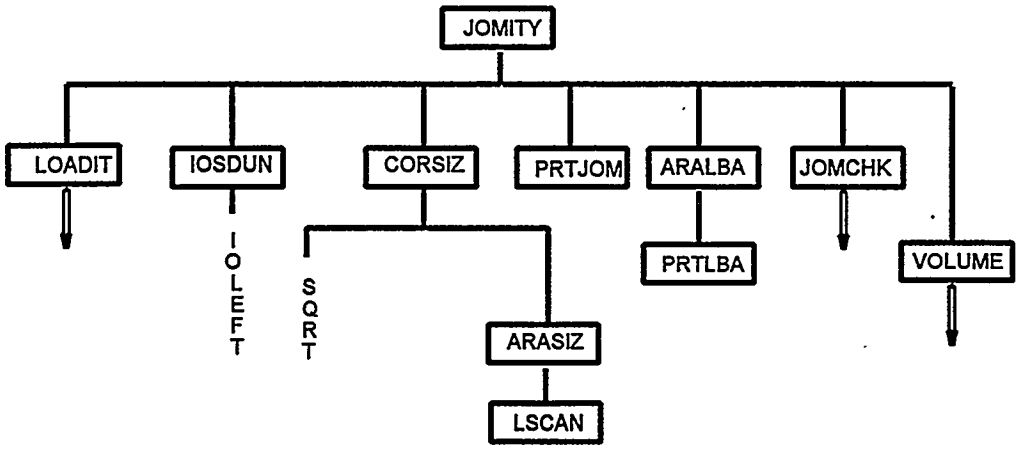



Figure C4.3.12 Flowchart for processing KENO V.a geometry data

ARASIZ - This subroutine uses the array unit orientation data to calculate the positive dimensions of the core boundary for that array or lattice. The function LSCAN is called to determine if a specified unit or box type has been used in the array. ARASIZ also checks to ensure that the faces of adjacent units are the same size and shape. Several error messages are written if errors are encountered.

LSCAN - This is a logical function that returns a value of true if the specified unit or box type is used in the unit orientation array. A value of false is returned if the unit or box type was not used in the unit orientation array.

PRTJOM - This subroutine prints the geometry data used in the problem.

ARALBA - This subroutine calls PRTLBA for each array that is used in the problem.

PRTLBA - This subroutine is called to print the unit orientation data for each lattice or array that is used in the problem. It may print a warning message associated with one or more lattices.

JOMCHK - The purpose of this subroutine is to perform consistency checks on the geometry data and write the appropriate error messages. See Sect. C4.3.4.11 for additional details.

VOLUME - This subroutine is responsible for calculating the volume of each geometry region and the cumulative volumes for each unit that is used in the problem. See Sect. C4.3.4.12 for additional details.

### C4.3.4.10. Load KENO V.a Data from the Direct-Access File

This portion of the program loads data from the direct-access data file into permanent memory as shown in Fig. C4.3.13.

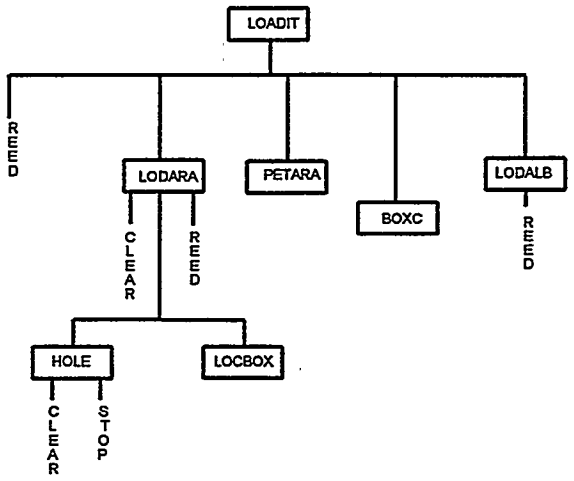



Figure C4.3.13 Flowchart for loading data from direct access

LOADIT - This subroutine calls the library routine REED to load the geometry data. If the problem is an array problem (lattice geometry), LODARA is called to load the lattices that are used in the problem and recompute and readjust the array nesting level array and hole nesting array. If multiple boxes are used in the problem, PRTARA is called to print the unit orientation array for each lattice used in the problem. BOXC is called to load the box correspondence array, and LODALB is called to load the nonsupergrouped portion of the albedo data.

LODARA - This subroutine is responsible for loading the lattices (unit orientation arrays) that are used in the problem, computing the hole nesting level array, and computing and adjusting the array nesting level array. CLEAR is used to initialize the arrays, and REED is used to load the unit orientation arrays. HOLE and LOCBOX are both called by LODARA.

HOLE - This subroutine is called from LODARA to determine which holes occur at the next nesting level and to adjust the array nesting level for arrays that occur in holes. It also checks to ensure that holes are not recursively nested. CLEAR is used for initialization purposes, and STOP is called if holes are recursively nested.

LOCBOX - This function is called from LODARA to return the unit or box type at a given position in the unit orientation array.

PRTARA - This subroutine prints a table of the arrays used in the problem. The array number, the number of units in the x, y, and z directions, and the nesting level are printed for each array.

BOXC - This subroutine uses the number of units or box types and the geometry region number corresponding to the first and last geometry region of each unit to generate the box correspondence array that contains the unit or box type number for each geometry region. This is loaded in the appropriate position as it is generated.

LODALB - This subroutine calls the library routine REED to load the pointer and length arrays for the albedo data from the direct-access data file. REED is used to load the nonsupergrouped albedo data, for each albedo that is used, into a temporary array. A loop over the number of angles is then used to load these data into the appropriate arrays. When all of the albedos used in the problem have been processed, REED is called to load the pointer arrays for the cross sections and albedos as well as the arrays defining the albedo to cross-section energy group correlation.

### C4.3.4.11. Check the KENO V.a Geometry Data

This portion of the program checks the KENO V.a geometry for inconsistencies, surface intersections, and other errors. The flowchart is shown in Fig. C4.3.14.

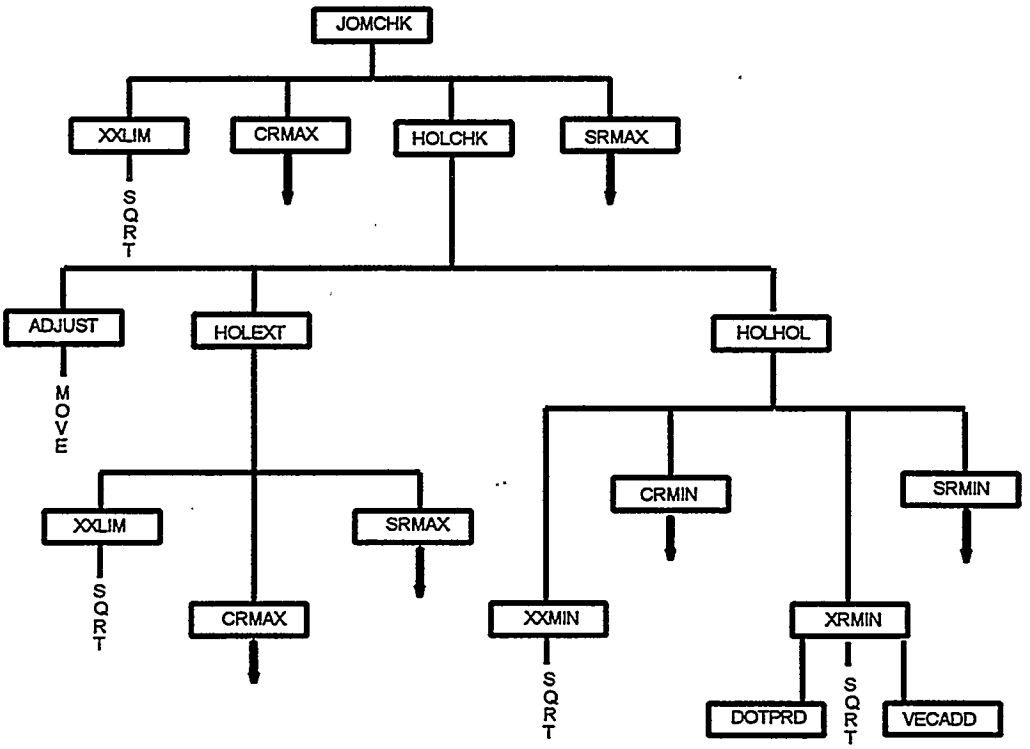



Figure C4.3.14 Flowchart of geometry-checking procedure

JOMCHK - This subroutine checks each geometry region to ensure that it does not intersect the next outer region. If an intersection occurs, an error flag is set and an error message is written. JOMCHK checks each surface of the outer region for an intersection. If the surface is a planar surface, XXLIM is used to return the farthermost point of the inner region in the direction of the plane specified in the call (i.e., for the -x face, XXLIM returns the most negative x value of the inner region). For a spherical surface, SRMAX is used to return the length of the maximum radius vector of the inner region with respect to the origin of the outer region. For a cylindrical surface, CRMAX is used to return the length of the maximum radius vector of the inner region with respect to the axis of the outer region. Subroutine HOLCHK is called to check for intersections between holes and other geometry regions and for holes with other holes.

CRMAX - This function determines the maximum cylindrical radius vector of a geometry region with respect to a given axis and then returns the magnitude of that vector. See Sect. C4.3.4.12 for additional details.

XXLIM - This routine returns the maximum coordinate of an interior region corresponding to a particular face direction (for negative face directions, maximum means most negative).

SRMAX - This function determines the maximum radius vector of a geometry region with respect to a given origin and then returns the magnitude of that vector. See Sect. C4.3.4.12 for additional details.

HOLCHK - This subroutine is responsible for ensuring that a hole does not intersect any other geometry region. ADJUST is called to adjust the dimensions of the hole with respect to the origin of the unit that contains the hole. HOLEXT is called to check for a hole intersecting the region external to it. HOLHOL is used to check for a hole intersecting the region internal to the region that contains the hole. It is also used for checking for the intersection of two holes.

ADJUST - This subroutine corrects the dimensions of a region represented as a hole with respect to the origin of the unit that contains the hole.

HOLEXT - This subroutine checks for a hole intersecting the region external to it. XXLIM is used to return the farthermost point of the hole in the direction of the plane specified in the call. CRMAX is used for a cylindrical surface to return the length of the maximum radius vector of the hole with respect to the axis of the outer region. SRMAX is used for a spherical surface to return the length of the maximum radius vector of the hole with respect to the outer region.

HOLHOL - This subroutine is responsible for ensuring that a hole does not intersect the geometry region internal to the region that contains the hole. It also checks for holes intersecting each other. XXMIN is used to return the nearest point of the hole in the direction of the plane specified in the call. CRMIN is used for a cylindrical surface to return the length of the minimum radius vector of the hole with respect to the axis of the surface. XRMIN determines the minimum coordinate of a hole with respect to a flat circular face. SRMIN is used for a spherical surface to return the length of the minimum radius vector of the hole with respect to the origin of the surface.

XXMIN - This function returns the minimum coordinate of the hole with respect to the fact of the cuboid specified in the call. SQRT is utilized in determining the minimum coordinate of spherical or cylindrical holes.

CRMIN - This function returns the magnitude of the minimum cylindrical radius vector of a hole with respect to a given axis. See Sect. C4.3.4.12 for additional details.

XRMIN - This function determines the minimum coordinate of a hole with respect to a flat circular face. SQRT is used in processing spherical and cylindrical holes. The functions DOTPRD and VECADD are called to perform vector arithmetic.

DOTPRD - This function returns the dot product of two vectors.

VECADD - This function returns the vector sum of two vectors.

SRMIN - This function returns the magnitude of the minimum radius vector from the center of a sphere to another geometry region. See Sect. C4.3.4.12 for additional details.

### C4.3.4.12. Determine Distances

This section of the KENO V.a geometry checking procedure is utilized in checking for geometry surface intersections. The flowcharts of the routines utilized in this procedure are given in Fig. C4.3.15.

CRMAX - This function determines the maximum cylindrical radius vector of a geometry region with respect to a given axis and then returns the magnitude of that vector. Depending on the geometry type, CRMAX may call CRSPRD to generate the cross production of two vectors, DOTPRD to generate the dot product of two vectors, VECADD to add two vectors together, and VECNRM to multiply a vector by a scalar. STOP is called if geometry inconsistencies are encountered.

SRMAX - This function determines the maximum radius vector of a geometry region with respect to a given origin and then returns the magnitude of that vector. Depending on the geometry type, SRMAX may call CRSPRD to take the cross production of two vectors, DOTPRD to take the dot product of two vectors, VECADD to add two vectors together, and VECNRM to multiply a vector by a scalar. STOP is called if geometry inconsistencies are encountered.

CRMIN - This function determines the minimum cylindrical radius vector of a geometry region with respect to a given axis and returns the magnitude of that vector. The geometry type determines which functions will be used to calculate the minimum radius vector. DOTPRD is used to generate the dot product of two vectors, CRSPRD generates the cross product of two vectors, VECNRM multiplies a vector by a scalar, and VECADD adds two vectors together. STOP is called if geometry inconsistencies are encountered.

SRMIN - This function determines the minimum radius vector of a geometry region with respect to a given origin (center of a sphere) and returns the magnitude of that vector. Depending on the geometry type, SRMIN may call DOTPRD to take the dot product of two vectors, CRSPRD to take the cross production of two vectors, VECNRM to multiply a vector by a scalar, and VECADD to add two vectors. STOP is called if geometry inconsistencies are encountered.

CRSPRD - This is a utility routine to generate the cross product of two vectors.

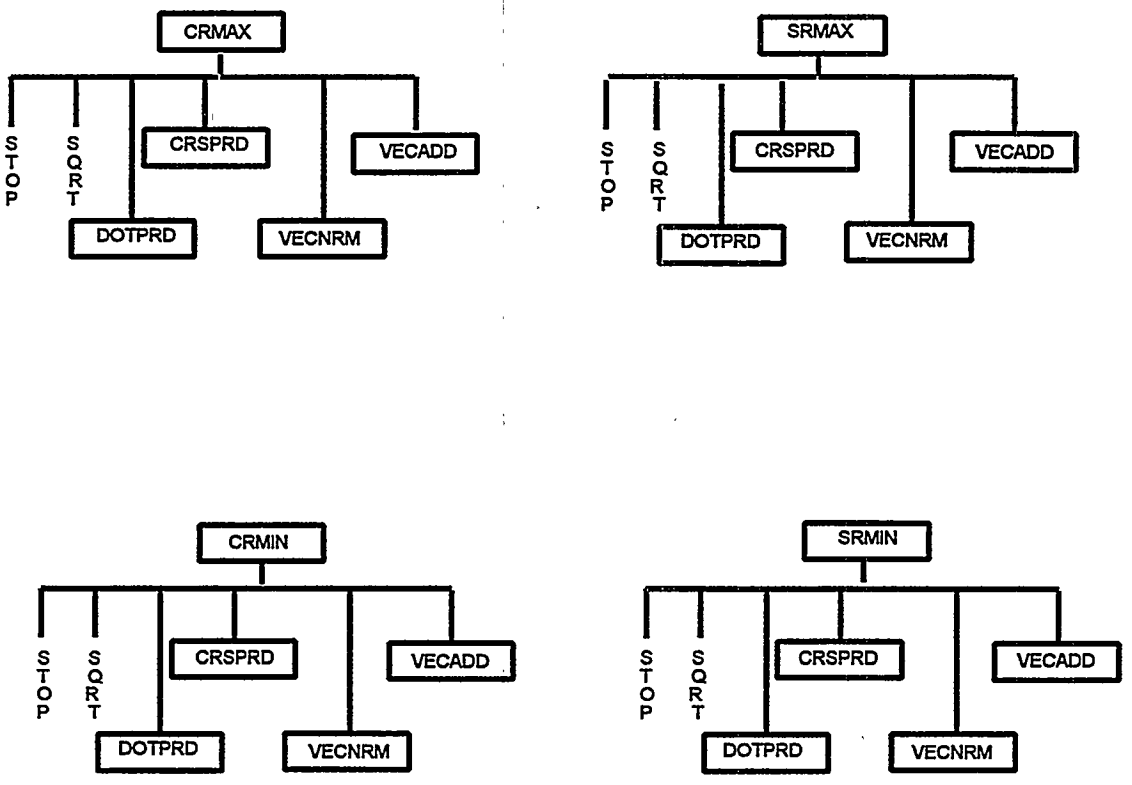



Figure C4.3.15 Flowchart for distance to intersection

DOTPRD - This function returns the dot product of two vectors.

VECADD - This subroutine returns the result of adding two vectors.

VECNRM - This routine scales a vector by a constant.

### C4.3.4.13. Calculate KENO V.a Geometry Volumes

This portion of the program, illustrated in Fig. C4.3.16, is responsible for calculating the volume of each KENO V.a geometry region used in the problem, the cumulative volumes for each unit used in the problem, the number of times each unit was used in the problem, and the total volume of each region summed over all occurrences.

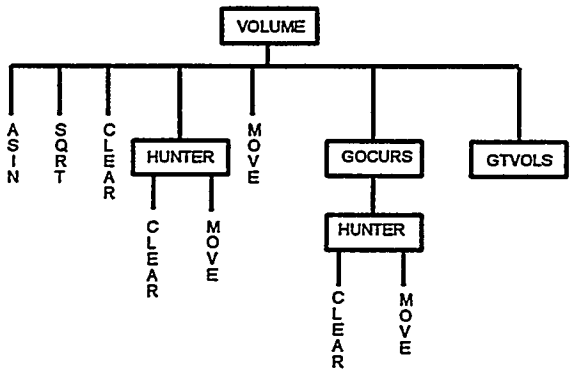



Figure C4.3.16 Flowchart for calculating KENO V.a geometry volumes

VOLUME - This subroutine calculates the volume of each region for every unit that is used in the problem. It then calculates the cumulative volumes for each unit. The intrinsic functions ASIN, which returns the arc sign of a variable, and SQRT, which returns the square root of an expression, are used when calculating the volume of a hemicylinder. CLEAR is used to initialize arrays. If an external reflector is present, HUNTER is called to determine the number of times each array and/or hole is used in the reflector. GOCURS is used to determine the number of times each unit, array, and hole is used in the problem. GTVOLS is called to calculate and print the number of occurrences for each unit and the corresponding total volumes for the entire system. CLEAR is called to initialize the array for the total volume of each mixture used in the problem prior to calculating and printing those totals.

HUNTER - This subroutine determines the number of times each unit, array, and hole is used. CLEAR is used to initialize storage arrays for the present hole level and the next hole level. MOVE is used to move the storage arrays.

GOCURS - This subroutine loops over the array size and calls HUNTER to determine the number of times each unit, lattice or array, and hole is used in the problem.

GTVOLS - This subroutine calculates the total volume of each region for the entire problem by multiplying the volume of the region by the number of times the region is used in the problem.

### C4.3.4.14. Generate Printer Plot

This portion of the program, illustrated in Fig. C4.3.17, generates printer plots of a 2-D slice through the KENO V.a geometry. As many plots as are desired can be printed.

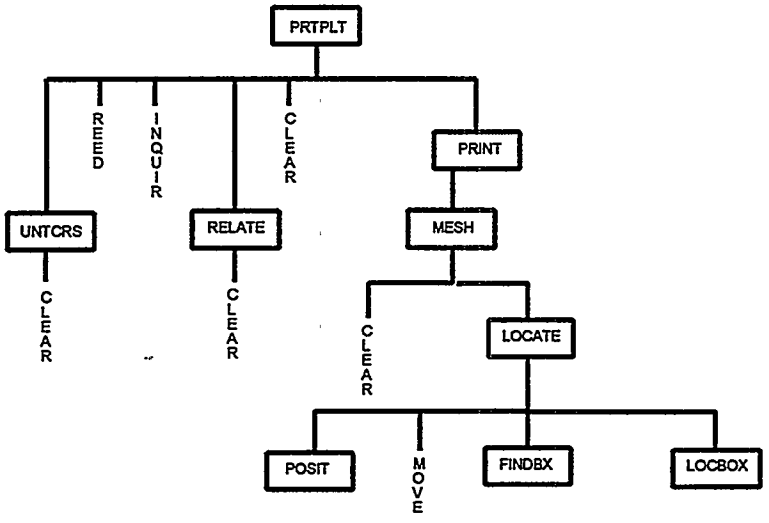



Figure C4.3.17 Flowchart for printer plots

PRTPLT - This subroutine controls the generation of the printer plots. The library routine CLEAR is called to zero the necessary space. Subroutine UNTCRS is called to generate a unit correspondence array. REED is used to load the picture data from direct access. INQUIR is called to return the value of the next direct-access data block. The picture title is printed, and subroutine RELATE is called to print a heading for the symbol map and print the symbol map. The picture coordinates, direction cosines, and number of symbols across and down the page, as well as the step intervals, are printed. Then subroutine PRINT is called to generate the actual picture.

UNTCRS - This subroutine calls CLEAR to zero the unit correspondence array. It then loads the appropriate unit number in the appropriate position of the array.

RELATE - This subroutine calls CLEAR to zero the reverse correspondence array. It then prints the plot header and loads the array that correlates the symbols.

PRINT - This subroutine determines the number of pages that will be needed to print the printer plot picture. Subroutine MESH is called to load the appropriate mixture numbers, unit numbers, or bias ID numbers for each line of print for the picture. Then PRINT prints the line of symbols corresponding to them.

MESH - This subroutine loads an array that contains the appropriate mixture number, unit number, or bias ID number for each character in a line. CLEAR is called to initialize arrays if nested holes or nested arrays are present in the problem. LOCATE is called to determine the geometry region and unit or box type that contains each mesh point. MESH then loads the mixture number unit number, or bias ID number.

LOCATE - This subroutine is responsible for determining the geometry region for each mesh point in the picture. Subroutine POSIT is called to determine the region that contains the specified mesh point. If array data are used, MOVE is called to load the current array data into the array stack. If the mesh point is within an array, FINDBX is called to determine the position within the lattice or array that contains the mesh point. LOCBOX is then called to determine the unit or box type that contains the mesh point. POSIT is used to determine the geometry region that contains the mesh point. If the mesh point is in a region that contains a hole, the coordinate of the mesh point is translated to the hole and POSIT is called again.

FINDBX - This subroutine locates the position in an array that contains a specified point.

LOCBOX - This function returns the unit or box type at a given position in an array.

POSIT - This subroutine determines the region within a unit that contains a specified point.

### C4.3.4.15. Write KENO V.a Input Data on Restart File

This portion of the program, illustrated in Fig. C4.3.18, is responsible for writing all data except the calculated results on the restart data file. This section of the program is omitted if a unit number has not been assigned for the restart data file. This information is entered as parameter data, WRS=, as described in Sect. F11.5.3. The calculated results are written on the restart data file later in the program. The restart data file is used for restarting a problem.

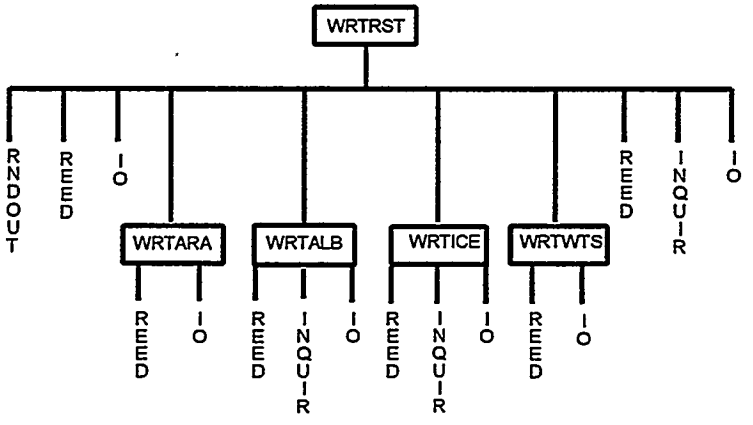



Figure C4.3.18 Flowchart for writing data on the restart data file

WRTRST - This subroutine writes the input data on the restart data file. RNDOUT is called to return the seed of the next random number. The problem title and parameter data are in the first record written on the restart data file. The array containing the IDs of the 1-D cross sections is then written. The geometry region data, mixing table data, extra 1-D data, biasing data, start data, plot data, and energy and inverse velocities are written.

The unit orientation header record is written, and WRTARA is called to write the unit orientation data. The albedo header record is written, and WRTALB is called to write the albedo data. The cross-section header record is written, and WRTICE is called to write the cross sections. The header record for user-supplied weighting data is written, and WRTWTS is called to write out the user-supplied weighting or biasing data.

WRTARA - This subroutine is called from WRTRST to write the array number, array size, and corresponding unit orientation array on the restart data file for each array that is entered in the problem. The library routine REED is called to load the direct-access pointers for the albedo data block. INQUIR is called to return the value of the next direct-access record. REED is used to read the data from the direct-access data file, and IO is used to write it on the restart data file.

WRTALB - This subroutine is called from WRTRST to write the albedo data on the restart data file. The library routine REED is used to read the albedo data from the direct-access data file, and IO is used to write the data on the restart data file.

WRTICE - This subroutine is called from WRTRST to write the cross-section data block on the restart data file. REED is called to load the direct-access pointers for the cross-section data blocks. INQUIR is called to return the value of the next direct-access record. The library routine REED is used to read the rest of the cross-section information from the direct-access data file, and IO is used to write it on the restart data file.

WRTWTS - This subroutine is called from WRTRST to write the biasing input data block on the restart data file. The library routine REED is used to read the data block from the direct-access data file, and IO is used to write it on the restart data file.

### C4.3.5 THE SEARCH DATA PROCESSOR

This portion of the program, shown in Fig. C4.3.19, is responsible for processing search data to be used by module MODIFY.

RDOPT - This subroutine is responsible for reading and processing the search data. AREAD is used to read the keywords READ, SEARCH, and the search type. Two calls are made to subroutine OPTYP to read and process data for an optimum value search and to read and process data for a minimum value search. Subroutine FIXTYP reads and processes data for a fixed value search. The library routine INQUIR is called twice to set the next direct-access block number. The library routine RITE is called to write common information for the control module MODIFY on the direct-access data file. The library routine CLEAR is called to clear space for data that is saved from pass to pass, such as the k-effective and search parameters. RITE is called twice more to rewrite the direct-access pointer and length arrays for the KENO V.a and CSAS data.

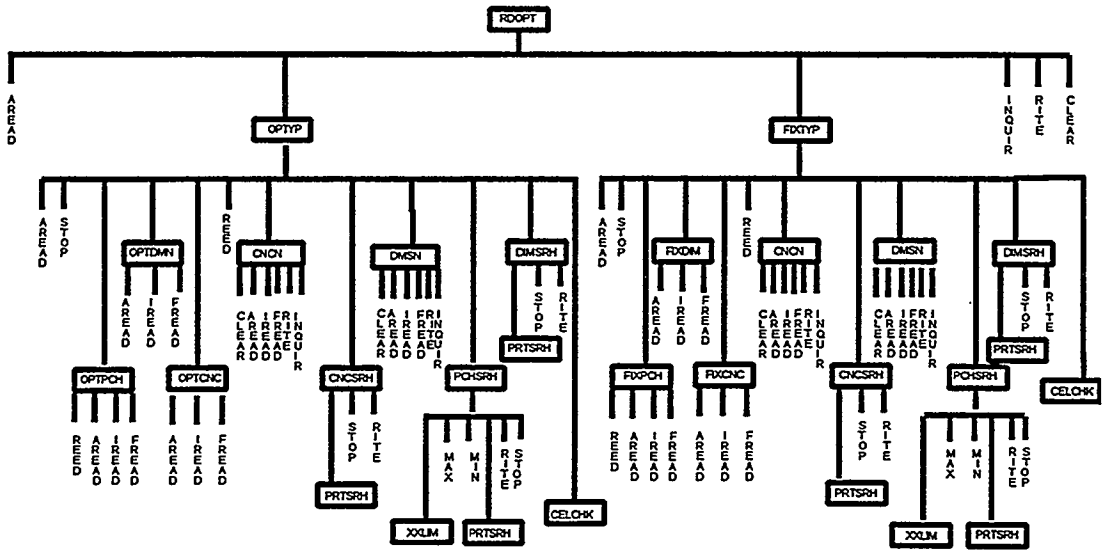



Figure C.4.3.19 Flowchart of the Search Data Processor

The KENO V.a pointers and their associated data are:

LPOINT(1) is the pointer for the current geometry region data
LPOINT(2) is the pointer for the unit orientation data
LPOINT(3) is the pointer for the mixing table information
LPOINT(4) is the pointer for the extra data
LPOINT(5) is the pointer for the weighting importance region data
LPOINT(6) is the pointer for the data defining start type 6
LPOINT(7) is the pointer for the albedo data
LPOINT(8) is the pointer for the mixed cross sections
LPOINT(9) is the pointer for the energy and inverse velocities
LPOINT(10) is the pointer for the plot or picture data
LPOINT(11) is the pointer for the biasing data
LPOINT(12) is the pointer for the albedo cross-section energy correspondence data

The CSAS pointers and their associated data are:

LPOINT(1) is the pointer for COMMON/REC1/
LPOINT(2) is the pointer for the original geometry region data
LPOINT(3) is the pointer for the original mixing table
LPOINT(4) is the pointer for the search data arrays
LPOINT(5) is the pointer for the calculated data that are saved from pass to pass

OPTYP - This subroutine is responsible for providing data for both optimum value searches and minimum value searches. AREAD is used to read the search type (pitch, dimension, or concentration). Pointers are created for the data arrays required for the search, and STOP is called to write a message if more computer storage is required. OPTPCH, OPTDMN, OPTCNC, CNCN, CNCSRH, DMSN, PCHSRH, and DIMSRH are used to read and process the search data. REED is called to load data from the direct-access device.

OPTPCH - This subroutine uses REED to reload the geometry data from the direct-access data file. The library routines AREAD, IREAD, and FREAD are used to read the optional search parameters for an optimum pitch search. The processed data are written on the scratch unit.

OPTDMN - This subroutine uses the library routines AREAD, IREAD, and FREAD to read the optional search parameters for an optimum dimension search.

OPTCNC - This subroutine uses the library routines AREAD, IREAD, and FREAD to read the optional search parameters for an optimum concentration search.

CNCN - This subroutine calls CLEAR to zero arrays that hold concentration search data. AREAD, IREAD, and FREAD are used to read the concentration search data that are then written on the scratch unit. RITE is called to load the mixing table data on the direct-access data file. INQUIR is called to set the next direct-access block number.

CNCSRH - This subroutine reads concentration search data from the scratch unit, checks data, sets the maximum and minimum constraints, calls PRTSRH to print the search data table, calls STOP to write an error message if no valid search commands are found, and uses RITE to write the search data on the direct-access data file.

PRTSRH - This subroutine prints the search data table that includes the search parameters and the search commands.

DMSN - This subroutine reads the search commands for a search that alters the geometry dimensions. CLEAR is used to zero data arrays. AREAD, IREAD, and FREAD are used to read the data. After reading the data, some of it is written on the scratch unit. The geometry data are written to a direct-access data file using RITE. INQUIR is used to set the next direct-access data block.

PCHSRH - This subroutine reads optimum pitch search data from the scratch unit and fills the search data arrays for an optimum pitch search. XXLIM, MIN, and MAX are used to determine the maximum and minimum search constraints. PRTSRH is called to print the search data table.
STOP is called to write an error message if no valid search commands are found. The library routine RITE is used to write the search data arrays to a direct-access data file.

XXLIM - This function is used to determine the maximum coordinate of the geometry region interior to the specified face of the unit.

DIMSRH - This subroutine reads dimension search data from the scratch unit, sets the maximum and minimum constraints, calls PRTSRH to print the search data table, calls STOP to write an error message if no valid search commands are found, and uses RITE to write the search data arrays to a direct-access data file.

FIXTYP - This subroutine is responsible for providing data for a "critical" or "fixed k" search. AREAD is used to read the search type (pitch, dimension, or concentration). Pointers are created for the data arrays required for the search, and STOP is called to write a message if additional computer storage is required. FIXPCH, FIXDIM, FIXCNC, CNCN, CNCSRH, DMSN, PCHSRH, and DIMSRH are used to read and process the search data. REED is called to read data from the direct-access device.

FIXPCH - This subroutine uses REED to reload the geometry data from the direct-access data file. The library routines AREAD, IREAD, and FREAD are used to read the optional search parameters for a critical pitch search. The processed search data arrays are written on the scratch unit.

FIXDIM - This subroutine uses the library routines AREAD, IREAD, and FREAD to read the optional search parameters for a critical dimension search.

FIXCNC - This subroutine uses the library routines AREAD, IREAD, and FREAD to read the optional search parameters for a critical concentration search.

CELCHK - This subroutine checks the input mixtures used in a concentration search to ensure mixtures used to create cell-weighted mixtures are not modified during the search.

### C4.3.6 MODIFY, THE SEARCH MODULE

This segment of the program, shown in Fig. C4.3.20, is part of a control module that is responsible for altering data for a search. CSAS4 and CSAS4X activate both MODIFY and KENO V.a to achieve a search. MODIFY provides new geometry data that are used by KENO V.a to calculate the corresponding k-effective. This procedure is repeated numerous times, and the data resulting from the calculations are analyzed until convergence is achieved, the specified number of search passes have been completed, or the problem uses up the allotted time. See Figs. C4.3.1 and C4.3.2 for the flowcharts that illustrate this procedure. LODCOM is called via ALOCAT from SEARCH, the main program of the control module, MODIFY. SEARCH is activated by the control module MODIFY. Figure C4.3.20 shows the subroutines that are part of the control module MODIFY and are responsible for actually altering the data.

LODCOM - This subroutine initiates the data modification procedure of module MODIFY. It strips the individual commons out of COMMON/REC1/ using the library routine MOVE, sets up storage for the required data arrays, and calls subroutine LODATA to load the data into arrays.

LODATA - This subroutine calls the library routine STOP to write an error message if incorrect search data are entered. LODATA uses REED to load the pass-to-pass data into storage from the direct-access data file. RESTRT is called to load data from the restart data file. PRTSRH is called to print the table of search data. AVGKEF is called to load the data that are saved from pass to pass. REED is used to load the search command arrays before calling OPTMIZ to evaluate the

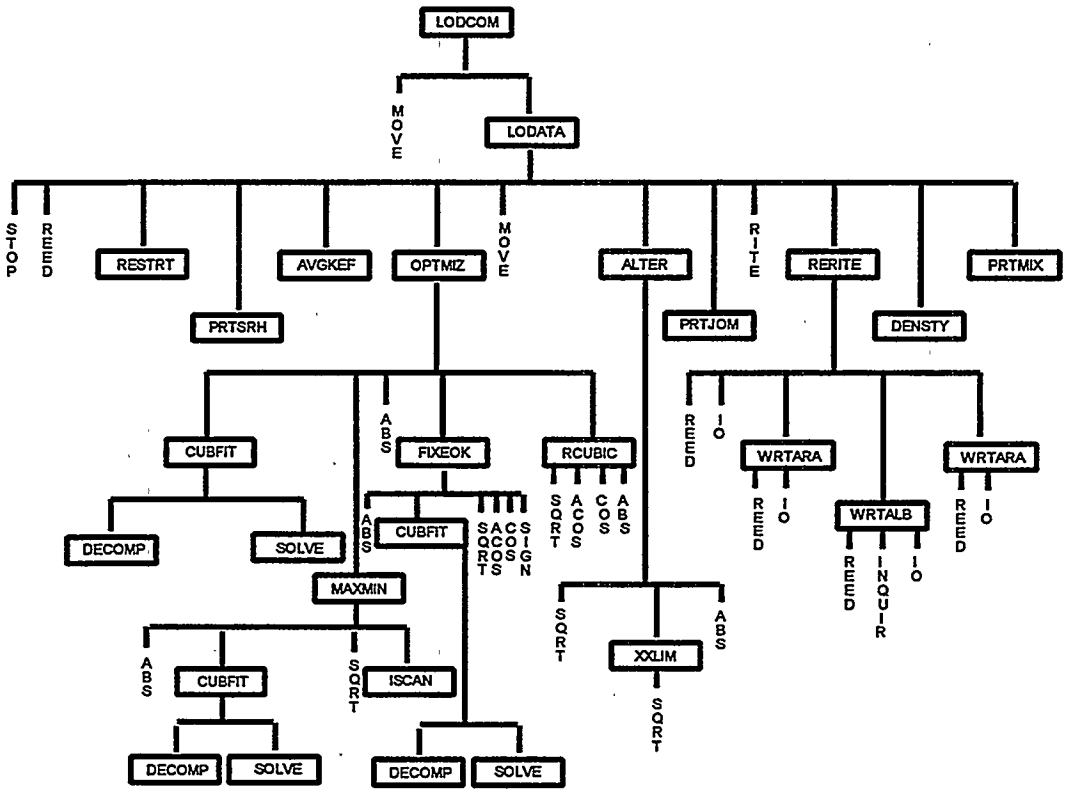



Figure C4.3.20 Flowchart of the search module, MODIFY

parameter values corresponding to values of k-effective. If a pitch or dimension search is to be made, REED is called to load the geometry data and temporary pointers are created for the geometry data. MOVE is called to load the new dimensions into the old dimensions array. ALTER is called to alter the geometry data based on the parameter values returned by OPTMIZ. If convergence has been achieved, a message is written and PRTJOM is called to print the corresponding KENO V.a geometry information. After convergence is achieved, ALTER is called to calculate the best estimate of the dimensions and PRTJOM is called to print them. RITE is used to write the new geometry data on the direct-access data file. REWRITE is called to rewrite the restart data file for KENO V.a, and RITE is used to rewrite the pass-to-pass search data on unit 90. If a concentration search is to be made, REED is called to load the permanent pointers and temporary pointers are created for the mixing table data. MOVE is used to load the new densities into the old density array. DENSTY is called to modify the concentrations, and if convergence is achieved, PRTMIX is called to print the converged mixing table data. After convergence is achieved, DENSTY is called to calculate the best estimate for the mixing table and PRTMIX is called to print it. RITE is used to rewrite the pass-to-pass data on unit 90.

RESTRT - This subroutine reads the restart data file from unit 95 and loads the k-effective by generation array for processing by subroutine AVGKEF.

AVGKEF - This subroutine loops over the number of generations. It determines the average k-effective and corresponding deviation, skipping an additional generation each time but never skipping more than half the generations. SQRT is used in determining the deviations. The minimum deviation and its corresponding k-effective are stored for use by subroutine OPTMIZ.

OPTMIZ - This subroutine includes methods for an optimization search and a fixed value search. The optimization search[1] is capable of determining a maximum or a minimum. For an optimum pitch search, it searches for the value of a parameter that corresponds to the maximum k-effective. The fixed value search finds the value of a parameter that corresponds to a fixed value of k-effective. Both searches rely on least squares curve fitting to obtain the information needed to make an estimate. The data points are least squares fitted to a cubic polynomial, which is solved analytically for parameter values. CUBFIT is called to determine the coefficients of the cubic polynomial. MAXMIN is called to provide the parameter value corresponding to the optimum k-effective, using a cubic fit. If a fixed k calculation is to be made, FIXEDK is called to solve the cubic equation. RCUBIC provides the analytic solution of the cubic equation. The FORTRAN-supplied function ABS provides the absolute value of a variable, and DSQRT returns the square root of a double-precision variable.

CUBFIT - This subroutine fits the input data to a cubic equation by determining the coefficients of the cubic polynomial. DECOMP decomposes the matrix, and SOLVE provides the solution for a linear system.

DECOMP - This subroutine decomposes a real matrix by Gaussian elimination and estimates the condition of the matrix.

SOLVE - This subroutine uses the triangularized matrix and pivot vector provided from DECOMP to determine the solution of the linear system.

MAXMIN - This subroutine uses a least-squares fit to a cubic polynomial to determine the parameter value corresponding to the maximum value of k-effective. It is also capable of determining the parameter value corresponding to the minimum k-effective. One portion of the subroutine is devoted to determining the optimum for a cubic having a maximum and a minimum; another portion of the subroutine determines the optimum for a cubic with only a point of inflection. CUBFIT is utilized in both portions to fit the input data to a cubic equation. SQRT is used in the first portion to evaluate the discriminate. The library routine ABS is used to provide the absolute value of a variable. ISCAN is called to ensure that calculations are not duplicated.

ISCAN - This function scans a vector for a specific value and returns the index if the value is found.

FIXEDK - This subroutine performs a fixed value search. It utilizes an extended mean value theorem method to determine the value of the parameter corresponding to the desired value of k-effective. CUBFIT supplies a least squares fit to a cubic polynomial. The library routines ABS, SQRT, ACOS, COS, and SIGN are used in providing the analytical solution of the cubic.

RCUBIC - This subroutine is responsible for determining the analytical solution to a cubic. It utilizes the library routines SQRT, ACOS, COS, and ABS.

ALTER - This subroutine translates changes in the search parameters into changes in the geometric dimensions. CLEAR is used to zero the region correspondence array. The search command array is used to determine which geometry regions will be altered. For pitch or dimension search, the geometry region data are altered. For a pitch search, the function XXLIM is used to provide the maximum coordinate of the interior region corresponding to the specified face of the unit. This capability ensures that the pitch does not decrease beyond the point of tangency. The new dimensions are calculated from the original dimensions and the parameters provided by OPTMIZ.

XXLIM - This function provides the maximum coordinate of the interior region corresponding to the specified face of the unit. SQRT is used to return the square root of a variable.

PRTJOM - When convergence has been achieved, this subroutine is called to print the geometry data. The corresponding parameter, pass number, and k-effective are printed by LODATA.

RERITE - This subroutine is called to rewrite the restart data file used by KENO V.a. It contains the new geometry data for which a k-effective will be calculated. REED is used to read the data from the direct-access data file, and IO is used to write it on the restart data file. WRTARA, WRTALB, and WRTWTS are used to write the KENO V.a array, albedo, and weighting data on the restart data file.

WRTARA - This subroutine is called from WRTRST to write the array number, array size, and corresponding unit orientation array on the restart data file for each array that is entered in the problem. The library routine REED is used to read the data from the direct-access data file, and IO is used to write it on the restart data file.

WRTALB - This subroutine is called from WRTRST to write the albedo data on the restart data file. The library routine REED is used to read the albedo data from the direct-access data file, INQUIR is called to return the value of the next direct-access record, and IO is used to write the albedo data on the restart data file.

WRTWTS - This subroutine is called from WRTRST to write the biasing input data block on the restart data file. The library routine REED is used to read the data block from the direct-access data file, and IO is used to write it on the restart data file.

DENSTY - This subroutine modifies the number densities of the mixture components according to search commands for a concentration search.

PRTMIX - This subroutine prints the new mixing table resulting from the concentration search.

### C4.3.7 REFERENCE

1. M. J. Lorek, *Improved Criticality Search Techniques for Low- and High-Enriched Systems*, ORNL/NUREG/CSD/TM-13, Union Carbide Corp., Nucl. Div., Oak Ridge Natl. Lab., 1981.

## C4.4 INPUT DATA GUIDE

This section describes the input data required for Criticality Safety Analysis Sequence No. 4 (CSAS4X). Several subsets of the CSAS4X sequence are available to achieve several different levels of processing. These are listed in Table C4.4.1 with the functional modules they invoke. Note that modules with an "X" in their names create cell-weighted libraries by using XSDRNPM-S.

Table C4.4.1 Functional modules executed by CSAS control modules

| Control module | Search function | Functional modules executed by the control module | | | | |
|---|---|---|---|---|---|---|
| CSASI | No search | BONAMI | NITAWL-II | | ICE | |
| CSASIX | No search | BONAMI | NITAWL-II | XSDRNPM | ICE | |
| CSASN | No search | BONAMI | NITAWL-II | | | |
| CSAS1X | No search | BONAMI | NITAWL-II | XSDRNPM | | |
| CSAS25 | No search | BONAMI | NITAWL-II | | KENO V.a | |
| CSAS2X | No search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | |
| CSAS4 | Search | BONAMI | NITAWL-II | | KENO V.a | MODIFY[a] |
| CSAS4X | Search | BONAMI | NITAWL-II | XSDRNPM | KENO V.a | MODIFY |

[a]MODIFY is a control module.

Each sequence generates various libraries that may be saved for future use. Table C4.4.2 lists the various I/O units that may be saved and their definitions. Table C4.4.3 lists those I/O units containing cross-section libraries that are generated by each sequence. See Table C4.5.1 in Sect. C4.5 for a complete list of I/O units utilized by Criticality Safety Analysis Sequence No. 4 and Table C4.5.2 for the I/O units utilized by the associated functional modules.

The input data for these CSAS modules are composed of three broad categories of data. The first (Material Information Processor, including Standard Composition Specification Data and Unit Cell Geometry Specification) specifies the cross-section library and defines the composition of each mixture and the unit cell geometry that is used to process the cross sections. This data block is necessary for CSASN, CSAS1X, CSASI, and CSASIX. The second category of data, the KENO V.a input data, are used to specify the geometric and boundary conditions that represent the physical 3-D configuration of the problem. Both data blocks are necessary for CSAS25 and CSAS2X. The last category of data is the search data. All three data blocks are required for CSAS4 and CSAS4X.

All data are entered in free form, allowing alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. Eighty columns of data entry are allowed, simulating a card of data. Data can usually start or end in any column. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating values. For example, 10 will be interpreted as 10.0. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as $1.0 \times 10^4$.

Table C4.4.2 I/O units on which cross-section libraries are written

| Unit | Type of data | Creating module |
|---|---|---|
| 2 | Resonance-corrected mixed working library | ICE |
| 3 | Resonance-corrected, cell-weighted working library | XSDRNPM |
| 4 | Resonance-corrected working library | NITAWL-II |
| 14 | Resonance-corrected, mixed cross-section, MORSE/KENO library. If the invoked sequence contains an X in its name, this library contains a cell-weighted mixture identified as mixture 500 | ICE, KENO V.a |

Table C4.4.3 I/O units generated by CSAS and associated modules

| | | Unit numbers of generated cross-section libraries | | | |
|---|---|---|---|---|---|
| Module name | Primary product | 2 | 3 | 4 | 14 |
| CSASN | Resonance-corrected working library | | | X | |
| CSAS1X | XSDRNPM eigenvalue calculation | | X | X | |
| CSASI | Resonance-corrected, mixed cross-section MORSE/KENO library | X | | X | X |
| CSASIX | Resonance-corrected, cell-weighted, mixed cross-section, MORSE/KENO library | X | X | X | X |
| CSAS25 | KENO V.a k-effective | | | X | X |
| CSAS2X | KENO V.a k-effective using homogenized cell | | X | X | X |
| CSAS4 | Dimension alterations | | | X | X |
| CSAS4X | Dimension alterations using homogenized cell | | X | X | X |

The word "END" is a special data item. An "END" may have a name or label associated with it. The name or label associated with an "END" is separated from the "END" by a single blank and is a maximum of 12 characters long. *At least 2 blanks MUST follow every labeled and unlabeled "END." It is the user's responsibility to ensure compliance with this restriction. Failure to observe this restriction can result in the use of incorrect or incomplete data without the benefit of warning or error messages.*

Multiple entries of the same data value can be achieved by specifying the number of times the data value is to be entered, followed by either R, *, or $, followed by the data value to be repeated. Imbedded blanks are not allowed between the number of repeats and the repeat flag. For example, 5R12, 5*12, 5$12, or 5R 12, etc., will enter five successive 12s in the input data. Multiple zeros can be specified as nZ where n is the number of zeroes to be entered. The SCALE free-form reading routines are described in more detail in Sect. M3.

The purpose of this section is to define the input data in discrete subsections relating to a particular type of data. Tables of the input data are included in each subsection, and the entries are described in more detail after the tables illustrated in Fig. C4.4.1. Section C4.4.1 is provided for experienced users and contains all the tables that are used to describe the input data. Section C4.4.2 describes input data requirements and the data-checking capability. Sections C4.4.3 through C4.4.8 describe the Material Information Processor data, Sect. C4.4.9 summarizes the KENO V.a data that are explained in detail in Sect. F11.4, and Sect. C4.4.10 describes the search data.

Resonance-corrected cross sections utilize Dancoff corrections based on the unit cell specification. Because only one unit cell can be specified in the data, resonance corrections can only be applied to one mixture via conventional methods. However, if multiple resonance-corrected mixtures are required, the necessary data can be entered via the OPTIONAL PARAMETER DATA described in Sect. C4.4.8 and Tables C4.4.4 and C4.4.13.

To check the input data without actually processing the cross sections, the words "PARM=CHECK" or "PARM=CHK" should be entered, starting in column 11 of the analytical sequence specification for IBM versions as shown below.

```
=CSAS4   PARM=CHK
           or
#CSAS4   PARM=CHK
```

This would cause the input data for CSAS to be checked and appropriate error messages to be printed. If plots are specified in the data, they will be printed. This feature allows the user to debug and verify the input data while using a minimum of computer time. Many problems can be checked in 10 s or less. The number of words of storage requested by CSAS and its associated sequences is defaulted to 100,000 words of storage. This can be respecified through the use of the PARM command on the analytical sequence specification. For example, if 150,000 words of storage are required for the problem, "PARM=SIZE=150,000" should be entered, starting in column 11 of the analytical sequence specification as shown below.

```
=CSAS4   PARM=SIZE=150000
```

To combine checking and size specification, specify "PARM=(CHECK,SIZE=nnn)" or "PARM=(CHK,SIZE=nnn)" starting in column 11 or beyond on the analytical sequence specification. The value of nnn is given in words of storage. The minimum value of nnn is the amount of storage required by the problem, and the maximum value of nnn is the available computer storage.

## C4.4.1 INPUT DATA SUMMARY FOR EXPERIENCED USERS

Section C4.4.1 is provided for the convenience of experienced users. Other users should skip to Sect. C4.4.2, page C4.4.26. All of the input data for Criticality Safety Analysis Sequence No. 4 are illustrated in Fig. C4.4.1, and are summarized in Table C4.4.4. This table is provided as an input data guide for users who are familiar with CSAS data requirements. Each component table of Table C4.4.4 is repeated and discussed in detail in Sects. C4.4.4 through C4.4.12. The KENO V.a data are included in this table in very general terms. Table C4.4.5 contains a summary of the KENO V.a input data. They are described in detail in Sect. F11.4 with

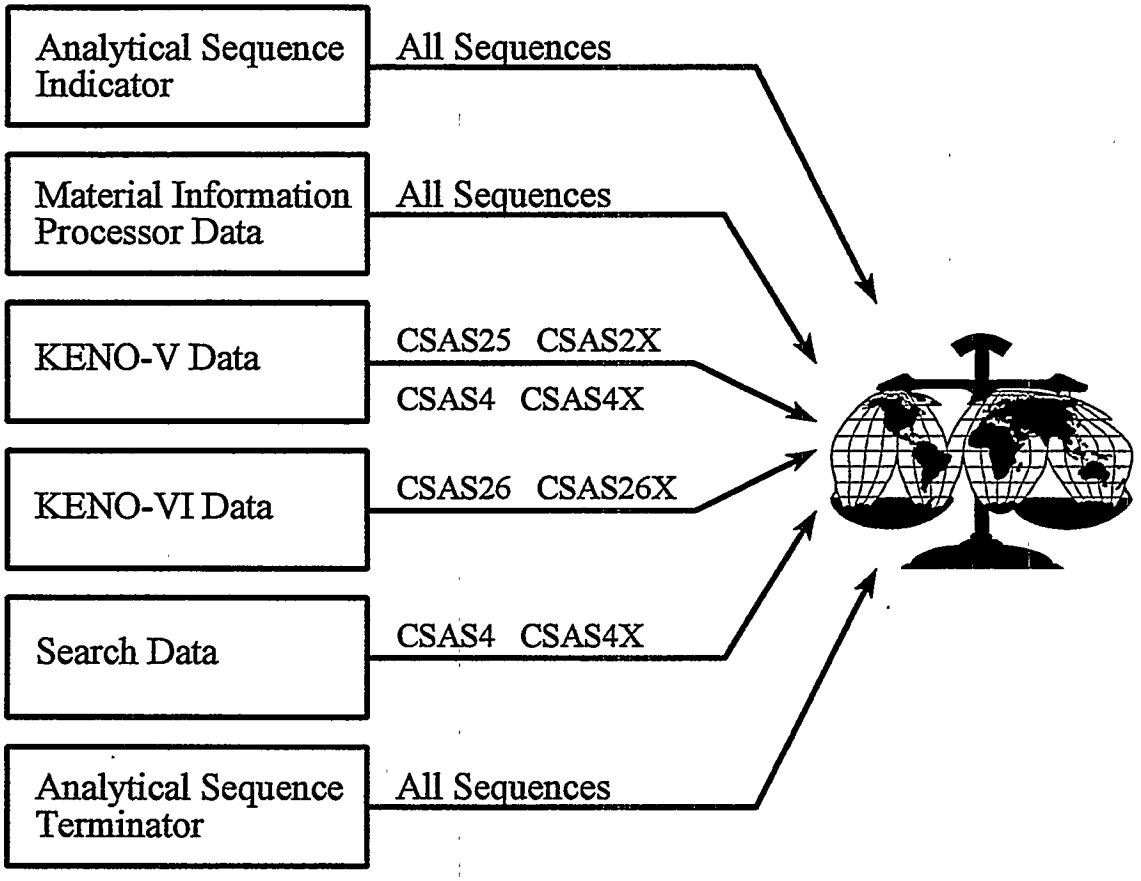



Figure C4.4.1 Outline of CSAS input data

## Table C4.4.4 Input data requirements

**Summary of data input requirements** (see Sect. C4.4.2 for details)

| | Type of data | Analytical sequence | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | Analytical sequence specification | =CSASN | =CSAS1X | =CSASI | =CSASIX | =CSAS25 | =CSAS2X | =CSAS4 | =CSAS4X |
| 2 | Material Information Processor data | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| 3 | KENO V.a data | No | No | No | No | Yes | Yes | Yes | Yes |
| 4 | Search data | No | No | No | No | No | No | Yes | Yes |
| 5 | END for the analytical sequence | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |

**Outline of Material Information Processor data** (see Sect. C4.4.3 for details)

| Data position | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | TITLE | Enter a title | 80 characters |
| 2 | Cross-section library name | HANSEN-ROACH<br>27GROUPNDF4<br>44GROUPNDF5<br>238GROUPNDF5<br>27BURNUPLIB<br>etc. | The currently available libraries are listed in Sect. M4. |
| 3 | Type of calculation | INFHOMMEDIUM<br>LATTICECELL<br>MULTIREGION | These are the available options. See the explanation in Sect. C4.4.3. |
| 4 | Standard Composition specification data | Enter the appropriate data | Terminate this data block with END COMP. See Sect. C4.4.4. |
| 5 | Unit cell geometry specification | Enter the appropriate data (optional for INFHOMMEDIUM) | Optional for INFHOMMEDIUM.<br>See Sect. C4.4.5 for INFHOMMEDIUM.<br>See Sect. C4.4.6 for LATTICECELL.<br>See Section C4.4.7 for MULTIREGION. |
| 6 | Optional parameter data | Enter the desired data | Precede this data block by MORE DATA if more parameter data are to be entered. Otherwise, omit these data entirely. See Sect. C4.4.8. |

Table C4.4.4 (continued)

**Outline of standard compositions specification data** (see Sect. C4.4.4 for details)

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 1 | SC | Standard composition component name | Always | Enter once for each standard composition. Enter the alphanumeric description from Sect. M8.2. Additional allowed names include those beginning with ARBM for arbitrary materials and SOLN for solutions |
| A1 | ROTH | Theoretical density of material (g/cc) | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A2 | NEL | Number of elements in the material | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A3 | IVIS | No longer used but must still be entered | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1 |
| A4 | ICP | Compound indicator | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 1 for a compound, 0 for alloys, mixtures, etc. |
| A5 | IRS | No longer used but must still be entered | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1 |
| A6 | NCZA | ID number (from far right column of Table M8.2.1 or M8.2.2) | ARBM | Repeat the sequences A6 and A7 for each element or isotope in the arbitrary material before entering entry number 2. Enter the number from the far right column of Table M8.2.1 or M8.2.2. (Premixed standard compositions cannot be used in an arbitrary material definition) |
| A7 | ATPM | Number of atoms of this element per molecule of arbitrary material | ARBM & ICP=1 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry number 2. Do not enter a value unless ICP=1 |
| | | or | or | or |
| | | Weight percent of this element in this arbitrary material | ARBM & ICP=0 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry number 2. Do not enter a value unless ICP=0 |

Table C4.4.4 (continued)

**Outline of standard compositions specification data (continued)**

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 2 | MX | Mixture ID number | Always | Enter once for each standard composition component |
| S1 | FD | Fuel density (grams of U or Pu per liter of solution) | SOLN | Enter once for a solution |
| S2 | AML | Acid molarity of the solution | SOLN | Enter once for a solution. AML=0 if there is no acid in the solution |
| O1 | SPGR<br>or<br>ROTH | Specific gravity of the solution<br>or<br>Density of the basic standard composition | Optional | If the specific gravity (SPGR) of the solution is known, it should be entered as SPG=SPGR<br>or<br>if the density of a basic standard composition (ROTH) is to be entered, use DEN=ROTH |
| 3 | VF | Density multiplier | See comment column | Enter the density multiplier (density fraction, volume fraction, or a combination). Default value is 1. This item can be omitted if entries 4, 5, 6a, and 6b are also omitted. VF=0 is not allowed for SOLN or ARBM |
| 4 | ADEN | Number density (atoms/b-cm) for the nuclide | VF=0 | Enter only if VF=0.0 |
| 5 | TEMP | Temperature, in K | See comment column | Default value is 293 K. This entry can be omitted if entries 6a and 6b are also omitted |
| 6a | IZA | Isotope's ZA number | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 6b | WTP | Weight percent of the isotope | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0.0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 7[a] | END | Terminate a standard composition | Always | Enter once for each standard composition component. This terminates the data for a standard composition component. Enter END to terminate the component. Repeat entries 1 through 7 until all the mixtures have been defined. At least two blanks must separate entry 7 from the next entry |
| | END COMP | Terminate the data block | Terminus | Enter once for a problem. Enter the words END COMP when all the standard composition components have been described. At least two blanks must follow the END COMP |

[a]NOTE: Entry 7 should not begin in column 1 unless a name is associated with it. At least two blanks should separate the last entry 7 from the END COMP.

Table C4.4.4 (continued)

**Optional unit cell specifications for INFHOMMEDIUM problems** (see Sect. C4.4.5 for details)

| Entry number | Variable name | Type of data | Data entry | Comments |
|---|---|---|---|---|
| 1 | NAME | Keyword | CELLMIX | Initiate reading cell data for INFHOMMEDIUM |
| 2 | MFUEL | Mixture number in the cell | Mixture number | Specifies the mixture number to be used in the cell. Defaults to the smallest mixture number entered in the standard composition data |

NOTE: Unit cell data are necessary for INFHOMMEDIUM only if a mixture number other than the smallest mixture number is to be used in the cell and if a control sequence that executes XSDRNPM has been specified. If mixtures 1, 2, and 3 are specified and if mixture 3 is to be used in the cell, enter CELLMIX 3.

Table C4.4.4 (continued)

**Unit cell specification for LATTICECELL problems** (see Sect. C4.4.6 for details)

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CTP | Type of lattice | Always | | Describes the type of lattice or array configuration |
| | | | | SQUAREPITCH | Use for cylindrical rods in a square pitch |
| | | | | ASQUAREPITCH | Use for annular cylindrical rods in a square pitch |
| | | | | TRIANGPITCH | Use for cylindrical rods in a triangular pitch |
| | | | | ATRIANGPITCH | Use for annular cylindrical rods in a triangular pitch |
| | | | | SPHSQUAREP | Use for spherical pellets in a cubic lattice |
| | | | | ASPHSQUAREP | Use for annular spherical pellets in a cubic lattice |
| | | | | SPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | SYMMSLABCELL | Use for a symmetric array of slabs |
| | | | | ASYMSLABCELL | Use for a periodic but asymmetric array of slabs |
| 2 | PITCH | Array pitch (cm) | Always | Appropriate dimension | The center-to-center spacing (cm) between fuel lumps. For asymmetric slab cell, enter the distance from the center of one moderator to the center of the other moderator (cm) |
| 3 | FUELOD | Outside dimension of fuel (cm) | Always | Appropriate dimension | Outside diameter of fuel (cm), or the thickness of the fuel in a slab |
| 4 | MFUEL | Fuel mixture number | Always | Mixture number | Mixture number representing the fuel |
| 5 | MMOD | Moderator mixture number | Always | Mixture number | Mixture number representing the moderator |
| 6 | MMOD2 | 2nd moderator mixture number | Annular cell | Mixture number | Mixture number representing the second moderator |
| 7 | TKMOD2 | 2nd moderator thickness | ASYMSLABCELL | Thickness | Thickness of the second moderator (cm) for ASYMSLABCELL |
| | | or | or | or | or |
| | | 2nd moderator diameter (cm) | annular cell | diameter | diameter of inner moderator (cm) for other annular cells |
| 8 | CLADOD | Outside diameter of clad (cm) | If clad | Clad OD | OMIT IF NO CLAD. For a slab, CLADOD is the sum of thickness of the fuel, gap, and clad |
| 9 | MCLAD | Clad mixture number | If clad | Mixture number | OMIT IF NO CLAD. Mixture number representing the clad |
| 10 | CLADID | Inside diameter of clad (cm) | If gap | Clad ID | OMIT IF NO GAP between the fuel and clad |
| 11 | MGAP | Gap mixture number | If gap | Mixture number | OMIT IF NO GAP between the fuel and clad. A mixture number of zero is often used |
| 12 | END | Terminate LATTICECELL data | Always | END | Terminate the LATTICECELL input data by entering the word END. Do not start in column 1. At least two blanks must follow entry number 12 |

Table C4.4.4 (continued)

**Unit cell specification for MULTIREGION problems** (see Sect. C4.4.7 for details)

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CS | Type of geometry | Always | | Describes the type of geometry. The options are: |
| | | | | SLAB | Use for slab geometry |
| | | | | CYLINDRICAL | Use for cylindrical geometry |
| | | | | SPHERICAL | Use for spherical geometry |
| | | | | BUCKLEDSLAB | Use for slab geometry with a buckling correction for the two transverse directions |
| | | | | BUCKLEDCYL | Use for cylindrical geometry with a buckling correction in the axial direction |
| 2 | BR | Right boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | | Default is VACUUM. Describes the right/outside boundary condition |
| | | | | VACUUM | This provides a non-return condition at the boundary |
| | | | | REFLECTED | Do not use for cylindrical or spherical |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | This provides isotopic return at the boundary |
| 3 | BL | Left boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | | Default is REFLECTED. Describes the left boundary condition |
| | | | | VACUUM | This provides a non-return condition at the boundary |
| | | | | REFLECTED | Recommended for cylindrical or spherical |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | This provides isotropic return at the boundary |
| 4 | ORGN | Location of left boundary on the x-axis (cm) | BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | Appropriate dimension | Default is 0.0. Should not be changed for cylindrical or spherical geometry. A value must be entered if subsequent data are to be entered |
| 5 | DY | Buckling height (cm) | BUCKLEDSLAB and BUCKLEDCYL | Appropriate dimension | Omit for SLAB, CYLINDRICAL, and SPHERICAL. This corresponds to one of the transverse dimensions of an actual 3-D slab assembly or to the length of a finite cylinder |
| 6 | DZ | Buckling depth (cm) | BUCKLEDSLAB | Appropriate dimension | Omit unless BUCKLEDSLAB was specified. This is the buckling depth corresponding to the second transverse dimension of a 3-D slab assembly |
| 7 | END | End geometry parameters | Always | END | Enter the word END. Do not start in column 1. At least two blanks must separate entry 7 from the first entry 8 |
| 8 | MXZ | Mixture number in the zone | Always | Mixture number | Repeat entry numbers 8-10 until all zones are defined. Enter the mixture number for this zone |
| 9 | RZ | Outside radius of the zone (cm) | Always | Appropriate dimension | Repeat entry numbers 8-10 until all zones are defined. Enter the outside dimension of the zone (cm) |
| 10 | XMOD | External moderator index | Optional | | Repeat entry numbers 8-10 until all zones are defined. Entry 10 is optional and can be omitted. If it is omitted, repeat entry numbers 8 and 9 until all zones are defined |
| | | | | NOEXTERMOD | No moderating materials in the adjacent zones |
| | | | | ONEEXTERMOD | A moderating material is present in one adjacent zone |
| | | | | TWOEXTERMOD | Moderating materials are present in two adjacent zones |
| | END ZONE | Terminate zone data | | END ZONE | Enter when all zones have been defined by repeating entry numbers 8 through 10 for each zone. At least two blanks must follow this entry |

Table C4.4.4 (continued)

Summary of available optional parameter data (see Sect. C4.4.8)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 1 | MORE DATA | Input flag | | This signals that optional parameter data will be entered. Enter only those parameters you wish to change |
| 2 | ISN= | Order of angular quadrature | XSDRNPM | The default value is 8. This allows using another value |
| 3 | SZF= | Spatial mesh size factor | XSDRNPM | The default value is 1.0. 0<SZF<1.0 gives a finer mesh. SZF>1.0 gives a coarser mesh |
| 4 | IIM= | Max. number of inner iterations | XSDRNPM | The default value is 20. This allows using another value |
| 5 | ICM= | Max. number of outer iterations | XSDRNPM | The default value is 25. This allows using another value |
| 6 | EPS= | Overall convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 7 | PTC= | Point convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 8 | BKL= | Buckling factor | XSDRNPM | The default value is 1.420892. Use ONLY for a multiregion problem that specifies BUCKLEDSLAB or BUCKLEDCYL |
| 9 | IUS= | Upscatter scaling factor | XSDRNPM | The default value is zero. IUS=0 doesn't utilize upscatter scaling. IUS=1 uses upscatter scaling to accelerate the solution and/or speed convergence |
| 10 | RES= | Resonance data | BONAMI NITAWL | Enter the mixture number, geometry type (SLAB, CYLINDER, SPHERE), and the thickness of the slab or radius of the sphere or cylinder in cm. Optionally enter the inner radius (cm) to specify an annular cylinder or sphere |
| 11 | DAN(mm)= | Dancoff factor for the specified mixture | BONAMI NITAWL | Enter the mixture number, mm, to which the Dancoff factor applies inside the parentheses; enter the Dancoff factor after the equal sign<br><br>**Repeat entry numbers 10 and 11 for all resonance mixtures used in the problem that are not treated in the LATTICECELL or MULTIREGION description** |

Table C4.4.4 (continued)

**Summary of available optional parameter data** (continued)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 12 | BAL= | Key to print balance tables | XSDRNPM | The default value is FINE. BAL=NONE suppresses printing the balance table. BAL=ALL prints all balance tables. BAL=FINE prints only the fine-group balance tables |
| 13 | DY= | First transverse dimension | XSDRNPM | The first transverse dimension, in cm, used in a buckling correction to calculate leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder) |
| 14 | DZ= | Second transverse dimension | XSDRNPM | The second transverse dimension, in cm, used for a buckling correction (i.e., the width of a slab) |
| 15 | COF= | Diffusion coefficient option for transverse leakage correction | XSDRNPM | The default is 0. See Sect. F3.5, 3$ array, variable IPN |
| 16 | FRD= | Unit from which fluxes will be read | XSDRNPM | Enter the unit number from which the flux guess for XSDRNPM will be read |
| 17 | FWR= | Unit on which fluxes will be written | XSDRNPM | Enter the unit number where the binary fluxes from XSDRNPM will be written |
| 18 | DAB= | Number of direct access data blocks | MIP[a] | The default is 200. Number of blocks allocated for direct access unit 90 |
| 19 | AXS= | Unit on which a mixed ANISN library will be written | ICE[a] | Enter the unit number where ICE is to write a mixed ANISN library |
| 20 | COLL | Key to activate collapse of thermal groups | MIP[a] | Enter COLL to collapse all thermal groups into one group for the shielding sequences |
| 21 | END | Terminus | | Terminate the optional parameter data. Do not start in column 1. At least two blanks must follow entry number 23 |

[a]MIP is the Material Information Processor.

Table C4.4.4 (continued)

**Outline of KENO data** (see Table C4.4.5 and Sects. C4.4.9 and F11.4 for details)

| Type of data | Starting flag | Comments | Termination flag |
|---|---|---|---|
| Title | None | 80 columns, must be entered first | None |
| Parameters | READ PARM | Enter desired parameter data | END PARM |
| Biasing or weighting | READ BIAS | Enter desired biasing data | END BIAS |
| Geometry | READ GEOM | Enter desired geometry data | END GEOM |
| Array data or unit orientation | READ ARRAY | Enter desired array data | END ARRAY |
| Boundary conditions or albedos | READ BNDS | Enter desired albedo data | END BNDS |
| Start data or initial source | READ START | Enter desired start data | END START |
| Mixing table data | READ MIXT | The Material Information Processor allows only NSCT= and PBXS= | END MIXT |
| Plot data | READ PLOT | Enter desired plot data | END PLOT |
| KENO V.a data terminus | END DATA | Enter to signal the end of all KENO V.a data | |

**Outline of SEARCH[a] data** (see Table C4.4.6 and Sect. C4.4.10 for details)

| | | |
|---|---|---|
| READ SEARCH | Enter desired search data | END SEARCH |

[a]Search data are required for CSAS4 and CSAS4X.

## Table C4.4.5 Summary of KENO V.a input data

### Summary of parameter data

TITLE: The title must be entered first (80 columns) See Sect. F11.4.3

PARAMETERS: Format: READ PARAM enter parameter data here END PARAM
If parameters are entered, they must follow the title. See Sects. F11.4.3, F11.5.2, and F11.5.3.

| KEY | STD. | DEFINITION | KEY | STD. | DEFINITION | KEY | STD. | DEFINITION | KEY | STD. | DEFINITION |
|---|---|---|---|---|---|---|---|---|---|---|---|
| RND= | Given | Random number | FLX= | NO | Fluxes | MKH= | NO | Matrix by hole | XSC= | 14 | Mixed xsecs |
| TME= | 30 min | Execution time (min) | FDN= | NO | Fission densities | CKH= | NO | Cofactor k by hole | ALB= | 79 | Albedo |
| TBA= | 0.5 min | Batch time (min) | ADJ= | NO | Adjoint calculation | FMH= | NO | Fiss. prod. by hole | WTS= | 80 | Weights |
| WTA= | 0.5 | Average weight | AMX= | NO | All mixture xsecs | HHL= | NO | MKH at highest level | LIB= | 4 | Working xsecs from NITAWL or XSDRN |
| WTH= | 3.0 | Wt. for splitting | XAP= | NO | Xsec angles & probs. | MKA= | NO | Matrix by array | | | |
| WTL= | 1/WTH | Russian Roulette wt. | XS1= | NO | 1-D xsecs | CKA= | NO | Cofactor k by array | | | |
| GEN= | 103 | No. of generations | XS2= | NO | 2-D xsecs | FMA= | NO | Fiss. prod. by array | SKT= | 16 | Scratch |
| NPG= | 300 | No. per generation | PKI= | NO | Fission spectrum | HAL= | NO | MKA at highest level | RST= | 95 | Read restart |
| NSK= | 3 | Generations skipped | P1D= | NO | Extra 1-D xsecs | BUG= | NO | Debug print | WRS= | 95 | Write restart |
| RES= | 0 | Gens. between restart | FAR= | NO | Fiss. & abs. | TRK= | NO | Print neutron tracks | XSC= | 14 | Mixed xsecs |
| NBK= | NPG+25 | Neutron bank | MKP= | NO | Matrix by location | PWT= | NO | Print avg. weight | | | |
| XNB= | 0 | positions | CKP= | NO | Cofactor k by loc. | PGM= | NO | Unprocessed geometry | | | |
| NFB= | NPG | Extra bank entries | FMP= | NO | Fiss. prod. by loc. | SMU= | NO | Self-multiplication | | | |
| XFB= | 0 | Fission bank positions | MKU= | NO | Matrix by unit | NUB= | YES | Neutrons per fission | | | |
| X1D= | 0 | Extra bank entries | CKU= | NO | Cofactor k by unit | PAX= | NO | Albedo-xsec array | | | |
| LNG= | 1000000 | No. of extra 1-Ds | FMU= | NO | Fiss. prod. by unit | RUN= | YES | Execute problem | | | |
| BEG= | 1 | Words of storage* | GAS= | FAR | FAR by energy group | PLT= | YES | Printer plots | | | |
| NB8= | 200 | Restart at this gen. | | | | | | | | | |
| NL8= | 512 | Blocks for d.a. unit<br>Length of d.a. block | | | | | | | | | |

*The words of storage in CSAS are defaulted to 100,000 and can be overridden by using the size= parameter on the analytical sequence specification which then becomes the default value of LNG in KENO V.a.
For example: =CSAS25 PARM=SIZE=200000
or =CSAS25 PARM=(CHECK,SIZE=200000)

# Table C4.4.5 (continued)

## Summary of array data

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sects. F11.4.5, F11.5.6, and F11.5.7.

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

| ARRAY PARAMETERS | | | DATA TYPE |
|---|---|---|---|
| KEYWORD | DEFAULT | DEFINITION | FILL<br>LOOP |
| ARA= | 1 | No. defining the array | |
| NUX= | 1 | No. of units in X direction | |
| NUY= | 1 | No. of units in Y direction | |
| NUZ= | 1 | No. of units in Z direction | |
| GBL= | Maxara | Global, universe, or overall array number* | |
| COM= | none | Delim comment delim optional comment is a maximum of 132 characters | |

*Can be defaulted by the code. If specified, it need be entered only once per problem.

### ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data utilizing the options in this table, the count field and option field must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

| COUNT FIELD | OPTION FIELD | OPERAND FIELD | COMMENTS |
|---|---|---|---|
| | | j | Stores j at the current position in the array |
| i | R | j | Stores j in the next i positions in the array |
| i | * | j | Stores j in the next i positions in the array |
| i | $ | j | Stores j in the next i positions in the array |
| | F | j | Fills remainder of the array with unit no. j starting with the current array position |
| | A | j | Sets the current position in the array to j |
| i | S | | increments; current position in the array by i allows skipping i positions. The value of i may be positive or negative |
| i | Q | j | Repeats the previous j entries i times. The default value of i is 1 |
| i | N | j | Repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1 |
| i | B | j | Starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1 |
| i | P | j | Alternately stores j and -j in the next i positions of the array |
| i | I | j k | Provides the end points, j and k, with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the I option should only be used to generate integer steps [i.e., (k-j)/(i+1) should be a whole number]. |
| | T | | Terminates the data reading for the array |

### ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP is terminated by entering END LOOP.

ENTER DATA IN THE FORM:

| DATA ENTRY | COMMENTS |
|---|---|
| LTYPE | The unit or box type. LTYPE must be greater than 1 |
| IX1 | Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX |
| IX2 | Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX |
| INCX | The number of units by which increments are made in the X direction |
| IY1 | The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY |
| IY2 | Ending position int he Y direction. IY2 must be at least 1 and no larger than the value of NUY |
| INCY | The number of units by which increments are made in the positive Y direction |
| IZ1 | Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ |
| IZ2 | Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ |
| INCZ | The number of units by which increments are made in the positive Z direction |

Table C4.4.5 (continued)

## Summary of biasing data

BIAS (weighting)

Format: READ BIAS keyword correlation data auxiliary END BIAS
See Sects. F11.4.7 and F11.5.8

| KEYWORD | DESCRIPTION | | MATERIAL | ID | ENERGY GROUPS | THICKNESS/ INCREMENT |
|---|---|---|---|---|---|---|
| ID= | CORRELATION DATA will be read next. | | | | | |
| | id | Material ID. Enter from table at right to use weighting data from the library | Concrete | 301 | 16,27,44,218,238 | 5 cm |
| | | | Paraffin | 400 | 16,27,44,218,238 | 3 cm |
| | ibgn | Beginning bias ID | Water | 500 | 16,27,44,218,238 | 3 cm |
| | iend | Ending bias ID | Graphite | 6100 | 16,27,44,218,238 | 20 cm |
| WT= | AUXILIARY DATA will be read next. | | | | | |
| WTS= | AUXILIARY DATA will be read next. | | | | | |
| | wttitl | Material title (12-character maximum) | | | | |
| | id | Material ID | | | | |
| | nsets | Number of sets of group structures | | | | |
| | REPEAT | (thkinc, numinc, ngpwt, wtavg) nsets times | | | | |
| | thkinc | Thickness per increment | | | | |
| | numinc | Number of increments | | | | |
| | ngpwt | Number of energy groups for this set of wts | | | | |
| | wtavg | Enter numinc x ngpwt values of wtavg | | | | |

For CORRELATION DATA, the material ID is chosen from material ID column above (the keyword is ID=).
For AUXILIARY DATA, the material ID is chosen by the user and the keyword is WT= or WTS=.
When AUXILIARY DATA are entered, CORRELATION DATA must also be entered to use the data.
Beginning and ending bias IDs are defined by the user. The geometry specification that has the bias ID equal to the beginning bias ID utilizes the wtavg's from the first interval of material ID.

## Table C4.4.5 (continued)

### Summary of boundary condition data

BNDS (albedo or boundary conditions)

Format: READ BNDS face code albedo name END BNDS
See Sect. F11.4.7

The sequence FACE CODE ALBEDO NAME is entered as many times as necessary to define the appropriate albedo boundary conditions. The default for all faces is vacuum.

FACE CODES FOR ENTERING BOUNDARY (ALBEDO) CONDITIONS

| FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION |
|---|---|---|---|---|---|---|---|
| +XB= | Positive X face | XFC= | Both X faces | +YX= | Positive X and Y faces | &ZY= | Positive Y and Z faces |
| &XB= | Positive X face | YFC= | Both Y faces | &XY= | Positive X and Y faces | -XY= | Negative X and Y faces |
| -XB= | Negative X face | ZFC= | Both Z faces | &YX= | Positive X and Y faces | =XZ= | Negative X and Z faces |
| +YB= | Positive Y face | +FC= | All positive faces | +XZ= | Positive X and Z faces | =YZ= | Negative Y and Z faces |
| &YB= | Positive Y face | &FC= | All positive faces | +ZX= | Positive X and Z faces | YXF= | All X and Y faces |
| -YB= | Negative Y face | -FC= | All negative faces | &XZ= | Positive X and Z faces | ZXF= | All X and Z faces |
| +ZB= | Positive Z face | XYF= | All X and Y faces | &ZX= | Positive X and Z faces | ZYF= | All Y and Z faces |
| &ZB= | Positive Z face | XZF= | All X and Z faces | +YZ= | Positive Y and Z faces | -YX= | Negative X and Y faces |
| -ZB= | Negative Z face | YZF= | All Y and Z faces | +ZY= | Positive Y and Z faces | -ZX= | Negative X and Z faces |
| ALL= | All 6 faces | +XY= | Positive X and Y faces | &YZ= | Positive Y and Z faces | -ZY= | Negative Y and Z faces |

ALBEDO NAMES AVAILABLE ON THE KENO V ALBEDO LIBRARY, FOR USE WITH THE FACE CODES

| ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION |
|---|---|---|---|---|---|
| DPOH20<br>DPOH20<br>DPO<br>DPO | 12-in. double PO water differential albedo with 4 incident angles | CONC-4<br>CON4<br>CONC4 | 4-in. concrete differential albedo with 4 incident angles | VACUUM<br>VOID<br>VACU<br>VAC | Vacuum condition |
| H20<br>WATER | 12-in. water differential albedo with 4 incident angles | CONC-8<br>CON8<br>CONC8 | 8-in. concrete differential albedo with 4 incident angles | SPECULAR<br>MIRROR<br>MIRR<br>SPEC<br>SPE<br>MIR | Mirror image reflection |
| PARAFFIN<br>PARA<br>WAX | 12-in. paraffin differential albedo with 4 incident angles | CONC-12<br>CON12<br>CONC12 | 12-in. concrete differential albedo with 4 incident angles | PERIODIC<br>PERI<br>PER | Periodic boundary condition |
| CARBON<br>GRAPHITE<br>C | 200-cm carbon differential albedo with 4 incident angles | CONC-16<br>CON16<br>CONC16 | 16-in. concrete differential albedo with 4 incident angles | | |
| ETHYLENE<br>POLY<br>CH2 | 12-in. polyethylene differential albedo with 4 incident angles | CONC-24<br>CON24<br>CONC24 | 24-in. concrete differential albedo with 4 incident angles | | |

## Table C4.4.5 (continued)

**Summary of geometry data**

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM
See Sects. F11.4.4, F11.5.1.2, F11.5.6, and F11.5.7.

GEOMETRY REGION DATA consists of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.

ENTER GEOMETRY REGION DATA IN THE FOLLOWING FORM:

OPTIONAL GLOBAL SPECIFICATION
UNIT n
OPTIONAL GEOMETRY COMMENT
SIMPLE GEOMETRY REGION DATA
and/or
EXTENDED GEOMETRY REGION DATA

* * * \

Table C4.4.5 (continued)

**Summary of geometry data (cont.)**

GEOMETRY (region) (cont.)

ENTER EXTENDED GEOMETRY DATA IN THE FOLLOWING FORM:

fgeom ref. ID bias ID thickness per region origin coordinates nreg

EXTENDED GEOMETRY REGION INPUT DATA REQUIREMENTS

| TYPE OF DATA | TYPE 1 DATA | TYPE 2 DATA | TYPE 3 DATA |
|---|---|---|---|
| fgeom | ARRAY<br>CORE<br>COREBDY<br>COREBNDS<br>COREBOUN | HOLE | REPLICATE<br>REFLECTOR |
| Ref. ID | Array no. | Emplaced unit No. | Mixture No. in generated regions |
| Bias ID | Omit for ARRAY | Omit | First bias ID |
| Thick./reg. | Omit | Omit | Variable* |
| Origin coord. | Enter the X Y Z coord. of most neg. pt. of array | Enter the X Y Z coord. of origin | Omit |
| nreg | Omit | Omit | No. of regions to be generated |

*The number of dimensions to be entered is the same as the region preceding the replicate or reflector specification because the generated regions have that shape. The value of the dimensions is the thickness of each generated region of material on that surface.

Table C4.4.5 (continued)

**Summary of mixing table data**

MIXTURES — Format: READ MIXT xsec parameters END MIXT

Note: In CSAS, only xsec parameters should be entered.
KENO V.a mixing table data consists of "xsec parameters" and a "mixing table." The "mixing" table" is used to define the materials used in the KENO V.a problem. This data is automatically created by CSAS and provided to KENO V.a. It cannot be overridden.
See Sects. F11.4.10 and F11.5.5.

XSEC PARAMETERS — consists of keywords and associated values.
These parameters, if entered, need be entered only once.

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| SCT= | 1 | No. of discrete scattering angles<br>0 is isotropic<br>1 is P1<br>2 is P3<br>3 is P5 |
| EPS= | 0.00003 | Cross-section message cutoff value; use to suppress message No. K5-60 |

Table C4.4.5 (continued)

## Summary of plot data

PLOT Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made.
See Sects. F11.4.1 and F11.5.9

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| TTL= | Prob. title | delim ptitl delim delim is a one-character delimiter that signals the beginning and end of the title. ptitl is the plot title (max. 132 char.) |
| PIC= | MAT | Type of picture: MIXTURE, UNIT NO. or BIAS ID NO. |
| | | MIXTURE ---------- MAT |
| | | MIX |
| | | MIXT |
| | | MIXTURE |
| | | MEDI |
| | | MEDIA |
| | | UNIT NO. ----------BOX |
| | | BOXT |
| | | BOXTYPE |
| | | UNT |
| | | UNIT |
| | | UNITTYPE |
| | | BIAS ID NO. ------- IMP |
| | | BIAS |
| | | BIASID |
| | | WTS |
| | | WEIG |
| | | WEIGHTS |
| | | WGT |
| | | WGTS |
| XUL= | Prev. plot | X coord. of upper left corner of plot |
| YUL= | Prev. plot | Y coord. of upper left corner of plot |
| ZUL= | Prev. plot | Z coord. of upper left corner of plot |
| XLR= | Prev. plot | X coord. of lower right corner of plot |
| YLR= | Prev. plot | Y coord. of lower right corner of plot |
| ZLR= | Prev. plot | Z coord. of lower right corner of plot |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| UAX= | Prev. plot 0 IF VAX OR WAX is read | X component of direction cosine for the AX axis of the plot (across) |
| VAX= | Prev. plot 0 IF UAX OR WAX is read | Y component of direction cosine for the AX axis of the plot (across) |
| WAX= | Prev. plot 0 IF UAX OR VAX is read | Z component of direction cosine for the AX axis of the plot (across) |
| UDN= | Prev. plot 0 IF VDN OR WDN is read | X component of direction cosine for the DN axis of the plot (down) |
| VDN= | Prev. plot 0 IF UDN OR WDN is read | Y component of direction cosine for the DN axis of the plot (down) |
| WDN | Prev. plot 0 IF UDN OR VDN is read | Z component of direction cosine for the DN axis of the plot (down) |
| DLX= | | Horizontal spacing between points on plot |
| DLD= | | Vertical spacing between points on plot |
| NAX= | | No. of intervals to be printed across page |
| NDN= | | No. of intervals to be printed down page |
| LPI= | 8 | Lines per inch printed down the page |
| NCH= | CHRS* | Delim CHRS delim a one-character delimiter signals the beginning and end of the character string |
| RUN= | YES | YES allows the problem to execute NO terminates problem after data checking |
| PLT= | YES | YES allows the plot(s) to be made NO allows reading the plot data without making a plot |
| SCR= | NO | NO specifies a printer plot. If YES, a .gif file is generated for each plot |
| CLR= | | Use only with SCR=YES to change colors. Input 4 integers: 1st is index into color table, next 3 are red, green, and blue color values |

PLOT ORIGIN:

(1) SINGLE UNIT - coincides with origin of geometry description.

(2) BASE ARRAY - at the most negative point of the array (lower left-hand back corner of the global array).

(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

*Default values of CHRS are given below:

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| CHRS | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F | G | H | I | J | K | L | M |

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 |
| CHRS | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | # | , | $ | - | + | ) | \| |

| | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 |
| CHRS | £ | > | : | ; | . | - | % | * | " | = | ! | ( | @ | < | / | 0 |

## Table C4.4.5 (continued)

### Summary of starting data

START Format: READ START enter start data here END START
The default value of start type is zero. See Sect. F11.4.8.

| START TYPE | REQUIRED DATA | OPTIONAL DATA | STARTING DISTRIBUTION |
|---|---|---|---|
| 0 | None | NST | Uniform |
| | | XSM | |
| | | XSP | |
| | | YSM | |
| | | YSP | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 1 | NST | XSM | Cosine |
| | | XSP | |
| | | YSM | |
| | | YSP | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 2 | NST | XSM | Cosine with |
| | NXS | XSP | fraction in |
| | NYS | YSM | specified |
| | NZS | YSP | unit |
| | FCT | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 3 | NST | KFS | Spike |
| | TFX | PSP | |
| | TFY | | |
| | TFZ | | |
| | NXS | | |
| | NYS | | |
| | NZS | | |
| 4 | NST | KFS | Multiple |
| | TFX | PSP | spikes |
| | TFY | | |
| | TFZ | | |
| | NBX | | |
| 5 | NST | PSP | In specified |
| | NBX | | units |
| 6 | NST | NXS | Arbitrary |
| | TFX | NYS | points |
| | TFY | NZS | |
| | TFZ | KFS | |
| | LNU* | PS6 | |
| | | PSP | |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| NST= | 0 | Start type |
| TFX= | 0.0 | X coordinate |
| TFY= | 0.0 | Y coordinate |
| TFZ= | 0.0 | Z coordinate |
| NXS= | 0 | X index of unit pos. |
| NYS= | 0 | Y index of unit pos. |
| NZS= | 0 | Z index of unit pos. |
| KFS= | | Fission spectra |
| LNU= | 0 | Number of last neutron |
| NBX= | 0 | Source unit number |
| FCT= | 0 | Fraction |
| XSM= | -X | -X of source cuboid |
| XSP= | +X | +X of source cuboid |
| YSM= | -Y | -Y of source cuboid |
| YSP= | +Y | +Y of source cuboid |
| ZSM= | -Z | -Z of source cuboid |
| ZSP= | +Z | +Z of source cuboid |
| RFL= | NO | Start in reflector |
| PS6= | NO | Print start 6 input |
| PSP= | NO | Print starting points |

*LNU must be the last entry for each set of start 6 data. The LNU of each successive set of data must be larger than the last.

additional information contained in Sect. F11.5. Table C4.4.6 contains a summary of the SEARCH data for CSAS4 and CSAS4X.

The Material Information Processor now allows terse data input for all data except the standard composition names and the keyword entries in the optional parameter data (MORE DATA). Terse data entry requires the user to enter only as many characters as is necessary to uniquely define the data entry. However, as many characters as are specified by the user must match the name allowed by the code.

For example, refer to Table C4.4.4, **Outline of Material Information Processor data**, data position 3. The available choices for the type of calculation are: INFHOMMEDIUM, LATTICECELL, and MULTIREGION. Therefore, I would be sufficient to specify an infinite homogeneous medium, L would be sufficient for lattice cell, and M would specify multiregion. If it is desirable to use more characters, the user is free to do so. However, as many characters as are entered must match the name allowed by the code. For example, if INFHOMMEDIUM is to be specified, INFH is correct, but INFI is not.

WARNING: If several choices begin with the same letters, sufficient characters should be entered to uniquely define the name. For example, ASPHSQUAREP and ASPHTRIANGP are identical through the first four characters. If only ASPH is entered, the code will accept the first name whose first four characters are ASPH. This may not be the one the user intended to use.

### C4.4.2 INPUT DATA REQUIREMENTS AND DATA CHECKING

Table C4.4.7 summarizes the input data required by Criticality Safety Analysis Sequence No. 4. The types of data needed are:

1. Analytical sequence specification. Enter the desired sequence name, starting in column 1. The available options are =CSAS4, =CSAS4X, =CSAS25, =CSAS2X, =CSAS1X, =CSASI, =CSASIX, and =CSASN. Enter PARM=CHK in column 11 to check the data.

2. Material Information Processor data. These data are required for processing the cross sections for use in the problem. These data are described in detail in Sects. M7.4 and C4.4.3 through C4.4.8.

3. KENO V.a data. These data are described briefly in Sect. C4.4.9. A detailed description is found in Sect. F11.4 with additional information in Sect. F11.5.

4. Search data. These data are required for CSAS4 and CSAS4X. A detailed description is given in Sect. C4.4.10. The data required for a simple optimum pitch search are

   READ SEARCH OPTIMUM PITCH END SEARCH

5. END for analytical sequence—Enter END, starting in column 1.

To check the input data, specify the desired analytical sequence and specify PARM=CHECK or PARM=CHK starting beyond column 10. For example, to check CSAS data, the analytical sequence specification would be:

=CSAS4 PARM=CHECK

Table C4.4.6 Summary of SEARCH input data

Outline of search type specification (see Sect. C4.4.10.1 for details)

| Entry number | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | Search descriptor | OPTIMUM<br>CRITICAL<br>MINIMUM | Initiates a search for the maximum value of k-effective<br>Initiates a search for a specified value of k-effective.<br>Initiates a search for the minimum value of k-effective |
| 2 | Search type | PITCH<br>DIMENSION<br>CONCENTRATION | Vary the pitch<br>Vary one or more dimensions in one or more regions<br>Vary the concentration of one or more standard compositions in one or more mixtures |
| 3 | Optional search parameters | | Optional search parameters allow changing default values. Any or all may be entered in any order |
| 3a | No. of search passes | PAS= | Enter the keyword PAS= followed by the desired number of search passes. default=10 |
| 3b | No. of search parameters | NPM= | Enter the keyword NPM= followed by the number of search parameters. Present capability is limited to 1. default=1 |
| 3c | Search convergence tolerance | EPS= | Enter the keyword EPS= followed by the desired convergence tolerance Default=0.005 |
| 3d | Desired value of k-effective | KEF= | Enter the keyword KEF= followed by the desired value of k-effective. The default value is 1.000. DO NOT ENTER FOR OPTIMUM OR MINIMUM SEARCHES |
| 3e | Maximum allowed pitch | MAXPITCH= | Enter the keyword MAXPITCH= followed by the maximum allowed pitch for a search whose search type, entry 2 above, is PITCH. The default value is the pitch corresponding to -5.0 times the parameter at the minimum possible pitch |
| 3f | Minimum allowed pitch | MINPITCH= | Enter the keyword MINPITCH= followed by the minimum allowed pitch for a search whose search type, entry 2 above, is PITCH. The default value is the minimum possible pitch (i.e., the pitch at which the shapes in the array touch) |
| 4 | Additional search data | MORE | Enter the delimiter MORE. This delimiter ends the optional search commands and initiates the auxiliary search commands found in Table C4.4.17. |

Table C4.4.6 (continued)
**Outline of auxiliary search commands and constraints**

| Entry number | Keyword name | Type of Data | Comments |
|---|---|---|---|
| GENERIC SEARCH DATA | | | -May be used with all Search Types- |
| 1a | ALTER<br>CHANGE<br>MODIFY | Begin a new search command | These words are used to specify that modifications will be made to the geometry or concentration according to subsequent commands (entries 3 through 6 as required to specify the desired changes) |
| 1b | MAINTAIN | Begin a new search command | The spacing (thickness) of the specified geometry regions will be maintained when the interior regions grow or shrink |
| 1c | KEEP<br>HOLD | Begin a new search command | This command resets the specified geometry to the original input specifications |
| 2 | PAR= | Parameter number | Enter the parameter number that the current command (ALTER, MAINTAIN, KEEP) applies to. The present capability is limited. Default=1 and should not be changed. |
| 3 | +CON= | Maximum constraint | Enter the maximum constraint for the current parameter. default=+10E10 |
| 4 | -CON= | Minimum constraint | Enter the minimum constraint for the current parameter. default=-10E10 |
| PITCH & DIMENSION SEARCH DATA | | | -Defines Geometric Changes- |
| 5 | UNIT= | Unit the current command applies to | Enter the unit in which regions are to be altered |
| 6 | REGION= | First region to be altered in the unit | Enter the first or only region in the unit that the search constants (entry 8a, b, c and/or d) apply to. Default is the first region |
| 7 | TO | Last region to be altered in the unit | Enter the last region in the unit that the search constants (entry 8a, b, c and/or d) apply to. Default is the first region<br>NOTE: Entry 6 must be entered in order to alter a single region in a unit. Entries 6 and 7 must both be entered in order to alter more than one region in a unit |
| 8 | ALL= | Search constant for all surfaces (faces) of the region(s) | Enter a value for the search constants for the specified regions. This value will be applied to all surfaces of the region(s) |
| 8a | +X= | Search constant for +X face of cuboid | Enter a value for the search constant for the +X face of a cuboid |
| | -X= | Search constant for -X face of cuboid | Enter a value for the search constant for the -X face of a cuboid |
| | +Y= | Search constant for +Y face of cuboid | Enter a value for the search constant for the +Y face of a cuboid |
| | -Y= | Search constant for -Y face of cuboid | Enter a value for the search constant for the -Y face of a cuboid |
| | +Z= | Search constant for +Z face of cuboid | Enter a value for the search constant for the +Z face of a cuboid |
| | -Z= | Search constant for -Z face of cuboid | Enter a value for the search constant for the -Z face of a cuboid |
| 8b | RADIUS= | Search constant for radius | Enter a value for the search constant for the radius of a sphere or a cylinder |
| 8c | +HEIGHT= | Search constant for + height | Enter a value for the search constant for the + height of a cylinder |
| | -HEIGHT= | Search constant for - height | Enter a value for the search constant for the - height of a cylinder |
| 8d | CHORD= | Search constant for chord | Enter a value for the search constant for the chord face of a hemisphere or hemicylinder |
| CONCENTRATION SEARCH DATA | | | -Define concentration changes- |
| 9 | MIX= | Search constant for mixture | Enter the mixture number containing the standard composition to be changed. |
| 10 | SCNAME= | Search constant for Standard Composition | Enter the standard composition name whose density is to be changed. |
| 11 | FACTOR= | Search proportionality constant | Enter the value of the search constant for the concentration search. |

Table C4.4.7 Summary of data input requirements

| Type of data | Analytical sequence | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 Analytical sequence specification | =CSASN | =CSAS1X | =CSASI | =CSASIX | =CSAS25 | =CSAS2X | =CSAS4 | =CSAS4X |
| 2 Material Information Processor data | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| 3 KENO V.a data | No | No | No | No | Yes | Yes | Yes | Yes |
| 4 Search data | No | No | No | No | No | No | Yes | Yes |
| 5 END for the analytical sequence | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |

Note that =CSAS4 starts in column 1 and PARM=CHECK starts beyond column 10. This causes the input data to be checked, the problem description to be printed, appropriate warning and error messages to be printed, and if KENO V.a data are entered, the data and printer plots are printed.

This feature allows checking the input data without executing any of the functional modules. Simply remove the PARM=CHECK to actually execute the problems.

## C4.4.3 MATERIAL INFORMATION PROCESSOR DATA

The Material Information Processor reads the standard compositions specification data and the unit cell geometry specifications. It then produces the mixing table and unit cell information necessary for processing the cross sections. Table C4.4.8 outlines the input data for the Material Information Processor.

The input data for the Material Information Processor are identical to the input for CSAS1 of SCALE 3.1 with the following exceptions:

- The analytical sequence card is considered to be part of the overall data input.
- Items 2, 3, 4, 6, and 7 from Sect. C1.6 are omitted from the parameter card.
- The words END COMP are entered to signal the completion of the standard compositions specification data.
- The MULTIREGION zone description data are terminated by the words END ZONE.
- If optional control parameters are entered, the words MORE DATA must precede the first optional parameter data entry.

Note that CSAS1 is not included in this SCALE release.

The Material Information Processor allows terse data input for all data except the standard composition names and the keyword entries in the optional parameter data (MORE DATA). Terse data entry requires the user to enter only as many characters as are necessary to uniquely define the data entry. However, as many characters as are specified by the user must match the name allowed by the code. For example, refer to Table C4.4.8, **Outline of Material Information Processor data**, data position 3. The available choices for the type of

Table C4.4.8. Outline of Material Information Processor data

| Data position | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | Title | Enter a title | 80 characters |
| 2 | Cross-Section Library Name | HANSEN-ROACH<br>27GROUPNDF4<br>44GROUPNDF5<br>238GROUPNDF5<br>27BURNUPLIB<br>etc. | The currently available libraries are listed in Sect. M4. |
| 3 | Type of Calculation | INFHOMMEDIUM<br>LATTICECELL<br>MULTIREGION | These are the available options. See the explanation in Sect. C4.4.3 |
| 4 | Standard Composition Specification Data | Enter the appropriate data | Terminate this data block with END COMP. See Sect. C4.4.4 |
| 5 | Unit Cell Geometry Specification | Enter the appropriate data (optional for INFHOMMEDIUM) | Optional for INFHOMMEDIUM<br>See Sect. C4.4.5 for INFHOMMEDIUM<br>See Sect. C4.4.6 for LATTICECELL<br>See Sect. C4.4.7 for MULTIREGION |
| 6 | Optional Parameter Data | Enter the desired data | Precede this data block by MORE DATA if more parameter data are to be entered. Otherwise, omit these data entirely. See Sect. C4.4.8 |

are INFHOMMEDIUM, LATTICECELL, and MULTIREGION. Therefore, it would be sufficient to specify I for an infinite homogeneous medium, L for lattice cell, and M for multiregion. If it is desirable to use more characters, the user is free to do so. However, as many characters as are entered must match the name allowed by the code. For example, if INFHOMMEDIUM is to be specified, INFH is correct, but INFI is not.

WARNING: If several choices begin with the same letters, sufficient characters should be entered to uniquely define the name. For example, ASPHSQUAREP and ASPHTRIANGP are identical through the first four characters. If only ASPH is entered, the code will accept the first name whose first four characters are ASPH. This may not be the one the user intended to use.

The types of data required for the Material Information Processor are given in Table C4.4.8. The individual entries are explained in detail in the text following the table.

1. Title. An 80-character title is required. The title is the first 80 characters of the Material Information Processor data.

2. Cross-Section Library Name. This item specifies the cross-section library that is to be used in the calculation. See Table C4.4.8 or Sect. M4 for a list of the available libraries.

3. Type of Calculation. The options are INFHOMMEDIUM, LATTICECELL, and MULTIREGION.

   INFHOMMEDIUM is used for an infinite homogeneous medium. It provides cross sections to represent

large homogeneous pieces. The cross sections are treated as if each mixture is infinite. Thus, the self-shielding calculations will not account for any geometrical effects. By default, the smallest mixture number will be used in the cell. A different mixture can be used as described in Sect. C4.4.5 and Table C4.4.10.

LATTICECELL is used when the geometry can be described as a lattice. It is especially suited for arrays of cylindrical rods or spherical pellets. The use of LATTICECELL requires the entry of additional geometry information (unit cell specification), which is described in Sect. C4.4.6 and Table C4.4.11. When the problem consists of an array of fuel bundles or similar geometry, the cross sections are corrected for resonance self-shielding, including geometry effects. If cell-weighted cross sections are desired, CSAS2X, CSAS1X, CSASIX, or CSAS4X must be used. A flux disadvantage factor is then applied to the cross sections of the materials that are used in the unit cell. The KENO V.a geometry description may represent a fuel bundle discretely or as a homogeneous region. By using CSAS2X or CSAS4X, a fuel bundle can be represented as a single homogeneous region by specifying a mixture number of 500 on the KENO V.a geometry card that defines the overall size and shape of the bundle. This utilizes the cell-weighted cross sections that represent the spatial behavior of the unit cell. The use of CSAS2X or CSAS4X precludes using individual mixtures from the unit cell in the KENO V.a geometry.

MULTIREGION may be used to define a geometric configuration that is more complicated than that allowed by LATTICECELL. It can also be used for a system involving large geometric regions where geometry effects may be minimal. The cross sections utilized in the unit cell are corrected for resonance self-shielding, and an appropriate Dancoff factor is applied (see Sect. M7.2.4.3). If a cell-weighted mixture is generated, it is derived from homogenizing the entire MULTIREGION unit cell. The additional geometry information (unit cell specification) required for a MULTIREGION problem is described in Sect. C4.4.7 and Table C4.4.12.

4. Standard Compositions. These data are used to define the mixtures that will be used in the problem. See Sect. C4.4.4, and Table C4.4.9, for a description of the standard compositions specification data. These data are required for every problem.

5. Cell Geometry Specificaiton. See Sect. C4.4.5 and Table C4.4.10 for an explanation of the optional unit cell data associated with an INFHOMMEDIUM problem. See Sect. C4.4.6 and Table C4.4.11 for an explanation of the data associated with LATTICECELL problems. Section C4.4.7 and Table C4.4.12 explain the data required for a MULTIREGION problem.

6. Optional Parameter Data. This option allows certain defaulted parameters to be respecified by the user. Omit these data unless they are needed. If data are to be entered, see Sect. C4.4.8 and Table C4.4.13 for assistance.

## C4.4.4 STANDARD COMPOSITIONS SPECIFICATION DATA

The standard compositions specification data are used to define mixtures that are to be utilized in the problem. These data can be considered to be a mixing table that defines mixtures in terms of standard compositions specification data (available from Sect. M8.2). The required input for the standard

Table C4.4.9 Outline of standard composition specification data

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 1 | SC | Standard composition component name | Always | Enter once for each standard composition. Enter the alphanumeric description from Sect. M8.2. Additional allowed names include those beginning with ARBM for arbitrary materials and SOLN for solutions |
| A1 | ROTH | Theoretical density of material (g/cc) | ARBM | Enter once for each arbitrary material |
| A2 | NEL | Number of elements in the material | ARBM | Enter once for each arbitrary material |
| A3 | IVIS | No longer used but must still be entered | ARBM | Enter once for each arbitrary material. Enter 0 or 1 |
| A4 | ICP | Compound indicator | ARBM | Enter once for each arbitrary material. Enter 1 for a chemical compound, 0 for alloys, mixtures, etc. |
| A5 | IRS | No longer used but must still be entered | ARBM | Enter once for each arbitrary material. Enter 0 or 1 |
| A6 | NCZA | ID number (from far right column of Table M8.2.1) | ARBM | Repeat the sequence A6 and A7 for each element or isotope in the arbitrary material before entering entry 2. Enter the number from the far right column of Table M8.2.1 or M8.2.2. (Premixed standard compositions cannot be used in an arbitrary material definition) |
| A7 | ATPM | Number of atoms of this element per molecule of arbitrary material<br>or<br>weight percent of this element in this arbitrary material | ARBM and ICP=1<br>or<br>ARBM and ICP=0 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry 2. Do not enter a value unless ICP=1<br>or<br>repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry 2. Do not enter a value unless ICP=0 |

Table C4.4.9 (continued)

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 2 | MX | Mixture ID number | Always | Enter once for each standard composition component |
| S1 | FD | Fuel density (grams of U or Pu per liter of solution) | SOLN | Enter once for a solution |
| S2 | AML | Acid molarity of the solution | SOLN | Enter once for a solution. AML=0 if there is no acid in the solution |
| O1 | SPGR<br>or<br>ROTH | Specific gravity of the solution<br>or<br>density of the basic standard composition | Optional | If the specific gravity (SPGR) of the solution is known, it should be entered as SPG=SPGR<br>or<br>if the density of a basic standard composition (ROTH) is to be entered, use DEN=ROTH |
| 3 | VF | Density multiplier | See comment column | Enter the density multiplier (density fraction, volume fraction, or a combination). Default value is 1. This item can be omitted if entries 4, 5, 6a, and 6b are also omitted. VF=0 is not allowed for SOLN or ARBM |
| 4 | ADEN | Number density (atoms/b-cm) for the nuclide | VF=0 | Enter only if VF=0.0 |
| 5 | TEMP | Temperature, in K | See comment column | Default value is 293 K. This entry can be omitted if entries 6a and 6b are also omitted |
| 6a | IZA | Isotope's ZA number | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 6b | WTP | Weight percent of the isotope | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 7[a] | END | Terminate a standard composition | Always | Enter once for each standard composition component. This terminates the data for a standard composition component. Enter END to terminate the component. Repeat entries 1 through 7 until all the mixtures have been defined. At least two blanks must separate entry 7 from the next entry |
| | END COMP | Terminate the data block | Terminus | Enter once for a problem. Enter the words END COMP when all the standard composition components have been described. At least two blanks must follow the END COMP |

[a]NOTE: Entry 7 should not begin in column 1 unless a name is associated with it. At least two blanks should separate the last entry 7 from the END COMP.

compositions specification data varies, depending on the standard composition component name. However, every standard composition specification must include the following:

1. a standard composition component name (SC),

2. the mixture that contains this component (MX),

3. a terminator for the standard compositions specification data (enter the word END).

Enter as many standard composition specifications as necessary to define all the mixtures needed for the problem. When all the necessary mixtures have been described, enter the words END COMP to signal the completion of this type of data.

The data required to define a standard composition specification are contained in Table C4.4.9. In the column entitled "Entry requirement," "ARBM" is used to denote entries required only for arbitrary materials, and "SOLN" is used for entries required only for solutions.

The input data for standard composition specifications are arranged with entry numbers 1 through 7 that apply to all standard composition data. Entries A1 through A7 apply only to arbitrary materials. Entries S1 and S2 apply only to solutions. Entry O1 is optional data that applies to both basic standard compositions and solutions. The individual entries specified in Table C4.4.9 are explained in detail in the text accompanying the table.

1. SC Standard Composition Component Name. The names of the standard composition components (the alphanumeric identifiers) may be (1) chosen from Sect. M8.2, which contains the lists of elements, compounds, and alloys found in the Standard Composition Library, (2) chosen from the table of available solutions, Sect. M8.3, or (3) designated as an arbitrary material (ARBM). An error message will be printed if the user enters an invalid standard composition component name.

   For standard compositions taken from Sect. M8.2 of the Standard Composition Library (basic standard compositions), certain data such as the volume fraction or theoretical density and other engineering type data may need to be entered. For standard compositions containing more than one isotope of an element (such as $UO_2$), the user is free to specify the weight percent for each isotope such that they total 100%. See Sect. M7.5.5.1 for examples of basic standard compositions.

   Solutions require the specification of the density of the heavy metal in the solution and the acid molarity of the solution. The alphanumeric specification of a solution starts with the four characters "SOLN." Some common fissile solutions are SOLNUO2(NO3)2, SOLNPU(NO3)4, and SOLNUO2F2. See Sect. M8.3 for a complete listing of the available solutions. Sections M7.5.5.3 and M7.5.9 contain examples of the input data for solutions.

   Arbitrary materials require the user to provide all the information normally found in the Standard Composition Library. The arbitrary material option allows specifying a compound or alloy not available in the Standard Composition Library by utilizing the standard compositions available in the library.

   An arbitrary material name must start with the four characters "ARBM." A maximum of twelve characters is allowed for the standard composition component name. Imbedded blanks are not allowed in a standard composition component name. Section M7.5.5.2 contains examples of input data for arbitrary materials.

An additional example of an arbitrary material specification is as follows:

Consider a mixture of boral having 35 wt % $B_4C$, 65 wt % aluminum and an overall density of 2.64 g/cc. If B4C were not available in the Standard Composition Library, the components BORON, C, and AL could be used as basic standard compositions to describe the boral. The necessary volume fractions would have to be determined as shown in Sect. M7.5.5.4, examples 3 and 4. By utilizing the arbitrary material option, both $B_4C$ and aluminum can be described as a component of the same mixture (assume it is desired to be mixture 3), both have a density of 2.64 g/cc, and the volume fractions are 0.35 for the $B_4C$ and 0.65 for the aluminum. The resulting standard compositions specification data do not require any calculations (see example 1, Sect. M7.5.5.5 for a similar situation) and could be entered as

```
ARBMTL-B4C 2.64 2 1 1 0 5000 4 6012 1 3 0.35 END
ARBMTL-AL  2.64 1 0 0 0 13027 100.0 3 0.65 END
```

A1. ROTH Theoretical density of material. Enter ONLY for arbitrary materials. The density of the arbitrary material is entered in units of grams per cubic centimeter.
NOTE: ROTH can be entered for a basic standard composition as shown by entry O1.

A2. NEL Number of elements in the material. Enter ONLY for arbitrary materials. Enter the number of components from the Standard Composition Library that are to be used to define this arbitrary material. In the above example, the ARBMTL-B4C contains two components, the B and the C. Therefore, NEL is 2 for the $B_4C$ arbitrary material. The ARBMTL-AL contains only one component; so NEL is 1 for it.

A3. IVIS NO LONGER USED BUT MUST STILL BE ENTERED...Enter ONLY for an arbitrary material. Enter 0 or 1.

A4. ICP Compound indicator. Enter ONLY for arbitrary materials. A compound is a substance containing more than one constituent element and having properties different from those which their constituents had as elementary substances. The composition of a pure compound is perfectly definite and is always the same, regardless of the method of formation. Examples of compounds are $H_2O$, $B_4C$, $CO_2$, $Ca(OH)_2$, $C_2H_6$, $Fe_2S_3$, $Cr_2O_3$, etc. ICP is 1 for a compound. ICP is 0 for alloys (stainless steel, monel, etc.) and conglomerates or mixtures such as concrete.

In the $B_4C$ arbitrary material example, ICP is 1; in the aluminum arbitrary material example, ICP is 0. These examples are given in the explanation of entry number 1, SC.

A5. IRS UNUSED VARIABLE...Enter ONLY for arbitrary materials. Enter 0 or 1.

A6. NCZA ID number. Enter ONLY for arbitrary materials. This is the "ZA" ID number for the element or isotope. Usually, NCZA=A+1000*Z, where A is the atomic mass or weight of the nuclide, and Z is the atomic number. For example, NCZA is 1001 for hydrogen, 8016 for oxygen, and 92235 for $^{235}U$. For elements for which the natural abundance is acceptable, NCZA can be entered as 1000*Z. For example, 92000 is used for uranium and 5000 is used for boron.

If the arbitrary material contains an element that has multiple isotopes, that element MUST be listed in Table M8.2.3. Uranium (92000), plutonium (94000), and boron (5000) are examples of isotope elements.

REPEAT THE SEQUENCE NCZA and ATPM (A6 and A7) FOR EVERY ELEMENT IN THE ARBITRARY MATERIAL.

A7. ATPM Enter ONLY for arbitrary materials. Repeat the sequence A6 and A7 for every element in the arbitrary material before going on to entry 2.

If ICP is 1 (entry A4), enter the NUMBER OF ATOMS PER MOLECULE.
If the arbitrary material is a compound (entry A4, ICP is 1), enter the number of atoms of this element per molecule of arbitrary material.

If ICP is 0 (entry A4), enter the WEIGHT PERCENT OF THIS ELEMENT IN THE ARBITRARY MATERIAL. If the arbitrary material is a mixture, alloy, or conglomerate rather than a compound (entry A4, ICP is 0), enter the weight percent of this element in the arbitrary material. If ICP is 0, the sum of all the weight percents for the arbitrary material MUST be 100.0.

2. MX Mixture ID number. A mixture number is required on every standard composition specification. It defines the mixture that contains the material defined by the standard composition specifications data. The mixture numbers are utilized in the unit cell specification for INFHOMMEDIUM, LATTICECELL, or MULTIREGION problems and the KENO V.a geometry data.

S1. FD Fuel density. Enter ONLY for a solution. The standard composition component name for a solution starts with the four characters "SOLN." The available solution names are listed in Sect. M8.3. Enter the fuel density in units of grams of heavy metal (fissile material) per liter of solution.

S2. AML Acid molarity. Enter ONLY for a solution. If there is no acid in the solution, enter zero. If acid is present in the solution, enter the molarity of the solution.

O1. SPGR or ROTH Optional data. Does NOT apply to ARBITRARY MATERIALS. These optional data can be entered for a solution or for a basic standard composition using a keyword in the form SPG=SPGR, DEN=SPGR, DEN=ROTH, or SPG=ROTH. The keywords SPG= and DEN= are interchangeable. SPGR is the specific gravity of the fuel solution. If the standard composition name begins with SOLN and if the specific gravity of the solution is known, it should be entered. Otherwise the code has to interpolate in a table to obtain a specific gravity to use in the calculation. ROTH is the specified density of the standard composition in grams per cubic centimeter and can be entered if desired. The interaction between ROTH, the density of the standard composition, and VF (entry 3) is demonstrated in Sect. M7.5.5.

3. VF Volume fraction. The default value of the volume fraction is 1.0. It can be omitted if items 4, 5, 6a, and 6b are also omitted. The volume fraction can be interpreted as:

a. the volume fraction of this standard composition component in the mixture,

b. the density of the standard composition component in this application divided by the theoretical density listed in the Standard Composition Library (Sect. M8.2), or a product of (a) and (b).

Section M7.5.5 discusses the interaction between ROTH and VF. For example, assume a homogenized mixture representing the water moderator and zircalloy cladding around a fuel pin is to be described. If the volume of the clad is 5.32 cc and if the volume of the water moderator is 44.68 cc, the mixture can be described using $H_2O$ with a volume fraction of 0.8936 [i.e., 44.68/(44.68+5.32)] and ZIRCALLOY with a volume fraction of 0.1064 [i.e., 5.32/(44.68+5.32)].

For $UO_2$ fuel at 95% theoretical density, the volume fraction is 0.95. For the coolant water in an operating pressurized water reactor, the volume fraction might typically be 0.71.

To describe a mixed-oxide fuel pin with a density of 10.5 g/cc, that is 17.8 wt % $PuO_2$ and 82.2 wt % $UO_2$, two standard compositions specifications must be entered: one for the $PuO_2$ component and one for the $UO_2$ component. From the Standard Composition Library, the theoretical density of the $PuO_2$ is 11.46 g/cc, and that of the $UO_2$ is 10.96 g/cc. Therefore, the volume fraction for the $PuO_2$ is 0.1631 (i.e., 0.178*10.50/11.46), and the volume fraction for the $UO_2$ is 0.7875 (i.e., 0.822*10.50/10.96).

Occasionally, a special mixture will be desired for which the determination of the component volume fractions will not be so straightforward. Consider a mixture of boral having 35 wt % $B_4C$, 65 wt % aluminum, and an overall density of 2.64 g/cc. Assume that neither boral nor $B_4C$ are in the Standard Composition Library. The volume fractions corresponding to boron, carbon, and aluminum may be computed as illustrated in Sect. M7.5.5.4, examples 3 and 4. Note that the calculation is complicated by the fact that $B_4C$ is a compound and boron has two isotopes. It might be simpler to enter $B_4C$ and aluminum as arbitrary materials as illustrated in example 1 of Sect. M7.5.5.5. Examples 1 and 2 of Sect. M7.5.5.4 illustrate the use of $B_4C$ and aluminum to describe a boral mixture. **If the volume fraction is set to 0.0, the user can enter the number density for the nuclide (in atoms per barn-cm) for item 4. Number densities cannot be entered for standard compositions that represent multiple nuclides.**

4. ADEN Number density. Enter a value for the number density ONLY if the volume fraction (VF, entry 3) was entered as zero. The number density is entered in units of atoms per barn-cm.

5. TEMP Temperature. The default value of the temperature is 293 K. The temperature can be omitted if items 6a and 6b are also omitted.

The temperature is used for Doppler broadening and/or the processing of thermal scattering data.

The resonance self-shielding calculation performed by NITAWL-II uses the Nordheim integral method and can account for Doppler broadening of the resonances at any specified temperature. If this material is known to contain a resonance nuclide, the user should enter an estimate of the temperature of the material.

For each of the light nuclides ($A \leq 20$ a.m.u.), the master cross-section library may contain one or more sets of thermal scattering data, each set corresponding to a different temperature. Scattering matrices for nuclides in media at elevated temperatures are generally fuller than those at lower

temperatures. Section M4 indicates which multiple sets of thermal scattering data are currently available. If multiple sets of thermal scattering data are available, the user should enter a rough estimate of the temperature of the material. The code will then use the set of cross-section data that is most appropriate.

6a. IZA Isotope ZA number. Enter a value for each isotope in the standard composition component, item 1. Do not enter a value if the volume fraction, VF, is zero (entry 3).

The ZA number of the isotope is entered if the user wishes to specify the isotopic distribution. This is done by entering 6a and 6b for each isotope until all the desired isotopes have been described. The "ZA" ID number is (A+1000*Z), where A is the atomic mass or weight of the isotope, and Z is the atomic number.

Entries 6a and 6b can be skipped if the default values listed in the isotopic distribution, Sect. M8.2, are acceptable.

6b. WTP Weight percent of the isotope. Do not enter a value if the volume fraction, VF, item 3 is zero. If entry 6a is entered, a value must also be entered for 6b. The weight percent of the isotope is the percent of this isotope in the element.

The user need not enter entries 6a and 6b for each isotope listed for the element in Sect. M8.2. The only requirement is that the sum of entries 6b for the element must sum to 100.0. Thus, uranium could be specified as 92235 3.2 92238 96.8 to represent 3.2% enriched uranium and ignore the 234 and 236 isotopes. However, 92235 3.0 and 92238 96.8 would not be correct because the 3.0 and 96.8 do not sum to 100.0. But 92235 3.0, 92234 0.2, and 92238 96.8 would be correct.

REPEAT THE SEQUENCE 6a and 6b until all the isotopes of the element have been defined as desired.

7. END The word END is entered to terminate the input data for a standard composition component. This END can have a label associated with it that can be as long as 12 characters. The label is optional and, if entered, is separated from the END by a single blank. At least two blanks must separate item 7 from the next data entry.

END COMP The words END COMP are entered only once for a problem. Their purpose is to signal the end of all the standard composition specification data that will be used in the problem. At least two blanks should separate entry 7 from the END COMP. Note that COMP is a label associated with this END. This label can be as long as 12 characters, but the first four characters must be COMP.

## C4.4.5 OPTIONAL UNIT CELL SPECIFICATION FOR INFINITE HOMOGENEOUS PROBLEMS

This section describes the unit cell data that can be entered for an INFHOMMEDIUM problem (data position 3 of Table C4.4.8). By default, the smallest mixture number is the mixture that will be used for the cell in XSDRNPM. In order to calculate the eigenvalue of the system, the mixture should contain fissionable material. Unit cell data are required for an infinite homogeneous medium problem ONLY if XSDRNPM is utilized by the control sequence and a mixture number other than the smallest mixture number is to be used in the cell. Control sequences that end in X (CSAS1X, etc.) utilize XSDRNPM. Additional information is available in Sect. M7.5.12 (see also Sect. M7.5.4.1 and M7.5.6.1).

The data required to specify the unit cell for an INFHOMMEDIUM problem are given in Table C4.4.10. The individual entries are explained in the text following the table.

Table C4.4.10 Optional unit cell specification for INFHOMMEDIUM problems

| Entry number | Variable name | Type of data | Data entry | Comments |
|---|---|---|---|---|
| 1 | NAME | Keyword | CELLMIX | Initiates reading cell data for INFHOMMEDIUM |
| 2 | MFUEL | Mixture number in the cell | Mixture number | Specifies the mixture number to be used in the cell. Defaults to the smallest mixture number entered in the standard composition data |

NOTE: Unit cell data are necessary for INFHOMMEDIUM only if a mixture number other than the smallest mixture number is to be used in the cell and if a control sequence that executes XSDRN has been specified. If mixtures 1, 2, and 3 are specified and if mixture 3 is to be used in the cell, enter CELLMIX 3.

1. NAME Keyword. The keyword CELLMIX is entered to indicate that unit cell data will be entered. If this entry is entered, entry 2 must also be entered. These data can be entered ONLY if item 3, the "type of calculation" from Table C4.4.8, is entered as INFHOMMEDIUM.

2. MFUEL Mixture number. The mixture number that defines the mixture to be used in the cell. The default value is the smallest mixture number specified in the standard compositions specification data. If either entry 1 or 2 is entered, the other must also be entered.

To enter the above data, enter the keyword CELLMIX, one or more blanks, and the desired mixture number, MFUEL, as shown below:

CELLMIX MFUEL

To specify mixture 5 to be used in the cell, enter the following data:

CELLMIX 5

**WARNING: If unit cell data are to be entered as in example 3 below, the word CELLMIX must be spelled correctly. If it is misspelled, the code will not give an error message and will attempt to use the smallest mixture number. The input data will get out of phase for CSAS and SAS sequences that expect to read data after the unit cell specification. For example, CSAS25 expects to read KENO V.a data after the unit cell specification. If CELLMIX is misspelled, the code assumes that it is reading the KENO V.a title card and the data reading gets out of phase.**

## C4.4.6 UNIT CELL SPECIFICATION FOR LATTICECELL PROBLEMS

This section describes the unit cell data that must be entered for a LATTICECELL problem (data position 3 of Table C4.4.8). Additional information is available in Sects. M7.5.4, M7.5.4.2, M7.5.6.2, and M7.5.8.2. The lattice cell description is especially suited to large arrays of identical cells. The code is limited to a single unit cell description for each problem. The unit cell data are used to provide the lump shape and dimensions for resonance cross-section processing, to provide lattice corrections for cross-section processing, and to provide data used by XSDRNPM to create cell-weighted cross sections. Thus the unit cell specification plays a major role in providing problem-dependent cross sections. The configuration of the unit cell is determined by the specified "type of lattice." The unit cells are limited to infinitely long cylindrical rods in a square or triangular lattice, spheres in a cubic or triangular lattice, a symmetric array of slabs, or an asymmetric array of slabs. Two distinct types of cells ("Regular" and "Annular") are available to be used in a lattice cell problem. "Regular" cells allow a concentric spherical, cylindrical, or symmetric slab configuration constrained to a central fuel region surrounded by an optional gap, an optional clad, and an external moderator. The outer boundary of the cell is limited to provide an array having a square or triangular pitch. "Annular" cells allow a concentric spherical, cylindrical, or asymmetric slab configuration constrained to a central (second) moderator region surrounded by a fuel region having an optional gap and optional clad on each side of the fuel. The gap, if present, is identical on both sides of the fuel. The clad, if present, is identical in dimension and composition on both sides of the fuel. The outer region of the cell is filled with an external (first) moderator. The outer boundary of the cell is limited to provide an array having a square or triangular pitch.

Regular cells are: SQUAREPITCH, TRIANGPITCH, SPHSQUAREP, SPHTRIANGP, and SYMMSLABCELL.

Annular cells are: ASQUAREPITCH, ASQP, ATRIANGPITCH, ATRP, ASPHSQUAREP, ASSP, ASPHTRIANGP, ASTP, and ASYMSLABCELL.

The unit cell data are utilized by the code to define the geometric and resonance self-shielding corrections that will be applied to the cross sections. Cell-weighted cross sections are created only if XSDRNPM is invoked by executing a control sequence ending in X, (CSAS1X, CSAS2X, etc.). The cell-weighted cross sections have a flux disadvantage factor applied to them. Because this weighting is applied to each mixture used in the cell, these cell-weighted mixtures should not be used in the geometry description of a subsequent code. However, the cell-weighted mixture that is created is always assigned a mixture number of 500 and can be used in a subsequent code. KENO V.a is the subsequent code in CSAS2X and CSAS4X. Note that cell-weighted cross sections are not generated if CSAS25 or CSAS4 are used.

The unit cell geometry data required to specify a LATTICECELL problem are given in Table C4.4.11. The individual entries are explained in the text accompanying the table.

1. CTP Type of lattice. This defines the type of lattice or array configuration. Any one of the following alphanumeric descriptions may be used. Note that the alphanumeric description must be separated from subsequent data entries by one or more blanks. Figures C4.4.2 through C4.4.7 are used to illustrate the type of lattice configurations available in a LATTICECELL problem. The fuel region is labeled F; the gap, G; the clad, C; and the moderator, M. If two moderators are present, M1 is the first moderator, and M2 is the second moderator. Both "regular" and "annular" unit cells are available. The arrangement of the materials in the unit cell is strictly constrained. Regular cells are limited to a minimum of two regions and a maximum of four regions. The innermost regions must be fuel, which is surrounded by an optional gap and/or optional clad. The outer region is moderator.

## Table C4.4.11 Unit cell specification for latticecell problems

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CTP | Type of lattice | Always | | Describes the type of lattice or array configuration |
| | | | | SQUAREPITCH | Use for cylindrical rods in a square pitch |
| | | | | ASQUAREPITCH | Use for annular cylindrical rods in a square pitch |
| | | | | ASQP | Use for annular cylindrical rods in a square pitch |
| | | | | TRIANGPITCH | Use for cylindrical rods in a triangular pitch |
| | | | | ATRIANGPITCH | Use for annular cylindrical rods in a triangular pitch |
| | | | | ATRP | Use for annular cylindrical rods in a triangular pitch |
| | | | | SPHSQUAREP | Use for spherical pellets in a cubic lattice |
| | | | | ASPHSQUAREP | Use for annular spherical pellets in a cubic lattice |
| | | | | ASSP | Use for annular spherical pellets in a cubic lattice |
| | | | | SPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASTP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | SYMMSLABCELL | Use for a symmetric array of slabs |
| | | | | ASYMSLABCEL | Use for a periodic, but asymmetric array of slabs |
| 2 | PITCH | Array pitch (cm) | Always | Appropriate dimension | The center-to-center spacing (cm) between fuel lumps<br>For asymmetric slab cell, enter the distance from the center of one moderator to the center of the other moderator (cm) |
| 3 | FUELOD | Outside dimension of fuel (cm) | Always | Appropriate dimension | Outside diameter of fuel (cm), or the thickness of the fuel in a slab |
| 4 | MFUEL | Fuel mixture number | Always | Mixture number | Mixture number representing the fuel |
| 5 | MMOD | Moderator mixture number | Always | Mixture number | Mixture number representing the moderator |
| 6 | MMOD2 | 2nd moderator mixture number | ASYMSLABCELL annular cells | Mixture number | Mixture number representing the second moderator |
| 7 | TKMOD2 | 2nd moderator thickness<br>or<br>2nd moderator diameter (cm) | ASYMSLABCELL<br>or<br>annular cells | Thickness<br>or<br>diameter | Thickness of the second moderator (cm) for ASYMSLABCELL<br>or<br>diameter of inner moderator (cm) for other annular cells |
| 8 | CLADOD | Outside diameter of clad (cm) | If clad | Clad OD | OMIT IF NO CLAD. For a slab, CLADOD is the sum of thicknesses of the fuel, gap, and clad |
| 9 | MCLAD | Clad mixture number | If clad | Mixture number | OMIT IF NO CLAD. Mixture number representing the clad |
| 10 | CLADID | Inside diameter of clad (cm) | If gap | Clad ID | OMIT IF NO GAP between the fuel and clad |
| 11 | MGAP | Gap mixture number | If gap | Mixture number | OMIT IF NO GAP between the fuel and clad. A mixture number of zero is often used |
| 12 | END | Terminate LATTICECELL data | Always | END | Terminate the LATTICECELL input data by entering the word END. Do not start in column 1. At least two blanks must follow entry 12 |

Annular cells are limited to a minimum of three regions. The innermost region must be the second moderator. The annular fuel region is outside the second moderator. If a clad is present, it must be identical in thickness and composition on both sides of the fuel. If a gap is present, it is the same thickness on both sides of the fuel. If the first and second moderators are the same material at the same density and temperature and do not contain resonance nuclides, the same mixture number can be used for both moderators. Otherwise, different mixture numbers must be used for the two moderators.

Regular and annular cell configurations are specified as shown below.

**Regular Cells**

SQUAREPITCH is used for an array of cylinders arranged in a square lattice as shown in Fig. C4.4.2. The clad and/or gap can be omitted.

TRIANGPITCH is used for an array of cylinders arranged in a triangular pitch lattice as shown in Fig. C4.4.3. The clad and/or gap can be omitted.

SPHSQUAREP is used for an array of spheres arranged in a square pitch lattice. A cross-section view through a cell is represented by Fig. C4.4.2. The clad and/or gap can be omitted.

SPHTRIANGP is used for an array of spheres arranged in a triangular pitch (dodecahedral) lattice. A cross-section view through a cell is represented by Fig. C4.4.3. The clad and/or gap can be omitted.

SYMMSLABCELL is used for an infinite array of symmetric slab cells as shown in Fig. C4.4.4. The clad and/or gap can be omitted.

**Annular Cells**

ASQUAREPITCH or ASQP is used for annular cylindrical rods in a square pitch lattice as shown in Fig. C4.4.5. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

ASPHTRIANGP or ASTP is used for spherical shells in a triangular pitch (dodecahedral) lattice as shown in Fig. C4.4.6. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

ASYMSLABCELL is used for a periodic, but asymmetric, array of slabs as shown in Fig. C4.4.7. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

2. PITCH Array pitch. This is the center-to-center spacing between the fuel lumps (rods, pellets, or slabs) in centimeters shown in Figs. C4.4.2 through C4.4.7. For an ASYMSLABCELL, enter the distance from the center of one moderator to the center of the other moderator. See Fig. C4.4.7.

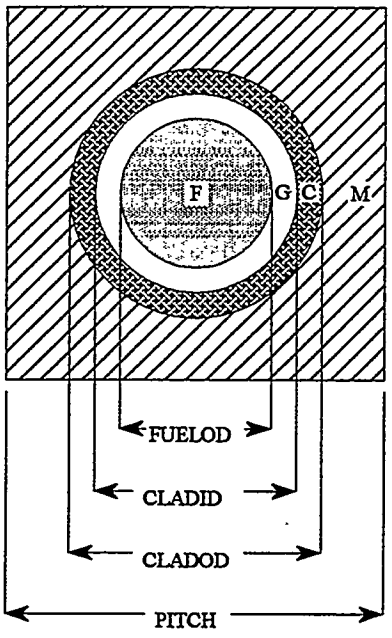



Figure C4.4.2 Arrangement of materials in a SQUAREPITCH and SPHSQUAREP unit cell

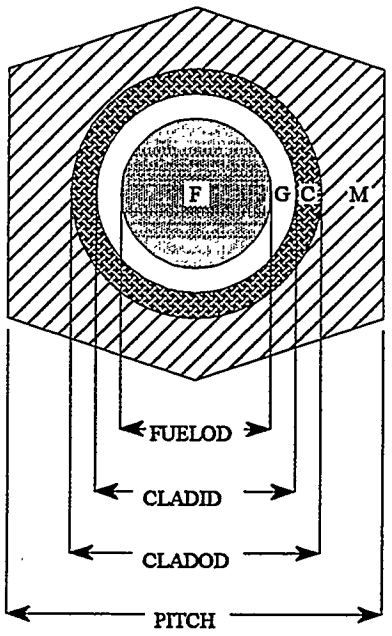



Figure C4.4.3 Arrangement of materials in a TRIANGPITCH and SHPTRIANGP unit cell

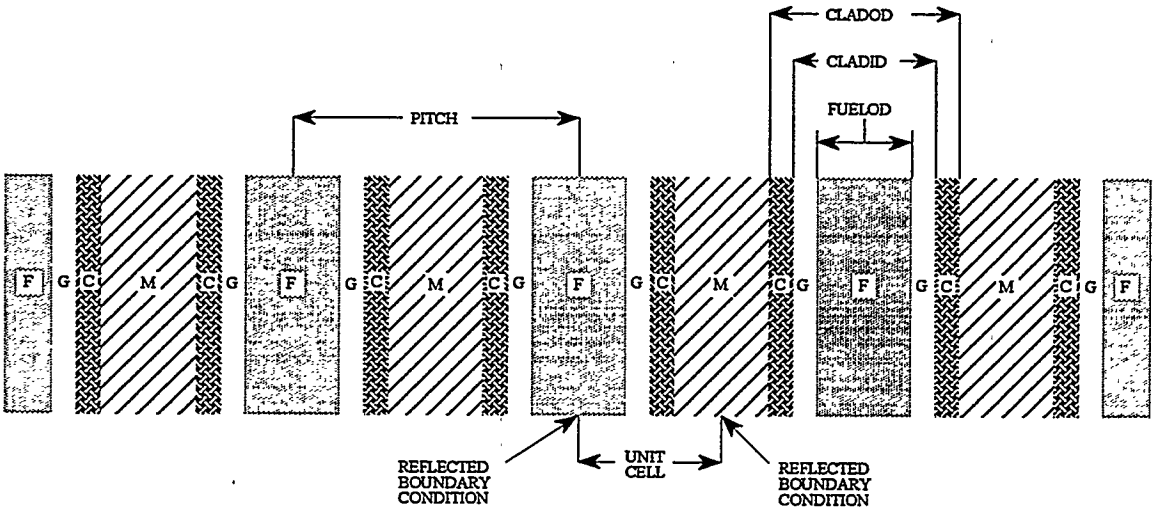



Figure C4.4.4 Arrangement of materials in a SYMMSLABCELL unit cell having reflected left and right boundary conditions

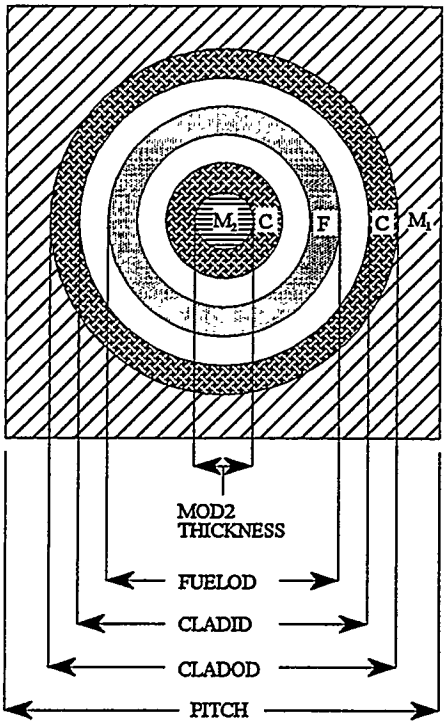



Figure C4.4.5 Arrangement of materials in an ASQUAREPITCH or ASPHSQUAREP unit cell

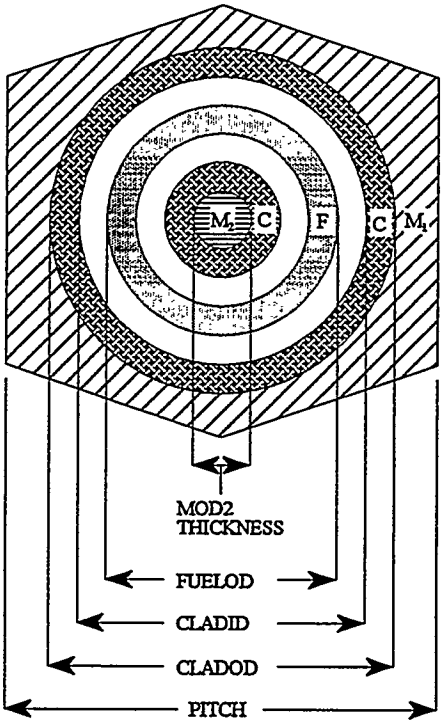



Figure C4.4.6 Arrangement of materials in an ATRIANGPITCH or ASPHTRIANGP unit cell

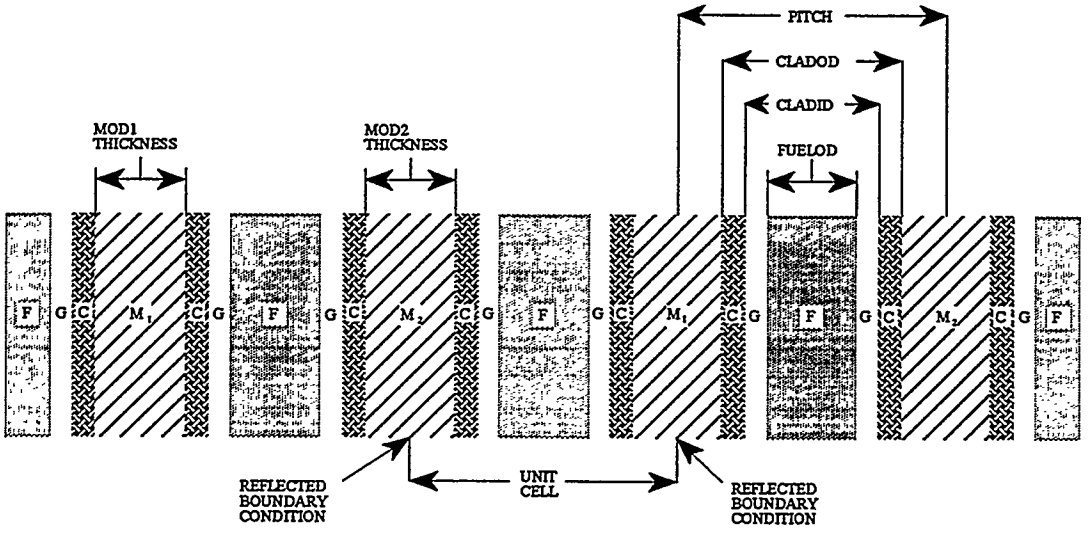



Figure C4.4.7 Arrangement of materials in an ASYMSLABCELL unit cell

3. FUELOD Outside dimension of fuel. This is the outside diameter of the fuel in centimeters as shown in Figs. C4.4.2 through C4.4.7. In slab geometry, enter the thickness of the fuel (see Figs. C4.4.2 through C4.4.7).

4. MFUEL Fuel mixture number. This is the mixture number representing the fuel (F in Figs. C4.4.2 through C4.4.7).

5. MMOD Moderator mixture number. This is the mixture number representing the moderator (M or M1 in Figs. C4.4.2 through C4.4.7).

6. MMOD2 Second moderator mixture number. Enter ONLY for annular cells. This is the mixture number representing the second moderator (M2 in Figs. C4.4.5 through C4.4.7).

7. TKMOD2 Second moderator thickness. Enter ONLY for annular cells. This is the thickness or diameter of the second moderator in centimeters. See Figs. C4.4.5 through C4.4.7.

8. CLADOD Outside diameter of clad. Enter ONLY if a clad is present. Enter the thickness of the clad in centimeters. For a slab, CLADOD is the sum of the fuel thickness, twice the gap thickness, and twice the clad thickness. See Figs. C4.4.2 through C4.4.7.

9. MCLAD Clad mixture number. Enter ONLY if a clad is present. Enter the mixture number that represents the clad (C in Figs. C4.4.2 through C4.4.7).

10. CLADID Inside diameter of clad. Enter ONLY if a gap is present between the fuel and the clad. Omit if there is no gap. If a gap is present, enter the inside diameter of the clad in centimeters. This corresponds to the outside diameter of the gap. In slab geometry, CLADID is the sum of the fuel thickness and twice the gap thickness. See Figs. C4.4.2 through C4.4.7.

11. MGAP Gap mixture number. Enter ONLY if a gap is present between the fuel and the clad. Omit if there is no gap. Enter the mixture number representing the gap (G in Figs. C4.4.2 through C4.4.7). Zero is often used to represent a void.

12. END The word END is entered to terminate the LATTICECELL data. An optional label can be associated with this END. The label can be as many as 12 characters long and is separated from the END by a single blank. At least two blanks must follow entry 12.

## C4.4.7 UNIT CELL SPECIFICATION FOR MULTIREGION PROBLEMS

These data are entered only if the problem is defined as a MULTIREGION problem. They describe the additional geometry data that are required for a MULTIREGION problem (data position 3 of Table C4.4.8). A MULTIREGION problem can be used to define a geometric arrangement that is more complicated than is allowed by a lattice cell. It can also be used for large geometric regions where the geometry effects for the cross sections are minimal. Additional information is available in Sects. M7.5.4, M7.5.4.3, M7.5.6.3, and M7.5.8.3. BE FULLY AWARE THAT CROSS SECTIONS GENERATED USING MULTIREGION IGNORE THE LATTICE GEOMETRY EFFECTS IN DOING THE RESONANCE CORRECTIONS.

The additional data required for a MULTIREGION problem are given in Table C4.4.12; individual entries are explained in the text accompanying the table.

1. CS Type of geometry. The type of geometry must always be specified for a MULTIREGION problem. The cross-section corrections do not account for lattice effects. The available geometry options are listed below.

   SLAB. This is used to describe a slab geometry.

   CYLINDRICAL. This is used to describe cylindrical geometry.

Table C4.4.12 Geometry specification for MULTIREGION problems

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CS | Type of geometry | Always | | Describes the type of geometry. The options are listed below |
| | | | | SLAB | Use for slab geometry |
| | | | | CYLINDRICAL | Use for cylindrical geometry |
| | | | | SPHERICAL | Use for spherical geometry |
| | | | | BUCKLEDSLAB | Use for slab geometry with a buckling correction for the two transverse directions |
| | | | | BUCKLEDCYL | Use for cylindrical geometry with a buckling correction in the axial direction |
| 2 | BR | Right boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | VACUUM | Default is VACUUM. Describes the right/outside boundary condition |
| | | | | REFLECTED | This provides a non-return condition at the boundary |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | Do not use for cylindrical or spherical<br>This provides isotopic return at the boundary |
| 3 | BL | Left boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | VACUUM | Default is REFLECTED. Describes the left boundary condition |
| | | | | REFLECTED | This provides a non-return condition at the boundary |
| | | | | PERIODIC | Recommended for cylindrical or spherical |
| | | | | WHITE | Do not use for cylindrical or spherical<br>This provides isotropic return at the boundary |
| 4 | ORGN | Location of left boundary on the x-axis (cm) | BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | Appropriate dimension | Default is 0.0. Should not be changed for cylindrical or spherical geometry. A value must be entered if subsequent data are to be entered |
| 5 | DY | Buckling height (cm) | BUCKLEDSLAB and BUCKLEDCYL | Appropriate dimension | Omit for SLAB, CYLINDRICAL, and SPHERICAL.<br>This corresponds to one of the transverse dimensions of an actual 3-D slab assembly or to the length of a finite cylinder |
| 6 | DZ | Buckling depth (cm) | BUCKLEDSLAB | Appropriate dimension | Omit unless BUCKLEDSLAB was specified.<br>This is the buckling depth corresponding to the second transverse dimension of a 3-D slab assembly |
| 7 | END | End geometry parameters | Always | END | Enter the word END. Do not start in column 1 |
| 8 | MXZ | Mixture number in the zone | Always | Mixture number | Repeat entry numbers 8-10 until all zones are defined.<br>Enter the mixture number for this zone |
| 9 | RZ | Outside radius of the zone (cm) | Always | Appropriate dimension | Repeat entry numbers 8-10 until all zones are defined.<br>Enter the outside dimension of the zone (cm) |
| 10 | XMOD | External moderator index | Optional | | Repeat entry numbers 8-10 until all zones are defined.<br>Entry 10 is optional and can be omitted. If it is omitted, repeat entries 8 and 9 until all zones are defined<br>No moderating materials in the adjacent zones |
| | | | | NOEXTERMOD | A moderating material is present in one adjacent zone |
| | | | | ONEEXTERMOD | Moderating materials are present in two adjacent zones |
| | | | | TWOEXTERMOD | |
| | END ZONE | Terminate zone data | | END ZONE | Enter when all zones have been defined by repeating entries 8 through 10 for each zone |

SPHERICAL. This is used to describe spherical geometry.

BUCKLEDSLAB. This is used for slab geometry with a buckling correction for the two transverse directions.

BUCKLEDCYL. This is used for cylindrical geometry with a buckling correction in the axial direction.

2. BR Right boundary condition. This is defaulted to vacuum. A value for BR **MUST** be entered for BUCKLEDSLAB and BUCKLEDCYL but is optional for SLAB, CYLINDRICAL, or SPHERICAL. The available options and their qualifications are listed below:

VACUUM. This imposes a vacuum at the boundary of the system.

REFLECTED. This imposes mirror image reflection at the boundary. Do not use for CYLINDRICAL or SPHERICAL.

PERIODIC. This imposes periodic reflection at the boundary. Do not use for CYLINDRICAL or SPHERICAL.

WHITE. This imposes isotropic return at the boundary.

3. BL Left boundary condition. This is defaulted to reflected. A value for BL **MUST** be entered for BUCKLEDSLAB but is optional for SLAB, CYLINDRICAL, or SPHERICAL. The available options and their qualifications are listed below:

VACUUM. This imposes a vacuum at the boundary of the system.

REFLECTED. This imposes mirror image reflection at the boundary. For CYLINDRICAL or SPHERICAL, this is the only valid boundary condition because the left boundary corresponds to the centerline.

PERIODIC. This imposes periodic reflection at the boundary.

WHITE. This causes isotropic return at the boundary.

4. ORGN Location of left boundary on the origin. The default value of ORGN is 0.0. This is the only value allowed for CYLINDRICAL or SPHERICAL geometry. Enter the location of the left boundary on the axis perpendicular to the slab (in cm).

5. DY Buckling height. Enter *ONLY* for BUCKLEDSLAB or BUCKLEDCYL. This is the buckling height in centimeters. It corresponds to one of the transverse dimensions of an actual 3-D slab assembly or the length of a finite cylinder.

6. DZ Buckling depth. Enter *ONLY* for BUCKLEDSLAB. This is the buckling depth in centimeters. It corresponds to the second transverse dimension of an actual 3-D slab assembly.

7. END The word END is entered to terminate these data before entering the zone description data. It must not be entered in columns 1 through 3. At least two blanks are required between items 1 and 7 if they are the only entries. A label can be associated with this END. The label can be a maximum of 12 characters and is separated from the END by a single blank. At least two blanks must follow entry 7.

The zone description data are entered at this point. Entries 8, 9, and 10 are entered for each zone, and the sequence is repeated until all the desired zones have been described. To terminate the data, enter the words END ZONE.

8. MXZ Mixture number in the zone. Enter the mixture number of the material that is present in the zone. Enter a zero for a void. Repeat the sequence of entries 8 through 10 for each zone.

9. RZ Outside radius of the zone. Enter the outside dimension of the zone in centimeters. In slab geometry, RZ is the location of the zone's right boundary on the x-axis. Repeat the sequence of entries 8 through 10 for each zone.

10. XMOD Optional external moderator index. This optional entry describes the conditions existing in the zones adjacent to the present zone. Appropriate conditions are provided by default. The available options are listed below.

NOEXTERMOD. This means there are no moderating materials between this zone and either boundary.

ONEEXTERMOD. This means one of the zones between this zone and one of the boundaries contains a material.

TWOEXTERMOD. This indicates that at least one zone on either side of this zone contains a material.

CAUTION: User-provided entries do not always supersede the default values: (1) For SLAB geometry, user-supplied values will be used everywhere except zone 1. If the left boundary is reflected, the default value of ONEEXTERMOD will be used unless NOEXTERMOD is specified. (2) For SPHERICAL and CYLINDRICAL geometry, default values will be used for all zones unless NOEXTERMOD is specified. NOEXTERMOD allows the user to "turn off" geometric corrections.

**Repeat the sequence of entries 8 and 9 or 8 through 10 for each zone. When all the zones have been described, enter the words END ZONE.**

END ZONE This is used to terminate the MULTIREGION zone data. Enter the words END ZONE when all the zones have been described. Note that ZONE is a label associated with this END. This label can be as long as 12 characters, but the first four characters **must** be ZONE. At least two blanks must follow this entry.

## C4.4.8 OPTIONAL PARAMETER DATA

Most of the parameter data for Criticality Safety Analysis Sequence No. 4 are determined by the control module and cannot be adjusted. However, certain parameters can be altered. If the default parameters are acceptable, this section of input data should be omitted in its entirety. Nondefault values for one or more of the parameters can be specified by entering the words MORE DATA followed by the desired keyworded parameters and their associated values. One or more of the parameters can be entered in any order. Default values are used for parameters that are not entered. Each parameter is entered by spelling its name, followed immediately by an equal sign and the value to be entered. There should not be a blank between the parameter name and the equal sign. Each parameter specification must be separated from the rest by at least one blank. For example

MORE DATA ISN=16 EPS=0.00008 END

would result in using an S16 angular quadrature set and tightening the convergence criteria to 0.00008 in the XSDRNPM calculation. Note that the END is not the end for the analytical sequence, entry 5 of Table C4.4.7.

Table C4.4.13 describes the optional parameters that can be entered as data and the code or codes the parameters apply to. A description of each entry is contained in the text accompanying the table.

1. MORE DATA...These words, followed by one or more blanks, are entered ONLY if optional parameter data are to be entered. They must precede all other optional parameter data. Entries 2 through 19 can be entered in any order.

2. ISN Order of angular quadrature for XSDRNPM. Quadrature sets are geometry-dependent quantities that are defaulted to order 8 by the Material Information Processor. See Sects. M7.2.5.7 and M7.5.7 for a more detailed explanation.

3. SZF Spatial mesh size factor for XSDRNPM. The size of the largest mesh interval can be adjusted by entering a value for SZF. The default value is 1.0. A value between zero and 1.0 yields a finer mesh; a value greater than 1.0 yields a coarser mesh. See Sects. M7.2.5.6 and M7.5.7 for more details.

4. IIM Maximum number of inner iterations for XSDRNPM. This is the maximum number of inner iterations to be used in the XSDRNPM calculation. The default value is 20. See Sect. M7.5.7 for a more detailed explanation.

5. ICM Maximum number of outer iterations for XSDRNPM. This is the maximum number of outer iterations to be used in the XSDRNPM calculation. The default value is 25. If the calculation reaches the outer iteration limit, a larger value should be used. See Sect. M7.5.7 for a more detailed explanation.

6. EPS Overall convergence criteria for XSDRNPM. This is used by XSDRNPM after each outer iteration to determine if the problem has converged. The default value of EPS is 0.0001. A value less than 0.0001 tightens the convergence criteria; a larger value loosens the convergence criteria. See Sects. M7.2.5.8 and M7.5.7 for additional information.

Table C4.4.13 Optional parameter data

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 1 | MORE DATA | Input flag | | This signals that optional parameter data will be entered. Enter only those parameters you wish to change |
| 2 | ISN= | Order of angular quadrature | XSDRNPM | The default value is 8. This allows using another value |
| 3 | SZF= | Spatial mesh size factor | XSDRNPM | The default value is 1.0.<br>0<SZF<1.0 gives a finer mesh.<br>SZF>1.0 gives a coarser mesh |
| 4 | IIM= | Max. number of inner iterations | XSDRNPM | The default value is 20. This allows using another value |
| 5 | ICM= | Max. number of outer iterations | XSDRNPM | The default value is 25. This allows using another value |
| 6 | EPS= | Overall convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 7 | PTC= | Point convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 8 | BKL= | Buckling factor | XSDRNPM | The default value is 1.420892.<br>Use ONLY for a multiregion problem that specifies BUCKLEDSLAB or BUCKLEDCYL |
| 9 | IUS= | Upscatter scaling factor | XSDRNPM | The default value is zero.<br>IUS=0 doesn't utilize upscatter scaling.<br>IUS=1 uses upscatter scaling to accelerate the solution and/or speed convergence |
| 10 | RES= | Resonance data | BONAMI NITAWL-II | Enter the mixture number, geometry type (SLAB, CYLINDER, SPHERE), and the thickness of the slab or radius of the sphere or cylinder, in cm. Optionally enter the inner radius (cm) to specify anannular cylinder or sphere |
| 11 | DAN(mm)= | Dancoff factor for the specified mixture | BONAMI NITAWL-II | Enter the mixture number, mm, to which the Dancoff factor applies inside the parentheses; enter the Dancoff factor after the equal sign<br><br>Repeat entries 10 and 11 for all resonance mixtures used in the problem that are not treated in the LATTICECELL or MULTIREGION description |

Table C4.4.13 (continued)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 12 | BAL= | Key to print balance tables | XSDRNPM | The default value is FINE. BAL=NONE suppresses printing the balance table. BAL=ALL prints all balance tables. BAL=FINE prints only the fine-group balance tables |
| 13 | DY= | First transverse dimension | XSDRNPM | The first transverse dimension, in centimeters, used in a buckling correction to calculate leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder) |
| 14 | DZ= | Second transverse dimension | XSDRNPM | The second transverse dimension, in centimeters, used for a buckling correction (i.e., the width of a slab) |
| 15 | COF= | Diffusion coefficient option for transverse leakage correction | XSDRNPM | The default is 0. See Sect. F3.5, 3$ array, variable IPN |
| 16 | FRD= | Unit from which fluxes will be read | XSDRNPM | Enter the unit number from which the flux guess for XSDRNPM will be read |
| 17 | FWR= | Unit on which fluxes will be written | XSDRNPM | Enter the unit number where the binary fluxes from XSDRNPM will be written |
| 18 | DAB= | Number of direct access data blocks | MIP[a] | The default is 200. Number of blocks allocated for direct-access unit 90 |
| 19 | AXS= | Unit on which a mixed ANISN library will be written | ICE | Enter the unit number where ICE is to write a mixed ANISN library |
| 20 | COLL | Key to activate collapse of thermal groups | MIP[a] | Enter COLL to collapse all thermal groups into one group for the shielding sequences |
| 21 | END | Terminus | | Terminate the optional parameter data. Do not start in column 1. At least two blanks must follow entry number 23 |

[a]MIP is the Material Information Processor.

7. PTC Pointwise convergence criteria for XSDRNPM. This is the point flux convergence criteria used by XSDRNPM to determine if convergence has been achieved after an inner iteration. The default value is 0.0001. A smaller value tightens convergence; a larger value loosens it. See Sects. M7.2.5.8 and M7.5.7 for more details.

8. BKL Buckling factor for XSDRNPM. A buckling factor should be used ONLY for a MULTIREGION BUCKLEDSLAB or BUCKLEDCYL problem. Because cylinders are assumed to be infinitely long and slabs are assumed to be infinite in both transverse directions, the analytic sequence may tend to overestimate the total flux for a finite system. A buckling correction can be used to approximate the leakage from the system in the transverse direction(s). The extrapolation distance factor, BKL, is defaulted to 1.420892. See Sects. M7.2.5.4 and M7.5.7 for additional details.

9. IUS Upscatter scaling flag for XSDRNPM. This option allows the use of upscatter scaling to accelerate the solution or force convergence. The default value is zero, in which case upscatter scaling is not used. IUS=1 facilitates upscatter scaling. Guidelines are not available to indicate when upscatter scaling is needed. Some problems will not converge with it, and some will not converge without it. See Sect. M7.5.7 for a more detailed explanation.

10. RES Resonance data for BONAMI and NITAWL-II. This parameter allows overriding the default resonance data for materials that are not used in the unit cell. The default resonance data are infinite homogeneous media that may not accurately represent the geometry (for example, a thin clad or container region). The resonance data are entered in the form:
RES= mixture number geometry type thickness.

The mixture number is the material for which the resonance data apply. The geometry type can be a sphere, slab, or cylinder and defines the third entry of the 3* array in NITAWL-II. The thickness is the thickness of a slab or the radius of a sphere or cylinder in centimeters and defines the fourth entry of the 3* array in NITAWL-II. This entire data sequence should be entered for each material for which the default data are inadequate. See Sect. M7.5.7 for additional details. WARNING: If this parameter is used to enter resonance data for a mixture that is used in the unit cell specification, it will NOT be used. Information provided in the unit cell data will be used instead.

11. DAN(mm)= Dancoff factor for the specified mixtures. This allows entering a Dancoff factor to be used in the resonance correction for the specified mixture. The Dancoff data are entered in the form:

DAN(mm) = Dancoff factor.

Note that the parentheses must be entered as part of the data. The mixture number, mm, to which the Dancoff factor applies, must be enclosed in the parentheses. The Dancoff factor to be used is entered after the equal sign. See Sect. M7.5.7 for additional details.

Items 10 and 11 should be entered for each resonance mixture that is not specified in the LATTICECELL or MULTIREGION unit cell specification data. See Sects. M7.5.10 and C4.5.1 for additional information.

12. BAL Balance table print flag for XSDRNPM. This allows control of the balance table print from XSDRNPM. The default value is FINE. BAL=NONE suppresses the balance table print.

BAL=ALL prints all of the balance tables. BAL=FINE prints only the fine-group balance tables. See Sect. M7.5.7 for additional details.

13. DY= First transverse dimension for XSDRNPM. This is the first transverse dimension, in centimeters, used in a buckling correction to calculate the leakage normal to the principal calculation direction (the height of a slab or cylinder). It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a cylinder or slab system of finite height for a LATTICECELL problem. DY= is defaulted to an infinite height, or is set to DY for a buckled MULTIREGION cell description. See Sects. M7.2.5.4 and M7.5.7 for additional details.

14. DZ= Second transverse dimension for XSDRNPM. This is the second transverse dimension, in centimeters, used for a buckling correction for a slab of finite width. It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a LATTICECELL slab of finite width. DZ= is defaulted to an infinite width, or is set to DZ for a buckled MULTIREGION slab cell of finite width. See Sects. M7.2.5.4 and M7.5.7 for additional details.

15. COF= Diffusion coefficient for transverse leakage corrections in XSDRNPM. The default value is 0. The available options are

    COF=0 sets a transport-corrected cross section for each zone.

    COF=1 use a spatially averaged diffusion coefficient for each zone.

    COF=2 use a diffusion coefficient for all zones that is one-third the diffusion coefficient determined from the spatially averaged transport cross section for all zones.

    COF=3 use a flux and volume weighting across all zones.

    See Sect. M7.5.7 or Sect. F3.5, 3$ array, variable IPN for more details.

16. FRD= Unit from which fluxes are read for XSDRNPM. If fluxes are to be used for the initial flux guess, enter the unit number from which they are to be read. The user is responsible for supplying job control language data that is necessary for reading the flux dataset. See Sect. M7.5.7 for more details.

17. FWR= Unit on which fluxes will be written for XSDRNPM. If binary fluxes are to be saved from this XSDRNPM calculation, enter the unit number where they will be written. The user is responsible for supplying job control language data that is necessary for creating and saving the flux dataset. See Sect. M7.5.7 for more details.

18. DAB= Number of direct-access data blocks for the Material Information Processor. This is the number of blocks allocated for direct access on unit 90. The default is 200. See Sect. M7.5.7 for additional details. If the job fails in the Material Information Processor with an LMP005 DA ERROR, then DAB must be used to enter a larger value. See Sect. C4.7.4 for additional details.

19. AXS= Logical unit number for ICE. Write a mixed ANISN library on this unit. The default is zero (no library). If the unit number is input as 7, the ANISN library will be in free-form card image. For any other unit number, the library will be binary. The user is responsible for supplying job control language data that is necessary for creating and saving the ANISN cross-section library. An ANISN cross section can only be created by the control sequences CSASI, CSASIX, and SAS3. See Sect. M7.5.7 for additional details.

20. MSH= THE MAXIMUM NUMBER OF MESH POINTS PER RESONANCE...The default value is 2001. This is the maximum number of mesh points allowed for the Simpson's rule integration in NITAWL. This should be an odd number. See Section M7.5.7 for additional details.

21. MLV= HIGHEST RESONANCE $\ell$-VALUE FOR SELF-SHIELDING...The default value is 2. This is the highest resonance $\ell$-value for which a self-shielding calculation will be performed in NITAWL. See Sect. M7.5.7 for additional details.

22. COLL Activate collapse of thermal groups. This parameter should only be used with the shielding analysis sequences (SAS1, etc.). It causes the Material Information Processor to instruct XSDRN to collapse all thermal groups into one group.

21. END The word END is entered to terminate the optional parameter data. A label can be associated with this END. The label can be as long by a single blank. If this END is entered without a label, it must not begin in column 1. At least two blanks must follow this entry.

### C4.4.9 SUMMARY OF KENO V.a DATA

KENO V.a data are required for CSAS25, CSAS2X, CSAS4, and CSAS4X. A brief outline of the KENO V.a data is given in Table C4.4.14. The input data are summarized in Table C4.4.15. The input data for KENO V.a are discussed in detail in Sects. F11.4 and F11.5. Data need not be supplied for entries whose default value is acceptable. KENO V.a mixing table data cannot be entered or used in CSAS because they are automatically supplied by the control module.

### C4.4.10 SEARCH DATA

Search data are required only for CSAS4 and CSAS4X. They are omitted for all other sequences. Search data are entered in the form:

READ SEARCH search data END SEARCH

The words READ SEARCH inform the code that search data are to be read, and the words END SEARCH terminate the search data. The search data options are described in Sects. C4.4.10.1 and C4.4.10.2. Additional information and examples are contained in Sects. C4.5.6 through C4.5.11.

Table C4.4.14 Outline of KENO V.a data

| Type of data | Starting flag | Comments | Termination flag |
|---|---|---|---|
| Title | None | 80 columns, must be entered first | None |
| Parameters | READ PARM | Enter desired parameter data | END PARM |
| Biasing or weighting | READ BIAS | Enter desired biasing data | END BIAS |
| Geometry | READ GEOM | Enter desired geometry data | END GEOM |
| Array data or unit orientation | READ ARRAY | Enter desired array data | END ARRAY |
| Boundary conditions or albedoes | READ BNDS | Enter desired albedo data | END BNDS |
| Start data or initial source | READ START | Enter desired start data | END START |
| Mixing table data | READ MIXT | The CSAS4 modules allow only SCT= and EPS= | END MIXT |
| Plot data | READ PLOT | Enter desired plot data | END PLOT |
| KENO V.a data terminus | END DATA | Enter to signal the end of all KENO V.a data | |

## Table C4.4.15 Summary of KENO V.a data

### Summary of parameter data

TITLE: The title must be entered first (80 columns) See Sect. F11.4.3

PARAMETERS: Format: READ PARAM enter parameter data here END PARAM
If parameters are entered, they must follow the title. See Sects. F11.4.3, F11.5.2, and F11.5.3.

| KEY | STD. | DEFINITION |
|---|---|---|
| RND= | given | Random number |
| TME= | 120 min | Execution time (min) |
| TBA= | 0.5 min | Batch time (min) |
| WTA= | 0.5 | Average weight |
| WTH= | 3.0 | Wt. for splitting |
| WTL= | 1/WTH | Russian Roulette wt. |
| GEN= | 203 | No. of generations |
| NPG= | 1000 | No. per generation |
| NSK= | 3 | Generations skipped |
| RES= | 0 | Gens. between |
| NBK= | NPG+25 | restart |
| XNB= | 0 | Neutron bank |
| NFB= | NPG | positions |
| XFB= | 0 | Extra bank entries |
| X1D= | 0 | Fission bank |
| LNG= | 1000000 | positions |
| BEG= | 1 | Extra bank entries |
| NB8= | 200 | No. of extra 1-D's |
| NL8= | 512 | Words of storage* |
| | | Restart at this gen. |
| | | Blocks for d.a. unit |
| | | Length of d.a. block |

| KEY | STD. | DEFINITION |
|---|---|---|
| FLX= | NO | Fluxes |
| FDN= | NO | Fission densities |
| ADJ= | NO | Adjoint calculation |
| AMX= | NO | All mixture xsecs |
| XAP= | NO | Xsec angles and probs. |
| XS1= | NO | 1-D xsecs |
| XS2= | NO | 2-D xsecs |
| PKI= | NO | Fission spectrum |
| P1D= | NO | Extra 1-D xsecs |
| FAR= | NO | Fiss. and abs. |
| MKP= | NO | Matrix by location |
| CKP= | NO | Cofactor k by loc. |
| FMP= | NO | Fiss. prod. by loc. |
| MKU= | NO | Matrix by unit |
| CKU= | NO | Cofactor k by unit |
| FMU= | NO | Fiss. prod. by unit |
| GAS= | FAR | FAR by energy group |

| KEY | STD. | DEFINITION |
|---|---|---|
| MKH= | NO | Matrix by hole |
| CKH= | NO | Cofactor k by hole |
| FMH= | NO | Fiss. prod. by hole |
| HHL= | NO | MKH at highest level |
| MKA= | NO | Matrix by array |
| CKA= | NO | Cofactor k by array |
| FMA= | NO | Fiss. prod. by array |
| HAL= | NO | MKA at highest level |
| PLT= | YES | Printer plots |
| BUG= | NO | Debug print |
| TRK= | NO | Print neutron tracks |
| PWT= | NO | Print avg. weight |
| PGM= | NO | Unprocessed geometry |
| SMU= | NO | Self-multiplication |
| NUB= | YES | Neutrons per fission |
| PAX= | NO | Albedo-xsec array |
| RUN= | YES | Execute problem |
| PLT= | YES | Printer plots |

| KEY | STD. | DEFINITION |
|---|---|---|
| XSC= | 14 | Mixed xsecs |
| ALB= | 79 | Albedo |
| WTS= | 80 | Weights |
| LIB= | 4 | Working xsecs from NITAWL or XSDRN |
| SKT= | 16 | Scratch |
| RST= | 95 | Read restart |
| WRS= | 95 | Write restart |
| XSC= | 14 | Mixed xsecs |

*The words of storage in CSAS are defaulted to 100,000 and can be overridden by using the size= parameter on the analytical sequence specification which then becomes the default value of LNG in KENO V.a.
For example: =CSAS25 PARM=SIZE=200000
or =CSAS25 PARM=(CHECK,SIZE=200000)

Table C4.4.15 (continued)

## Summary of array data

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sects. F11.5.5, F11.5.6, and F11.5.7.

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

| ARRAY PARAMETERS | | | DATA TYPE |
|---|---|---|---|
| KEYWORD | DEFAULT | DEFINITION | FILL LOOP |
| ARA= | 1 | No. defining the array | |
| NUX= | 1 | No. of units in X direction | |
| NUY= | 1 | No. of units in Y direction | |
| NUZ= | 1 | No. of units in Z direction | |
| GBL= | Maxara | Global, universe, or overall array number** | |
| COM= | None | Delim comment delim Optional comment is a maximum of 132 characters | |

**Can be defaulted by the code. If specified, it need be entered only once per problem.

### ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data utilizing the options in this table, the count field and option field must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

| COUNT FIELD | OPTION FIELD | OPERAND FIELD | COMMENTS |
|---|---|---|---|
| | | j | Stores j at the current position in the array |
| i | R | j | Stores j in the next i positions in the array |
| i | * | j | Stores j in the next i positions in the array |
| i | $ | j | Stores j in the next i positions in the array |
| | F | j | Fills remainder of the array with unit no. j starting with the current array position |
| | A | j | Sets the current position in the array to j |
| i | S | | increments; current position in the array by i allows skipping i positions. The value of i may be positive or negative |
| i | Q | j | Repeats the previous j entries i times. The default value of i is 1 |
| i | N | j | Repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1 |
| i | B | j | Starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1 |
| i | P | j | Alternately stores j and -j in the next i positions of the array |
| i | I | j k | Provides the end points, j and k, with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the I option should only be used to generate integer steps [i.e., (k-j)/(i+1) should be a whole number]. |
| | T | | Terminates the data reading for the array |

### ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP is terminated by entering END LOOP.

ENTER DATA IN THE FORM:

| DATA ENTRY | COMMENTS |
|---|---|
| LTYPE | The unit or box type. LTYPE must be greater than 1 |
| IX1 | Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX |
| IX2 | Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX |
| INCX | The number of units by which increments are made in the X direction |
| IY1 | The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY |
| IY2 | Ending position int he Y direction. IY2 must be at least 1 and no larger than the value of NUY |
| INCY | The number of units by which increments are made in the positive Y direction |
| IZ1 | Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ |
| IZ2 | Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ |
| INCZ | The number of units by which increments are made in the positive Z direction |

Table C4.4.15 (continued)

## Summary of biasing data

BIAS (weighting)

Format: READ BIAS keyword correlation data auxiliary END BIAS
See Sects. F11.4.7 and F11.5.8

| KEYWORD | DESCRIPTION | |
|---|---|---|
| ID= | CORRELATION DATA will be read next. | |
| | id | Material ID. Enter from table at right to use weighting data from the library |
| | ibgn | Beginning bias ID |
| | iend | Ending bias ID |
| WT= | AUXILIARY DATA will be read next. | |
| WTS= | AUXILIARY DATA will be read next. | |
| | wttitl | Material title (12-character maximum) |
| | id | Material ID |
| | nsets | Number of sets of group structures |
| | REPEAT | (thkinc, numinc, ngpwt, wtavg) nsets times |
| | thkinc | Thickness per increment |
| | numinc | Number of increments |
| | ngpwt | Number of energy groups for this set of wts |
| | wtavg | Enter numinc x ngpwt values of wtavg |

| MATERIAL | ID | ENERGY GROUPS | THICKNESS/ INCREMENT |
|---|---|---|---|
| Concrete | 301 | 16,27,44,218,238 | 5 cm |
| Paraffin | 400 | 16,27,44,218,238 | 3 cm |
| Water | 500 | 16,27,44,218,238 | 3 cm |
| Graphite | 6100 | 16,27,44,218,238 | 20 cm |

For CORRELATION DATA, the material ID is chosen from material ID column above (the keyword is ID=).
For AUXILIARY DATA, the material ID is chosen by the user and the keyword is WT= or WTS=.
When AUXILIARY DATA are entered, CORRELATION DATA must also be entered to use the data.
Beginning and ending bias ID's are defined by the user. The geometry specificaiton that has the bias ID equal to the beginning bias ID utilizes the wt avg's from the first interval of material ID.

Table C4.4.15 (continued)

## Summary of boundary condition data

BNDS (albedo or boundary conditions)

Format: READ BNDS face code albedo name END BNDS
See Sect. F11.4.7

The sequence FACE CODE ALBEDO NAME is entered as many times as necessary to define the appropriate albedo boundary conditions. The default for all faces is vacuum.

FACE CODES FOR ENTERING BOUNDARY (ALBEDO) CONDITIONS

| FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION |
|---|---|---|---|---|---|---|---|
| +XB= | Positive X face | XFC= | Both X faces | +YX= | Positive X and Y faces | &ZY= | Positive Y and Z faces |
| &XB= | Positive X face | YFC= | Both Y faces | &XY= | Positive X and Y faces | -XY= | Negative X and Y faces |
| -XB= | Negative X face | ZFC= | Both Z faces | &YX= | Positive X and Y faces | =XZ= | Negative X and Z faces |
| +YB= | Positive Y face | +FC= | All positive faces | +XZ= | Positive X and Z faces | =YZ= | Negative Y and Z faces |
| &YB= | Positive Y face | &FC= | All positive faces | +ZX= | Positive X and Z faces | YXF= | All X and Y faces |
| -YB= | Negative Y face | -FC= | All negative faces | &XZ= | Positive X and Z faces | ZXF= | All X and Z faces |
| +ZB= | Positive Z face | XYF= | All X and Y faces | &ZX= | Positive X and Z faces | ZYF= | All Y and Z faces |
| &ZB= | Positive Z face | XZF= | All X and Z faces | +YZ= | Positive Y and Z faces | -YX= | Negative X and Y faces |
| -ZB= | Negative Z face | YZF= | All Y and Z faces | +ZY= | Positive Y and Z faces | -ZX= | Negative X and Z faces |
| ALL= | All 6 faces | +XY= | Positive X and Y faces | &YZ= | Positive Y and Z faces | -ZY= | Negative Y and Z faces |

ALBEDO NAMES AVAILABLE ON THE KENO V ALBEDO LIBRARY, FOR USE WITH THE FACE CODES

| ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION |
|---|---|---|---|---|---|
| DPOH20<br>DPOH20<br>DPO<br>DPO | 12-in. double PO water differential albedo with 4 incident angles | CONC-4<br>CON4<br>CONC4 | 4-in. concrete differential albedo with 4 incident angles | VACUUM<br>VOID<br>VACU<br>VAC | Vacuum condition |
| H20<br>WATER | 12-in. water differential albedo with 4 incident angles | CONC-8<br>CON8<br>CONC8 | 8-in. concrete differential albedo with 4 incident angles | SPECULAR<br>MIRROR<br>MIRR | Mirror image reflection |
| PARAFFIN<br>PARA<br>WAX | 12-in. paraffin differential albedo with 4 incident angles | CONC-12<br>CON12<br>CONC12 | 12-in. concrete differential albedo with 4 incident angles | SPEC<br>SPE<br>MIR | |
| CARBON<br>GRAPHITE<br>C | 200-cm carbon differential albedo with 4 incident angles | CONC-16<br>CON16<br>CONC16 | 16-in. concrete differential albedo with 4 incident angles | PERIODIC<br>PERI<br>PER | Periodic boundary condition |
| ETHYLENE<br>POLY<br>CH2 | 12-in. polyethylene differential albedo with 4 incident angles | CONC-24<br>CON24<br>CONC24 | 24-in. concrete differential albedo with 4 incident angles | | |

Table C4.4.15 (continued)

## Summary of geometry data

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM
See Sects. F11.4.4, F11.5.1.2, F11.5.6, and F11.5.7.

GEOMETRY REGION DATA consists of SIMPLE GEOMETRY REGION DATA and EXTENDED GEOMETRY REGION DATA.

ENTER GEOMETRY REGION DATA IN THE FOLLOWING FORM:

OPTIONAL GLOBAL SPECIFICATION
UNIT n
OPTIONAL GEOMETRY COMMENT
SIMPLE GEOMETRY REGION DATA
and/or
EXTENDED GEOMETRY REGION DATA
* * * *

Table C4.4.15 (continued)

## Summary of geometry data (continued)

GEOMETRY (region) (cont.)

ENTER EXTENDED GEOMETRY DATA IN THE FOLLOWING FORM:

fgeom ref. ID bias ID thickness per region origin coordinates nreg

EXTENDED GEOMETRY REGION INPUT DATA REQUIREMENTS

| TYPE OF DATA | TYPE 1 DATA | TYPE 2 DATA | TYPE 3 DATA |
|---|---|---|---|
| fgeom | ARRAY<br>CORE<br>COREBDY<br>COREBNDS<br>COREBOUN | HOLE | REPLICATE<br>REFLECTOR |
| Ref. ID | Array no. | Emplaced unit number | Mixture no. in generated regions |
| Bias ID | Omit for ARRAY | Omit | First bias ID |
| Thick./reg. | Omit | Omit | Variable* |
| Origin coord. | Enter the X Y Z coord. of most neg. pt. of array | Enter the X Y Z coord. of origin | Omit |
| nreg | Omit | Omit | No. of regions to be generated |

*The number of dimensions to be entered is the same as the region preceding the replicate or reflector specification because the generated regions have that shape. The value of the dimensions is the thickness of each generated region of material on that surface.

Table C4.4.15 (continued)

**Summary of mixing table data**

MIXTURES — Format: READ MIXT xsec parameters END MIXT

Note: In CSAS, only xsec parameters should be entered.
KENO V.a mixing table data consists of "xsec parameters" and a "mixing table." The "mixing" table is used to define the materials used in the KENO V.a problem. These data are automatically created by CSAS and provided to KENO V.a. It cannot be overridden.
See Sects. F11.4.10 and F11.5.5.

XSEC PARAMETERS — consists of keywords and associated values.
These parameters, if entered, need be entered only once.

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| SCT= | 1 | No. of discrete scattering angles<br>0 is isotropic<br>1 is P1<br>2 is P3<br>3 is P5 |
| EPS= | .00003 | Cross-section message cutoff value<br>Use to suppress message No. K5-60 |

Table C4.4.15 (continued)

## Summary of plot data

PLOT Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made. See Sects. F11.4.11 and F11.5.9

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| TTL= | Prob. title | delim ptitl delim delim is a one-character delimiter that signals the beginning and end of the title. ptitl is the plot title (max. 132 char.) |
| PIC= | MAT | Type of picture: MIXTURE, UNIT NO. or BIAS ID NO. |
| | | MIXTURE ---------- MAT |
| | | MIX |
| | | MIXT |
| | | MIXTURE |
| | | MEDI |
| | | MEDIA |
| | | UNIT NO. ---------- BOX |
| | | BOXT |
| | | BOXTYPE |
| | | UNT |
| | | UNIT |
| | | UNITTYPE |
| | | BIAS ID NO. ------- IMP |
| | | BIAS |
| | | BIASID |
| | | WTS |
| | | WEIG |
| | | WEIGHTS |
| | | WGT |
| | | WGTS |
| XUL= | Prev. plot | X coord. of upper left corner of plot |
| YUL= | Prev. plot | Y coord. of upper left corner of plot |
| ZUL= | Prev. plot | Z coord. of upper left corner of plot |
| XLR= | Prev. plot | X coord. of lower right corner of plot |
| YLR= | Prev. plot | Y coord. of lower right corner of plot |
| ZLR= | Prev. plot | Z coord. of lower right corner of plot |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| UAX= | Prev. plot 0 IF VAX OR WAX is read | X component of direction cosine for the AX axis of the plot (across) |
| VAX= | Prev. plot 0 IF UAX OR WAX is read | Y component of direction cosine for the AX axis of the plot (across) |
| WAX= | Prev. plot 0 IF UAX OR VAX is read | Z component of direction cosine for the AX axis of the plot (across) |
| UDN= | Prev. plot 0 IF VDN OR WDN is read | X component of direction cosine for the DN axis of the plot (down) |
| VDN= | Prev. plot 0 IF UDN OR WDN is read | Y component of direction cosine for the DN axis of the plot (down) |
| WDN | Prev. plot 0 IF UDN OR VDN is read | Z component of direction cosine for the DN axis of the plot (down) |
| DLX= | | Horizontal spacing between points on plot |
| DLD= | | Vertical spacing between points on plot |
| NAX= | | No. of intervals to be printed across page |
| NDN= | | No. of intervals to be printed down page |
| LPI= | 8 | Lines per inch printed down the page |
| NCH= | CHRS" | Delim CHRS delim, a one character delimiter signals the beginning and end of the character string |
| RUN= | YES | YES allows the problem to execute NO terminates problem after data checking |
| PLT= | YES | YES allows the plot(s) to be made NO allows reading the plot data without making a plot |
| SCR= | NO | NO specifies a printer plot. If YES, a .gif file is generated for each plot |
| CLR= | | Use only with SCR=YES to change colors. Input 4 integers: 1st is index into color table, next 3 are the red, green, and blue color values |

PLOT ORIGIN:

(1) SINGLE UNIT - coincides with origin of geometry description.

(2) BASE ARRAY - at the most negative point of the array (lower left-hand back corner of the global array).

(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

"default values of CHRS are given below:

MEDIA 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
CHRS 1 2 3 4 5 6 7 8 9 A B C D E F G H I J K L M

MEDIA 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
CHRS N O P Q R S T U V WX Y Z # , $ - + ) |

MEDIA 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58
CHRS £ > : ; . - % * " = ! ( @ < / 0

Table C4.4.15 (continued)

## Summary of starting data

START Format: READ START enter start data here END START
The default value of start type is zero. See Sect. F11.4.8.

| START TYPE | REQUIRED DATA | OPTIONAL DATA | STARTING DISTRIBUTION |
|---|---|---|---|
| 0 | None | NST<br>XSM<br>XSP<br>YSM<br>YSP<br>ZSM<br>ZSP<br>RFL<br>PSP | Uniform |
| 1 | NST | XSM<br>XSP<br>YSM<br>YSPO<br>ZSM<br>ZSP<br>RFL<br>PSP | Cosine |
| 2 | NST<br>NXS<br>NYS<br>NZS<br>FCT | XSM<br>XSP<br>YSM<br>YSP<br>ZSM<br>ZSP<br>RFL<br>PSP | Cosine with fraction in specified unit |
| 3 | NST<br>TFX<br>TFY<br>TFZ<br>NXS<br>NYS<br>NZS | KFS<br>PSP | Spike |
| 4 | NST<br>TFX<br>TFY<br>TFZ<br>NBX | KFS<br>PSP | Multiple spikes |
| 5 | NST<br>NBX | PSP | In specified units |
| 6 | NST<br>TFX<br>TFY<br>TFZ<br>LNU* | NXS<br>NYS<br>NZS<br>KFS<br>PS6<br>PSP | Arbitrary points |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| NST= | 0 | Start type |
| TFX= | 0.0 | X coordinate |
| TFY= | 0.0 | Y coordinate |
| TFZ= | 0.0 | Z coordinate |
| NXS= | 0 | X index of unit pos. |
| NYS= | 0 | Y index of unit pos. |
| NZS= | 0 | Z index of unit pos. |
| KFS= | | Fission spectra |
| LNU= | 0 | Number of last neutron |
| NBX= | 0 | Source unit number |
| FCT= | 0 | Fraction |
| XSM= | -X | -X of source cuboid |
| XSP= | +X | +X of source cuboid |
| YSM= | -Y | -Y of source cuboid |
| YSP= | +Y | +Y of source cuboid |
| ZSM= | -Z | -Z of source cuboid |
| ZSP= | +Z | +Z of source cuboid |
| RFL= | NO | Start in reflector |
| PS6= | NO | Print start 6 input |
| PSP= | NO | Print starting points |

*LNU must be the last entry for each set of start 6 data. The LNU of each successive set of data must be larger than the last.

### C4.4.10.1 Search Type Specification

The basic search data to define the type of search and various search parameters that set limits for the search are described in this section. These data are listed in Table C4.4.16.

Table C4.4.16 Outline of search type specification

| Entry number | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | Search descriptor | OPTIMUM<br>CRITICAL<br>MINIMUM | Initiates a search for the maximum value of k-effective<br>Initiates a search for a specified value of k-effective. Initiates a search for the minimum value of k-effective |
| 2 | Search type | PITCH<br>DIMENSION<br>CONCENTRATION | Vary the pitch<br>Vary one or more dimensions in one or more regions<br>Vary the concentration of one or more standard compositions in one or more mixtures |
| 3 | Optional search parameters | | Optional search parameters allow changing default values. Any or all may be entered in any order |
| 3a | No. of search passes | PAS= | Enter the keyword PAS= followed by the desired number of search passes. default=10 |
| 3b | No. of search parameters | NPM= | Enter the keyword NPM= followed by the number of search parameters. Present capability is limited to 1. default=1 |
| 3c | Search convergence tolerance | EPS= | Enter the keyword EPS= followed by the desired convergence tolerance. Default=0.005 |
| 3d | Desired value of k-effective | KEF= | Enter the keyword KEF= followed by the desired value of k-effective. The default value is 1.000. DO NOT ENTER FOR OPTIMUM OR MINIMUM SEARCHES |
| 3e | Maximum allowed pitch | MAXPITCH= | Enter the keyword MAXPITCH= followed by the maximum allowed pitch for a search whose search type, entry 2 above, is PITCH. The default value is the pitch corresponding to -5.0 times the parameter at the minimum possible pitch |
| 3f | Minimum allowed pitch | MINPITCH= | Enter the keyword MINPITCH= followed by the minimum allowed pitch for a search whose search type, entry 2 above, is PITCH. The default value is the minimum possible pitch (i.e., the pitch at which the shapes in the array touch) |
| 4 | Additional search data | MORE | Enter the delimiter MORE. This delimiter ends the optional search commands and initiates the auxiliary search commands found in Table C4.4.17. |

The desired search type must be specified in the search data before the individual search commands can be entered. The desired search type is specified only once in a problem. The data specifications are listed below.

1. Search descriptor

   Enter a word (OPTIMUM, CRITICAL, MINIMUM) to define the basic search mode. A default value is not supplied. The three basic search modes are:

   OPTIMUM initiates a search for the maximum value of k-effective.

   CRITICAL initiates a search for a specified value of k-effective.

   MINIMUM initiates a search for the minimum value of k-effective.

2. Search type

   Enter a word (PITCH, DIMENSION, CONCENTRATION) to specify the variable that is changed during the search. A default value is not supplied. The variables are:

   PITCH causes the pitch (center-to-center spacing in the x and y directions) to vary according to the search parameters.

   DIMENSION causes the dimensions in one or more geometry regions to vary according to the search parameters.

   CONCENTRATION causes the concentration of one or more standard composition materials to vary in one or more mixtures.

3. Optional search parameters

   Optional search parameters are available for changing default values. They are specified by entering a keyword followed by the desired value. The optional search parameters are listed below.

3a. PAS= Number of search passes

   The number of search passes is the maximum number of times the search will change data and calculate the corresponding k-effective. The default is 10.

3b. NPM= Number of search parameters

   The number of search parameters is not fully implemented. The default value is 1 and should not be overridden.

3c. EPS= Search convergence tolerance

The search convergence tolerance is the search convergence criteria. OPTIMUM and MINIMUM searches are terminated when a k-effective is calculated that is within EPS of the optimum or minimum respectively. A CRITICAL search is terminated when a calculated k-effective is within EPS of the target or desired k-effective. The default value is 0.005.

3d. KEF= Desired value of k-effective

This information is entered ONLY for a CRITICAL search. The default value is 1.000.

3e. MAXPITCH= Maximum allowed pitch

The maximum allowed pitch (center-to-center spacing in the x and y directions) is specified using this keyword. It applies to a pitch search type and causes the search to terminate if the specified pitch is exceeded. The default value is the pitch corresponding to -5 times the parameter at the minimum pitch.

3f. MINPITCH= Minimum allowed pitch

The minimum allowed pitch (center-to-center spacing in the x and y directions) is specified using this keyword. It applies to a PITCH search type and causes the search to terminate if the pitch needs to be smaller than MINPITCH. The default value is the pitch at which the pins in the array are touching.

4. More additional search data will follow.

The word MORE terminates the optional search parameters and initiates the individual search commands. Do not enter MORE unless individual search commands are to be entered.

### C4.4.10.2 Auxiliary Search Commands and Constraints

Auxiliary search commands are entered ONLY if MORE, item 4, of the search type specification data were entered (see Table C4.4.16). Individual search commands are used to specify search constraints and to communicate to the search program. Searches can alter geometric dimensions (PITCH or DIMENSION Search) or alter concentrations (CONCENTRATION Search). A PITCH or DIMENSION search may require the user to specify the units that will be altered, the regions that will be altered within those units, and the faces or surfaces of those regions that will be altered. A CONCENTRATION search requires the user to specify the mixture, standard composition name, and the search constant for the component being altered.

## Table C4.4.17 Outline of auxiliary search commands and constraints

| Entry number | Keyword name | Type of Data | Comments |
|---|---|---|---|
| GENERIC SEARCH DATA | | | -May be used with all Search Types- |
| 1a | ALTER<br>CHANGE<br>MODIFY | Begin a new search command | These words are used to specify that modifications will be made to the geometry or concentration according to subsequent commands (entries 3 through 6 as required to specify the desired changes) |
| 1b | MAINTAIN | Begin a new search command | The spacing (thickness) of the specified geometry regions will be maintained when the interior regions grow or shrink |
| 1c | KEEP<br>HOLD | Begin a new search command | This command resets the specified geometry to the original input specifications |
| 2 | PAR= | Parameter number | Enter the parameter number that the current command (ALTER, MAINTAIN, KEEP) applies to. The present capability is limited. Default=1 and should not be changed. |
| 3 | +CON= | Maximum constraint | Enter the maximum constraint for the current parameter. default=+10E10 |
| 4 | -CON= | Minimum constraint | Enter the minimum constraint for the current parameter. default=-10E10 |
| PITCH & DIMENSION SEARCH DATA | | | -Defines Geometric Changes- |
| 5 | UNIT= | Unit the current command applies to | Enter the unit in which regions are to be altered |
| 6 | REGION= | First region to be altered in the unit | Enter the first or only region in the unit that the search constants (entry 8a, b, c and/or d) apply to. Default is the first region |
| 7 | TO | Last region to be altered in the unit | Enter the last region in the unit that the search constants (entry 8a, b, c and/or d) apply to. Default is the first region<br>NOTE: Entry 6 must be entered in order to alter a single region in a unit. Entries 6 and 7 must both be entered in order to alter more than one region in a unit |
| 8 | ALL= | Search constant for all surfaces (faces) of the region(s) | Enter a value for the search constants for the specified regions. This value will be applied to all surfaces of the region(s) |
| 8a | +X= | Search constant for +X face of cuboid | Enter a value for the search constant for the +X face of a cuboid |
| | -X= | Search constant for -X face of cuboid | Enter a value for the search constant for the -X face of a cuboid |
| | +Y= | Search constant for +Y face of cuboid | Enter a value for the search constant for the +Y face of a cuboid |
| | -Y= | Search constant for -Y face of cuboid | Enter a value for the search constant for the -Y face of a cuboid |
| | +Z= | Search constant for +Z face of cuboid | Enter a value for the search constant for the +Z face of a cuboid |
| | -Z= | Search constant for -Z face of cuboid | Enter a value for the search constant for the -Z face of a cuboid |
| 8b | RADIUS= | Search constant for radius | Enter a value for the search constant for the radius of a sphere or a cylinder |
| 8c | +HEIGHT= | Search constant for + height | Enter a value for the search constant for the + height of a cylinder |
| | -HEIGHT= | Search constant for - height | Enter a value for the search constant for the - height of a cylinder |
| 8d | CHORD= | Search constant for chord | Enter a value for the search constant for the chord face of a hemisphere or hemicylinder |

Table C4.4.17 (continued)

| CONCENTRATION SEARCH DATA | | -Define concentration changes- | |
|---|---|---|---|
| 9 | MIX= | Search constant for mixture | Enter the mixture number containing the standard composition to be changed. |
| 10 | SCNAME= | Search constant for Standard Composition | Enter the standard composition name whose density is to be changed. |
| 11 | FACTOR= | Search proportionality constant | Enter the value of the search constant for the concentration search. |

The data comprising the auxiliary search commands are listed in Table C4.4.17. All data except items 1a, 1b, and 1c are keyworded (i.e., the data are entered by specifying a keyword, followed by a value). An explanation of each individual search command follows the table.

1. Command definition. A command definition tells the code what action is to be taken. A new search command is initiated whenever an item 1a through 1c is encountered. The code will vary the geometry according to subsequent commands.

1a. ALTER
CHANGE
MODIFY — Alter geometry regions. These words specify that modifications will be made to the geometry according to subsequent commands.

1b. MAINTAIN — Maintain the thickness. The thickness of the specified geometry region(s) will be maintained when the interior regions grow or shrink (i.e., the specified region will grow or shrink in conjunction with the interior region in such a way as to maintain the original distance between the two regions). This means that the original thickness of the region is preserved. For instance, the inner radius of a pipe can be altered and the wall thickness can be preserved by applying the MAINTAIN command to the region defining the outer radius of the pipe.

1c. KEEP
HOLD — Keep the original specification. This command causes the specified geometry region(s) to be reset to their original input value for every search pass. Therefore they go through the entire search process unchanged.

2. PAR= — Parameter number. The search parameter number is not fully implemented. The default number is 1 and should not be overridden.

3. +CON= — Maximum constraint. Enter the maximum value you wish to allow the search parameter to obtain. The maximum constraint must be larger than the minimum constraint. See Sect. C4.5.6.2b for additional information on how to determine constraints. For a DIMENSION or PITCH search the default value of the maximum constraint is $10^{11}$. For a CONCENTRATION search the default value of the maximum constraint is as follows:
+CON = min( -1/FACTOR ), if any FACTOR < 0
+CON = -5*( -CON ), if all FACTOR > 0

NOTE: For a PITCH search, the maximum constraint is redefined and is calculated from the data entered for MAXPITCH. +CON should not be entered if a value was entered for MAXPITCH. If constraints, +CON= and/or -CON= are not entered as data, the code computes the minimum constraint corresponding to the pins touching, and the maximum constraint is then negative five times the magnitude of the minimum constraint.

4. -CON= — Minimum constraint. Enter the minimum value you wish to allow the search parameter to obtain. The minimum constraint must be smaller than the maximum constraint but need not be a negative number. See Sect. C4.5.6.2b for additional information on how to determine constraints. The default value of the minimum constraint for a dimension search is $-10^{11}$. The default value of the minimum constraint for a pitch search is redefined to correspond to the pins in the lattice touching. For a CONCENTRATION search the default value of the minimum constraint is as follows:

-CON = -5( +CON ), if all FACTOR < 0
-CON = max( -1/FACTOR ), if any FACTOR > 0

5. UNIT= Geometry unit number. This is the geometry unit to which the previously entered command definition (item 1a, 1b, or 1c) is applied. Items 5, 6, and 7 specify the region(s) within the unit and the surfaces of the region(s) to be altered.

6. REGION= First region to be altered. This is used to specify the first or only region in the unit (specified by item 5) that is to be altered according to the search command (item 1a, 1b, or 1c). The region(s) are altered according to the search constants (items 8a, 8b, 8c, and/or 8d).

7. TO Last region to be altered. This item is entered to specify the last region to be altered, starting with the region specified by REG=. For example, assume unit 3 contains 8 regions and you wish to make changes to regions 4, 5, 6, 7, and 8. These regions are identified by entering the following data. UNIT=3 REG=4 TO 8

8. Geometric search constants. A search constant is the proportionality factor utilized to alter a geometry region. A search constant must be entered for each surface of a region that is to be altered. A nonzero search constant will cause the region dimension for that surface to be changed. A search constant of 0.0 will cause the region dimension to remain unchanged. The default value of the search constant is 0.0.

8a. ALL= Search constant for all surfaces. All of the surfaces in a region are altered simultaneously by using this search command.

8b. +X= Search constant for +X face. This parameter is used to specify the value of the search constant for the +X face of a cuboid.

8b. -X= Search constant for -X face. This parameter is used to specify the value of the search constant for the -X face of a cuboid.

8b. +Y= Search constant for +Y face. This parameter is used to specify the value of the search constant for the +Y face of a cuboid.

8b. -Y= Search constant for -Y face. This parameter is used to specify the value of the search constant for the -Y face of a cuboid.

8b. +Z= Search constant for +Z face. This parameter is used to specify the value of the search constant for the +Z face of a cuboid.

8b. -Z= Search constant for -Z face. This parameter is used to specify the value of the search constant for the -Z face of a cuboid.

NOTE: If it is desirable to change all the faces of a cuboid except the -z face by some amount (search constant of 1.0), items 8a and 8b can be used together as follows: ALL=1.0 -Z=0.0. This is the same as entering +X=1.0 -X=1.0 +Y=1.0 -Y=1.0 +Z=1.0. If both z faces are to remain unchanged, items 8a and 8b can be entered as: ALL=1.0 +Z=0.0 -Z=0.0 or as +X=1.0 -X=1.0 +Y=1.0 -Y=1.0.

8c. RADIUS= Search constant for radius. This parameter is used to specify the value of the search constant for the radius of a sphere, hemisphere, cylinder, or hemicylinder.

8c. +HEIGHT= Search constant for +height. This parameter is used to specify the value of the search constant for the +height of a cylinder or hemicylinder.

8c. -HEIGHT= Search constant for -height. This parameter is used to specify the value of the search constant for the -height of a cylinder or hemicylinder.

8d. CHORD= Search constant for chord. This parameter is used to specify the value of the search constant for the chord of a hemisphere or hemicylinder.

9. MIX= Search constant for mixture. This parameter is the mixture number containing the standard composition that is to be changed during the search.

10. SCNAME= Search constant for the standard composition name. This parameter is the standard composition name associated of the material that is to be changed during the search. Only compositions listed in Sect. M8.2 are allowed. Standard compositions beginning with SOLN and ARBM cannot be altered directly.

NOTE: If the standard composition name specified in the Material Information Data begins with SOLN and the Concentration Search Data specifies SCNAME=UO2(NO3)2, the amount of UO2(NO3)2 in the solution is altered but the amount of H2O and nitric acid is not altered during the search. The resulting mixture, when the search is finished, may no longer meet the criteria associated with the SOLN specification.

11. FACTOR= Concentration search factor used to specify the value of the search constant used in the concentration search. It is a proportionality factor used to alter the specified mixture standard composition. A search constant must be entered for each standard composition that is altered. A non-zero search constant will cause the concentration of the associated standard composition to be altered. The default value of the search constant is 1.0. A set of concentration search data consists of the mixture to be altered, the standard composition to be altered in the mixture, and the search factor. Keywords are used to enter the data. Each set of data consists of items 9, 10 and 11. The keywords used in this data may be entered using terse notation.

## C4.5 NOTES FOR CSAS USERS

This section provides tips concerning the use of CSAS and its associated sequences. Section M7.5 provides information about the use of the Material Information Processor, and Sect. F11.7 provides assistance about the use of the KENO V.a data required by CSAS.

### C4.5.1 CSAS MODULES

Several CSAS modules are contained in the CSAS family. These modules enable the user to generate and save working format cross sections for use with KENO IV, XSDRNPM, or KENO V.a, or a MORSE/KENO cross-section library for use with MORSE or KENO V.a. Therefore, the user can run different codes using the same cross-section library for comparison purposes. A list of the available modules is given below. Table C4.5.1 lists the I/O devices utilized by CSAS, and Table C4.5.2 lists the type of I/O files utilized by the functional modules invoked by CSAS.

The CSAS input data allow the user to define optional I/O files. The CSAS optional parameter data, Sect. C4.4.8, allow the user to define the following optional files: (1) the unit from which XSDRNPM may read fluxes, (2) the unit on which XSDRNPM may write fluxes, and (3) the unit on which ICE may write an ANISN library. The KENO V.a input data used by CSAS allow the user to define optional logical units used by KENO V.a. See "Logical Unit Numbers" in Sect. F11.4.3. The unit on which a KENO V.a restart data file is written should not be set to 95.

CSASN — Runs BONAMI and NITAWL-II and generates a working format cross-section library on unit 4. The user may save that library for future use or stack a KENO V.a or XSDRNPM problem after it with the input library specified as unit 4. If the user wishes to exercise XSDRNPM options that are not available in CSAS (such as a search or adjoint calculation), an XSDRNPM problem can be stacked immediately after the CSASN problem. When a KENO V.a problem is stacked after a CSASN problem, the parameter LIB=4 should be entered in the KENO V.a parameter data and a mixing table must be entered in order to utilize the working format library from CSASN.

CSAS1X — Runs BONAMI, NITAWL-II, and XSDRNPM. A cell-weighted working format cross-section library is written on unit 3. Note that any mixture utilized in the cell cannot be used in a subsequent KENO V.a calculation because the weighting applied to those mixtures is for the purpose of generating the cell-weighted cross section. This difficulty is circumvented by adding the same mixtures to the CSAS data with different mixture numbers. To utilize the cell-weighted library in a KENO V.a problem following a CSAS1X problem, LIB=3 should be entered in the KENO V.a parameter data, a mixing table must be entered, and the cell-weighted mixture can be used in the geometry.

CSASI — Runs BONAMI, NITAWL-II, and ICE. A working format library is written on unit 2, and a MORSE/KENO mixture cross-section library is written on unit 14. To utilize the MORSE/KENO library in a subsequent KENO V.a problem, the parameter XSC=14 must be entered in the KENO V.a parameter data, and the LIB= parameter should be omitted. The mixing table must be omitted, and the mixtures utilized in the geometry must match the mixtures specified in the CSASI data.

Table C4.5.1 Description of units utilized by the Criticality Safety Analysis Sequence No. 4 control module

| Unit number | Type of data | Creating module | User module | Type of file |
|---|---|---|---|---|
| 5 | Input | DRIVER | CSAS | Sequential |
| 8 | Scratch file | CSAS | CSAS | Direct or random access data |
| 11 | Short master cross-section library | CSAS | BONAMI | Sequential |
| 16 | Scratch file | CSAS | CSAS | Sequential |
| 17 | Scratch file | CSAS | CSAS | Sequential |
| 18 | Scratch file | CSAS | CSAS | Sequential |
| 19 | Scratch file | CSAS | CSAS | Sequential |
| 70 | User-supplied cross sections | | CSAS | AMPX master format library |
| 81 | 16-group cross sections | | CSAS | AMPX master format library |
| 82 | 27-group cross sections | | CSAS | AMPX master format library |
| 83 | 123-group cross sections | | CSAS | AMPX master format library |
| 84 | 218-group cross sections | | CSAS | AMPX master format library |
| 85 | 22 neutron-18 gamma cross sections | | CSAS | AMPX master format library |
| 86 | 18-gamma cross sections | | CSAS | AMPX master format library |
| 87 | 27-group cross sections for burnup | | CSAS | AMPX master format library |
| 88 | 27 neutron-18 gamma cross sections | | CSAS | AMPX master format library |
| 89 | Standard Composition Library | | CSAS | Direct- or random-access data file |
| 90 | Search data | CSAS | MODIFY | Direct or random access data |
| 92 | ICE input data | CSAS | ICE | Sequential |
| 95 | KENO V.a restart data | CSAS<br>MODIFY<br>KENO V.a | KENO V.a<br>MODIFY | Sequential |
| 96 | BONAMI input data | CSAS | BONAMI | Sequential |
| 97 | NITAWL-II input data | CSAS | NITAWL-II | Sequential |
| 98 | XSDRNPM input data | CSAS | XSDRNPM | Sequential |
| 99 | Printed output | CSAS<br>MODIFY | CSAS<br>MODIFY | Sequential |

## Table C4.5.2 Description of units used by the functional modules of CSAS

| Unit No. | Module | Action | Type of file |
|---|---|---|---|
| 1 | BONAMI | WRITE | Microscopic master working format cross sections |
| | NITAWL-II | READ | Microscopic master working format cross sections |
| 2 | ICE | WRITE | Macroscopic working format cross sections |
| 3 | XSDRNPM | WRITE | Microscopic cell-weighted working format cross sections |
| | ICE | READ | Microscopic cell-weighted working format cross sections |
| | KENO V.a | READ | Microscopic cell-weighted working format cross sections |
| 4 | NITAWL-II | WRITE | Microscopic working format cross sections |
| | ICE | READ | Microscopic working format cross sections |
| | KENO V.a | READ | Microscopic working format cross sections |
| | XSDRNPM | READ | Microscopic working format cross sections |
| 6 | BONAMI | WRITE | Printed output |
| | ICE | WRITE | Printed output |
| | KENO V.a | WRITE | Printed output |
| | NITAWL-II | WRITE | Printed output |
| | XSDRNPM | WRITE | Printed output |
| 8 | BONAMI | WRITE/READ | Direct- or random-access scratch files |
| | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 9 | BONAMI | WRITE/READ | Direct- or random-access scratch files |
| | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct- or random-access scratch files |
| | NITAWL-II | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 10 | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO V.a | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 11 | BONAMI | READ | Short AMPX master format cross sections |
| 14 | ICE | WRITE | Monte Carlo-formatted macroscopic cross sections |
| | KENO V.a* | READ/WRITE | Monte Carlo-formatted macroscopic cross sections |
| 16 | KENO V.a | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Angular flux file |
| 17 | XSDRNPM | WRITE/READ | Sequential scratch file |
| 18 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 19 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 79 | KENO V.a | READ | KENO V.a albedo file |
| 80 | KENO V.a | READ | KENO V.a weighting file |
| 92 | ICE | READ | Binary input file |
| 95 | KENO V.a | READ | Binary input file |
| 96 | BONAMI | READ | Binary input file |
| 97 | NITAWL-II | READ | Binary input file |
| 98 | XSDRNPM | READ | Binary input file |

*KENO V.a always reads from unit 14. If microscopic input cross sections are used, KENO V.a prepares macroscopic cross sections and writes them on unit 14 to be read later.

CSASIX — Runs BONAMI, NITAWL-II, XSDRNPM, and ICE. A working format library is written on unit 2, and a MORSE/KENO mixture cross-section library, including the cell-weighted mixture 500, is written on unit 14. To utilize the MORSE/KENO library in KENO V.a, the parameter XSC=14 must be entered in the KENO V.a parameter data, and the LIB= parameter should be omitted. The mixing table must be omitted, and the mixtures utilized in the geometry must match the mixtures specified in the CSASIX data.

CSAS25 — Runs BONAMI, NITAWL-II, and KENO V.a. A working format library is written on unit 4, and a MORSE/KENO mixture cross-section library is written on unit 14. Either or both of these libraries can be saved for future use. Because KENO V.a data are entered as part of the CSAS25 data, LIB= and XSC= should not be specified in the KENO V.a parameter data, and the mixing table should be omitted from the KENO V.a data. All KENO V.a options are available through CSAS25.

CSAS2X — Runs BONAMI, NITAWL-II, XSDRNPM, and KENO V.a. A working format library is written on unit 3, and a MORSE/KENO mixture cross-section library is written on unit 14. The cell-weighted mixture 500 is included on unit 14. Either or both of these libraries can be saved for future use. Because KENO V.a data are entered as part of the CSAS25 data, LIB= and XSC= should not be specified in the KENO V.a parameter data, and the mixing table should be omitted from the KENO V.a data. All KENO V.a options are available through CSAS2X. To process cross sections and run KENO V.a, CSAS25 should be used unless use of a cell-weighted mixture in KENO V.a is planned.

CSAS4 — Runs BONAMI, NITAWL-II, KENO V.a, and MODIFY. Use CSAS to perform an automatic search on dimensions or pitch. CSAS is an iterative procedure in which the KENO V.a geometry data is modified and KENO V.a is run until the convergence criteria is met or the problem runs out of time. A working format library is written on unit 4, and a MORSE/KENO mixture cross-section library is written on unit 14. Either or both of these libraries can be saved for future use. Because KENO V.a data are entered as part of the CSAS data, LIB= and XSC= should not be specified in the KENO V.a parameter data, and the mixing table should be omitted from the KENO V.a data. All KENO V.a options are available through CSAS. The search data follow the KENO V.a data.

CSAS4X — Runs BONAMI, NITAWL-II, XSDRNPM, KENO V.a, and MODIFY. Use CSAS4X for search problems where the geometry utilizes a cell-weighted cross section and an automatic search on the KENO V.a geometry dimensions or pitch is desired. The cell-weighted mixture or geometries cannot be modified as part of the search problem. CSAS4X is an iterative procedure in which KENO V.a geometry data are modified and KENO V.a is run until the convergence criteria is met or the problem runs out of time. A working format library is written on unit 3, and a MORSE/KENO mixture cross-section library is written on unit 14. The cell-weighted mixture 500 is included on unit 14. Either or both of these libraries can be saved for future use. Because the KENO V.a data is entered as part of the CSAS data, LIB= and XSC= should not be specified in the KENO V.a parameter data, and the mixing table should be omitted from the KENO V.a data. All KENO V.a options are available through CSAS4X. The search data follows the KENO V.a data.

## C4.5.2 AUTOMATIC CROSS-SECTION PROCESSING FOR CRITICALITY CALCULATIONS

The CSAS family of control modules can prepare an AMPX working format problem-dependent cross-section library for use by KENO V.a. CSAS25, CSAS2X, CSAS4, and CSAS4X can be used to prepare an AMPX working format cross-section library and run KENO V.a without the necessity of having to enter the mixing table data in KENO V.a. Examples are given in Sects. C4.5.2.1 through C4.5.2.4.

### C4.5.2.1 Run KENO V.a using CSAS25 or CSAS4

CSAS25 and CSAS4 create a microscopic working format library and a mixing table that is passed to KENO V.a. CSAS25 executes KENO V.a and calculates k-effective for the problem. CSAS4 executes KENO V.a repeatedly to perform a search as specified in the search data.

EXAMPLE 1. CSAS25 - Determine the k-effective of a system.

Consider a problem consisting of four uranium metal cylinders that are 93.2% wt enriched, having a density of 18.76 g/cc. The cylinders are arranged in a 2 × 2 × 2 array. Each has a radius of 5.748 cm and a height of 10.765 cm. The center-to-center spacing in the horizontal plane (X-Y) is 13.74 cm and the vertical center-to-center spacing is 13.01 cm. The input data for this problem follow. The calculated k-effective for this critical experiment was 0.9934 ± 0.0045.

```
=CSAS25
SET UP 2C8 IN CSAS25
HANSEN-ROACH  INFHOMMEDIUM
URANIUM  1 DEN=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
KENO V.a SAMPLE PROBLEM 1  CASE 2C8 BARE
READ PARAMETERS  TME=1.0 FLX=YES FDN=YES FAR=YES
END PARAMETERS
READ GEOMETRY
CYLINDER 1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
END GEOMETRY
READ ARRAY  NUX=2 NUY=2 NUZ=2  END ARRAY
END DATA
END
```

EXAMPLE 2. CSAS - Determine the unit size for k-eff = 0.95.

Consider the problem of example 1 above. Determine the size of the uranium metal pieces that will yield a k-effective of 0.95 ± 0.005. Maintain the H/D ratio of the cylinders and the original center-to-center spacings. The input data are

```
=CSAS4
SET UP 2C8 IN CSAS25
HANSEN-ROACH  INFHOMMEDIUM
URANIUM  1 DEN=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
```

```
END COMP
CRITICAL DIMENSION SEARCH FOR CASE 2C8 BARE
READ PARAMETERS  TME=1.0 FLX=YES FDN=YES FAR=YES
END PARAMETERS
READ GEOMETRY
CYLINDER 1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
END GEOMETRY
READ ARRAY  NUX=2 NUY=2 NUZ=2  END ARRAY
END DATA
READ SEARCH   CRITICAL DIMENSION KEF=0.95 EPS=0.005
MORE
ALTER  UNIT=1 REG=1  ALL=1.0
+CON=0.0  -CON=-0.25
END SEARCH
END
```

The calculated results for this search are the following:

```
SEARCH PASS1  KEFF=9.95561E-01  +OR - 4.36660E-03  PARAMETER=0.00000E+00
SEARCH PASS2  KEFF=6.69976E-01  +OR - 3.95561E-03  PARAMETER=-2.50000E-01
SEARCH PASS3  KEFF=9.46110E-01  +OR - 4.09973E-03  PARAMETER=-3.49840E-02
```

The final configuration for region 1 is:

```
CYLINDER  1 1  5.5469  5.1942  -5.1942
```

EXAMPLE 3. CSAS25 - Determine the k-effective of an array of fuel assemblies.

Consider a 4 × 4 × 1 array of fuel assemblies in a square aluminum cask. Each assembly consists of a 17 × 17 × 1 array of zirconium-clad, 2.35%-enriched $UO_2$ fuel pins in a square pitched array. The $UO_2$ has a density of 9.21 g/cc. The pin diameter is 0.8 cm and is 366 cm long. The clad is 0.07 cm thick, and the pitch is 1.3 cm. Each fuel bundle is contained in a 0.65-cm-thick boral sheath. The bundles are separated by an edge-to-edge spacing of 1 cm. The array of bundles is centered in a 10-cm-thick aluminum cask whose inside dimensions are 0.5 cm beyond the outer edges of the fuel bundles. Determine the k-effective of a dry cask. Input data for this cask follow. The calculated k-effective for this system is 0.7940 ± 0.0035.

```
=CSAS25
SQUARE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2  1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
H2O  6 1 END
AL  5 1 END
```

```
END COMP
SQUAREPITCH  1.3 .8 1 3 .94 2 END
SQUARE FUEL CASK EXAMPLE USING DETAILED MOCKUP
READ PARAM  TME=12.0  NUB=YES FAR=YES GEN=153
END PARAM
READ GEOM
UNIT  1
COM='FUEL PIN'
CYLINDER 1 1 0.4  2P183.0
CYLINDER 2 1 0.47 2P183.07
CUBOID  3 1 4P0.65 2P183.07
UNIT  2
COM='FUEL ASSEMBLY'
ARRAY  1 2R-11.05 -183.07
REPLICATE  4 1 6*0.65 1
REPLICATE  3 1 6*0.5 1
GLOBAL
UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
ARRAY  2 2R-48.8 -184.22
REPLICATE 5 1 6R10.0 1
END GEOM
READ ARRAY ARA=1 NUX=17 NUY=17 NUZ=1 FILL F1 END FILL
ARA=2 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
END
```

EXAMPLE 4. CSAS - Determine the optimum pitch for an array of fuel assemblies

Consider the 4 × 4 × 1 array of fuel assemblies in a square aluminum cask described in example 3 above. Search for the assembly spacing that yields the maximum value of k-effective. Because the spacing between the assemblies is to be altered, the last region of unit 2 will be altered. In order to do this, the fuel assembly shroud cannot be defined as a replicate, because the code does not know the size of the array until later. In order to perform a pitch or dimension search, the code calculates the distance between the outermost region of the unit and the region interior to it. Therefore, the fuel assembly gap must be defined as a cuboid, and the water gap between assemblies can be entered as either a cuboid or a replicate. This search has been defined as an optimum dimension search. The calculated k-effectives for this system are listed after the following input data:

```
=CSAS4
SAMPLE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2  1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
```

```
AL 5 1 END
END COMP
SQUAREPITCH 1.3 .8 1 3 .94 2 END
SQUARE FUEL CASK EXAMPLE WITH DETAILED MOCKUP
READ PARAM TME=6.0 NUB=YES FAR=YES GEN=103
END PARAM
READ GEOM
UNIT 1
COM='FUEL PIN'
CYLINDER 1 1 0.4 2P183.0
CYLINDER 2 1 0.47 2P183.07
CUBOID 3 1 4P0.65 2P183.07
NIT 2
COM='FUEL ASSEMBLY'
ARRAY 1 2R-11.05 -183.07
CUBOID 4 1 4P11.7 2P183.72
REPLICATE 3 1 6*0.5 1
GLOBAL
UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
ARRAY 2 2R-48.8 -184.22
REPLICATE 5 1 6R10.0 1
END GEOM
READ ARRAY ARA=1 NUX=17 NUY=17 NUZ=1 FILL F1 END FILL
UARA=2 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
READ SEARCH OPTIMUM DIMENSION MORE
ALTER UNIT=2 REG=3 +X=1 -X=1 +Y=1 -Y=1
+CON=0.204918    -CON=-0.0409836
END SEARCH
END
```

The calculated results for this search are:

```
SEARCH PASS 1 KEFF=7.9334-01 + OR - 4.0541-03  PARAMETER= 0.00000E+00
SEARCH PASS 2 KEFF=8.4564-01 + OR - 4.0814-03  PARAMETER=-4.09836E-02
SEARCH PASS 3 KEFF=6.6002-01 + OR - 4.0889-03  PARAMETER= 2.04918E-01
SEARCH PASS 4 KEFF=6.9373-01 + OR - 3.9071-03  PARAMETER= 1.02459E-01
SEARCH PASS 5 KEFF=8.1896-01 + OR - 3.8378-03  PARAMETER=-2.04918E-02
CONVERGENCE WAS ACHIEVED ON PASS 2
```

The corresponding geometry for unit 2 was:

```
ARRAY NUMBER 1 1 11.05 -11.05 11.05 -11.05 183.07 -183.07
CUBOID          4 1 11.70 -11.70 11.70 -11.70 183.72 -183.72
CUBOID          3 1 11.70 -11.70 11.70 -11.70 183.72 -183.72
```

The highest k-effective is achieved on PASS number two, when the fuel assemblies are touching.

An alternative method of entering the search data for this problem is to define the search as an optimum pitch search and require that the spacing of the fuel pin cells remain unchanged. Because a pitch search is always conducted at the lowest array level, in this case the spacing between the pins in the fuel assembly (the outer region of unit 1), it is necessary to countermand the automatic alteration of the outer region of unit 1 by entering the KEEP command. Search constants must be entered for the X and Y faces to instruct the code to KEEP those dimensions unchanged. It is easier to use the keyword ALL, which applies the KEEP command to all of the faces since the optimum pitch search would have changed only the X and Y faces leaving the Z faces unchanged. The ALTER command must then be entered to instruct the search to alter the spacing between the fuel assemblies (region 3 of unit 2). Only the X and Y dimensions are to be altered; so the search constants are entered individually for those dimensions. It is acceptable to enter ALL=1 +Z=0 -Z=0 rather than +X=1 -X=1 +Y=1 -Y=1. It is not necessary to enter the constraints for an optimum pitch search (+CON= and -CON=). They were entered in this case to ensure that the alternative data more nearly duplicate the optimum dimension search data of example 4. These alternative search data are:

```
READ SEARCH  OPTIMUM PITCH MORE
KEEP UNIT=1 REG=3 ALL=1
ALTER UNIT=2 REG=3 +X=1 -X=1 +Y=1 -Y=1
END SEARCH
```

### C4.5.2.2 Run KENO V.a using CSAS2X or CSAS4X

CSAS2X and CSAS4X create a microscopic working format library and a mixing table that is passed to KENO V.a. The microscopic cross sections of the nuclides used in the unit cell geometry description are cell-weighted for use in mixture 500. Mixture 500 utilizes the cell-weighted cross sections that represent the heterogeneous system. CSAS2X executes KENO V.a and calculates k-effective for the problem. CSAS4X executes KENO V.a repeatedly to perform a search as specified in the search data. Cell-weighted mixtures cannot be modified during the search problem.

EXAMPLE 1. CSAS2X - Calculate the k-effective of an array of fuel assemblies using cell-weighted cross sections.

Consider the 4 × 4 × 1 array of fuel assemblies in a square aluminum cask described in Sect. C4.5.2.1, example 3. Calculate the k-effective of the system by using the cell-weighted mixture 500 to represent the fuel pins in the fuel assembly. Note that mixtures 1, 2, and 3, representing $UO_2$, zirconium, and water, respectively, are used in the unit cell description. Cell-weighting is applied to the microscopic cross sections that are used in the cell, making them incorrect for use elsewhere. Because water is used both inside the cell and between the fuel assemblies, an additional mixture, mixture 6, has been added to represent the water between the fuel assemblies. The input data for this problem follow. The calculated k-effective is 0.79400 ± 0.0030.

```
=CSAS2X
SQUARE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2  1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
```

```
AL  4 0.636 END
AL  5 1 END
H2O  6 1 END
END COMP
SQUAREPITCH  1.3 .8 1 3 .94 2 END
SAMPLE SQUARE FUEL CASK USING HOMOGENEOUS MOCKUP
READ PARAM  TME=12.0  NUB=YES FAR=YES GEN=153
END PARAM
READ GEOM
UNIT  2
COM='FUEL ASSEMBLY'
CUBOID  500 1 4P11.05 2P183.07
REPLICATE  4 1 6*0.65 1
REPLICATE  6 1 6*0.5 1
GLOBAL
UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
ARRAY  1 2R-48.8 -184.22
REPLICATE 5 1 6R10.0 1
END GEOM
READ ARRAY ARA=1 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
END
```

EXAMPLE 2. CSAS4X - Determine the optimum pitch for an array of fuel assemblies using cell-weighted cross sections.

Consider the 4 × 4 × 1 array of fuel assemblies in a square aluminum cask described using cell-weighted cross sections in example 3 above. Search for the assembly spacing that yields the maximum value of k-effective. The calculated k-effectives for this system are listed after the following input data.

```
=CSAS4X
SQUARE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2  1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
AL  5 1 END
H2O  6 1 END
END COMP
SQUAREPITCH  1.3 .8 1 3 .94 2 END
SAMPLE SQUARE FUEL CASK USING HOMOGENIZED MOCKUP
READ PARAM  TME=5.0  NUB=YES FAR=YES GEN=103
END PARAM
```

```
READ GEOM
UNIT 2
COM='FUEL ASSEMBLY'
CUBOID 500 1 4P11.05 2P183.07
REPLICATE 4 1 6*0.65 1
REPLICATE 6 1 6*0.5 1
GLOBAL
UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
ARRAY 1 2R-48.8 -184.22
REPLICATE 5 1 6R10.0 1
END GEOM
READ ARRAY ARA=1 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
READ SEARCH OPTIMUM PITCH END SEARCH
END
```

The calculated results for this search are:

```
SEARCH PASS 1 KEFF=7.9303-01 + OR - 3.6699-03 PARAMETER= 0.00000E+00
SEARCH PASS 2 KEFF=8.4365-01 + OR - 3.9363-03 PARAMETER=-4.09836E-02
SEARCH PASS 3 KEFF=6.5431-01 + OR - 3.9188-03 PARAMETER= 2.04918E-01
SEARCH PASS 4 KEFF=7.0925-01 + OR - 3.6657-03 PARAMETER= 1.02459E-01
CONVERGENCE WAS ACHIEVED ON PASS 2
```

The corresponding geometry for unit 2 was:

```
CUBOID 500 1 11.05 -11.05 11.05 -11.05 183.07 -183.07
CUBOID   4 1 11.70 -11.70 11.70 -11.70 183.72 -183.72
CUBOID   6 1 11.70 -11.70 11.70 -11.70 184.22 -184.22
```

Thus, the highest k-effective is achieved when the fuel assemblies are touching.

### C4.5.2.3 Run XSDRNPM using CSAS1X

XSDRNPM can be utilized to calculate the eigenvalue of a system without the necessity of manually preparing the cross sections and mixing table data by executing CSAS1X as shown in example 1 below.

EXAMPLE 1. CSAS1X - Calculate the eigenvalue of a system.

This example is a critical experiment of plutonium nitrate solution in a spherical stainless steel tank reflected by water. The plutonium is 95.4% wt $^{239}Pu$, the remainder being $^{240}Pu$. The fuel is a 0.52-M nitric acid solution of plutonium-nitrate, containing 24.4 g of heavy metal per liter of solution. Optional parameters were entered to decrease the number of mesh intervals and to use an $S_4$ calculation rather than $S_8$. The input data for this problem follow. The eigenvalue calculated by XSDRNPM is 1.00946.

```
CRITICAL PU(NO3)4 SOLUTION IN A SS SPHERICAL TANK REFLECTED BY WATER
HANSEN-ROACH  MULTIREGION
SOLNPU(NO3)4   1 24.4 .52 1 295 94239 95.4 94240 4.6 END
SS304  2 1.0  END
H2O  3 1  END
END COMP
SPHERICAL  VACUUM  REFLECTED  0  END
1 19.304 ONEEXTERMOD  2 19.426 ONEEXTERMOD  3 39.746 NOEXTERMOD
END ZONE
MORE DATA  ISN=4 SZF=1.5 END
END
```

### C4.5.3 AUTOMATED CROSS-SECTION PREPARATION FOR USE BY SUBSEQUENT CODES

Many of the control modules available through the CSAS family of codes are used primarily to process cross sections for use by other codes. Some of these applications are described in the following subsections.

#### C4.5.3.1 Create a Microscopic AMPX Working Format Cross-Section Library for Use by KENO V.a

This example demonstrates the use of CSASN to create a microscopic, AMPX working format library on unit 4. KENO V.a can be executed as part of the same job and use the created cross sections, or the cross-section data set on unit 4 can be saved for later use. It should be noted that the cross-section library created by CSASN is a problem-dependent cross-section library containing only those nuclides specified in the problem. The resonance corrections applied to these cross sections are appropriate only for the specified problem. A simple demonstration of the use of CSASN to create the library and its use by KENO V.a is given in example 1.

EXAMPLE 1

Consider a problem consisting of four uranium metal cylinders that are 93.2% wt enriched, having a density of 18.76 g/cc. The cylinders are arranged in a 2 × 2 × 2 array. Each has a radius of 5.748 cm and a height of 10.765 cm. The center-to-center spacing in the horizontal plane (X-Y) is 13.74 cm, and the vertical center-to-center spacing is 13.01 cm. Use CSASN to create the microscopic, AMPX working format library for use by KENO V.a. The input data for this problem are:

```
=CSASN
CREATE A MICROSCOPIC AMPX WORKING FMT LIBRARY FOR 2C8
HANSEN-ROACH  INFHOMMEDIUM
URANIUM  1 DEN=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
END
=KENOV
KENO V.A SAMPLE PROBLEM 1  CASE 2C8 BARE
READ PARAMETERS  TME=1.0 FLX=YES FDN=YES FAR=YES LIB=4
END PARAMETERS
READ MIXT MIX=1 92235 4.48006-2 92238 2.6578-3 92234 4.827-4
```

```
92236 9.57-5 END MIXT
READ GEOMETRY
CYLINDER 1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.87 -6.87 6.87 -6.87 6.505 -6.505
END GEOMETRY
READ ARRAY  NUX=2 NUY=2 NUZ=2  END ARRAY
END DATA
END
```

Note that the KENO V.a data given above are the same as those given in Sect. C4.5.2.1 except (1) the parameter data specify LIB=4 to instruct KENO V.a where to obtain the cross-section library and (2) the mixing table data must be supplied to KENO V.a. The number densities in the above data were hand- calculated. An easy method of obtaining the mixing table is to run the CSASN problem with PARM=CHECK, capture the output, and delete everything except the mixing table, which is shown in Fig. C4.5.1. This mixing table can then be edited into the correct format for the KENO V.a mixing table as shown in Figs. C4.5.2 and C4.5.3. Either of these mixing tables can be used in the place of the mixing table shown in example 1.

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY | N.I.T. POINTER |
|---|---|---|---|
| 1 | 1092235 | 4.48072E-02 | 274 |
| 1 | 1092238 | 2.65827E-03 | 276 |
| 1 | 1092234 | 4.82825E-04 | 273 |
| 1 | 1092236 | 9.57448E-05 | 275 |

Figure C4.5.1 Mixing table from CSASN with PARM=CHECK

```
READ MIXT
MIX=1 1092235        4.48072E-02
              1092238  2.65827E-03
              1092234  4.82825E-04
              1092236  9.57448E-05
END MIXT
```

Figure C4.5.2 Mixing table from CSASN edited to KENO V.a format

```
READ MIXT
  MIX=1        1092235  4.48072E-02
  MIX=1        1092238  2.65827E-03
  MIX=1        1092234  4.82825E-04
  MIX=1        1092236  9.57448E-05
END MIXT
```

Figure C4.5.3 Mixing table from CSASN edited to KENO V.a format

### C4.5.3.2 Create a Microscopic Cell-Weighted AMPX Working Format Cross-Section Library

EXAMPLE 1. Use CSAS1X to create a microscopic cell-weighted library.

This example demonstrates the use of CSAS1X to create a microscopic, cell-weighted, AMPX working format, cross-section library for use with KENO V.a. CSAS1X produces the microscopic, cell-weighted AMPX working format library on unit 3. All nuclides that are used in the mixtures specified in the unit cell are flux-weighted according to the characteristics of the unit cell and therefore should not be used in any mixtures utilized in KENO V.a other than the cell-weighted mixture. To utilize this cross-section library in KENO V.a, LIB=3 must be specified in the parameter data and an appropriate mixing table must be supplied. The input data for a CSAS1X problem follow:

```
=CSAS1X
PWR 15x15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH  LATTICECELL
UO2  1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY  2 1.0 605.0 END
H2O  3 0.719 581.0 END
END COMP
SQUAREPITCH  1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

Because the printed mixing tables from CSASN and CSAS1X with PARM=CHECK do not contain mixing table data for the cell-weighted mixture, the number densities can be calculated by hand or another sequence (CSASIX or CSAS2X with PARM=CHECK) can be used to obtain the mixing table.

EXAMPLE 2. Use CSASIX to create a microscopic cell-weighted library.

A microscopic, cell-weighted, AMPX format library is created on unit 3 when CSASIX is run. The number densities of the cell-weighted mixture are volume averaged. To obtain the mixing table, CSASIX can be run with PARM=CHECK. The second mixing table (labeled COMPLETE MIXING TABLE) will contain the data for the cell-weighted mixture, but the mixture number will not be designated as mixture 500. Instead, it will be one larger than the largest mixture specified in the standard compositions data for the problem. In example 2, the cell-weighted mixture will be designated as mixture 4 in the complete mixing table. The CSASIX data are the following:

```
=CSASIX
PWR 15x15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH LATTICECELL
UO2 1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY 2 1.0 605.0 END
H2O 3 0.719 581.0 END
END COMP
SQUAREPITCH 1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

The second mixing table from CSASIX is shown in Fig. C4.5.4. In this problem, mixtures 1, 2, and 3 are the only mixtures defined in the input data, and all of them are used in the unit cell description. The cell-weighted mixture is always one larger than the largest defined mixture number; in this case the cell-weighted mixture is mixture 4. All of the mixtures defined in the standard compositions data are used in the unit cell description. Therefore, all of the nuclides used in this problem are cell-weighted, and it is not valid to use them in any mixture other than the cell-weighted mixture. In Fig. C4.5.5, the cell-weighted mixture is mixture 4. In KENO V.a, LIB=3 must be specified in the parameter data in order to access the cell-weighted cross sections. The cell-weighted mixture is usually defined as mixture 500 in KENO V.a.

The mixing table shown in Fig. C4.5.4 can be edited for use in KENO V.a as shown in Fig. C4.5.5. Because mixtures 1, 2, and 3 were used in the unit cell description, they should not be used in KENO V.a. The mixing table shown in Fig. C4.5.4 can be edited by eliminating the ENTRY column and inserting MIX= in front of each mixture number, and inserting a READ MIXT at the beginning of the data and an END MIXT at the end of the data. Since mixtures 1, 2, and 3 should not be used in KENO V.a, they can be edited out as shown in Fig. C4.5.5, or they can be left in the mixing table and not used in the KENO V.a geometry. It should be noted that the easiest way to use a cell-weighted mixture in KENO V.a is to run CSAS2X and utilize mixture number 500 to designate the cell-weighted mixture in the KENO V.a geometry data so the user does not need to create a mixing table.

COMPLETE MIXING TABLE

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY |
|---|---|---|
| 1 | 1092235 | 7.76007E-04 |
| 4 | 1092235 | 2.57581E-04 |
| 1 | 1092238 | 2.24522E-02 |
| 4 | 1092238 | 7.45257E-03 |
| 1 | 1008016 | 4.64564E-02 |
| 4 | 1008016 | 1.54203E-02 |
| 3 | 3008016 | 2.39971E-02 |
| 4 | 3008016 | 1.34078E-02 |
| 2 | 2040302 | 4.25156E-02 |
| 4 | 2040302 | 4.07230E-03 |
| 3 | 3001001 | 4.79943E-02 |
| 4 | 3001001 | 2.68156E-02 |

Figure C4.5.4 Second mixing table from CSASIX

```
READ MIXT    MIX=500
1092235        2.57581E-04
1092238        7.45257E-03
1008016        1.54203E-02
3008016        1.34078E-02
2040302        4.07230E-03
3001001        2.68156E-02
END MIXT
```

Figure C4.5.5 Mixing table from CSASIX edited to KENO V.a format

EXAMPLE 3. Use CSASIX to create a microscopic cell-weighted library with extra mixtures.

This example is identical to example 2 except water and Zircaloy are to be used in the KENO V.a geometry description in addition to the cell-weighted mixture. The temperature of this water and Zircaloy is 450 K. Note that even if the temperatures of the water and Zircaloy were identical to the temperatures specified for the mixtures that are used in the LATTICECELL description, they would have to be specified again with a different mixture number in order to use them in the KENO V.a geometry data. The input data for this example are the following:

```
=CSASIX
PWR 15X15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH LATTICECELL
UO2 1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY 2 1.0 605.0 END
H2O 3 0.719 581.0 END
ZIRCALLOY 4 1.0 450.0 END
H2O 5 0.847 450.0 END
END COMP
SQUAREPITCH 1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

In this example, the cell-weighted mixture will be identified as mixture 7. The mixing tables from CSASIX are shown in Fig. C4.5.6. In order to use the microscopic, cell-weighted, cross-section library in KENO V.a, LIB=3 must be specified in the KENO V.a parameter data because the CSAS sequences whose names contain an X write the microscopic, cell-weighted, cross-section library on unit 3. The appropriate mixing table for KENO V.a can be created by capturing and editing the COMPLETE MIXING TABLE. The order of scatter should be specified in the KENO V.a mixing table. KENO V.a always defaults to P1 scattering, but CSAS sets the order of scattering based on the cross-section library (presently P1 for the Hansen-Roach 16-group library and P3 for all other libraries). In KENO V.a the order of scattering is set using the parameter SCT= in the mixing table data. The resultant mixing table is shown in Figs. C4.5.7 through C4.5.9 below. In Fig. C4.5.7 all of the entries from the CSASIX COMPLETE MIXING TABLE have been edited to conform to the KENO V.a format. In the CSASIX output, mixture 6 is the cell-weighted mixture. Its mixture number has been changed to 500, but it is not necessary to do so. CSASIX, CSAS2X, and CSAS4X automatically label the cell-weighted mixture cross

section as mixture 500 on the macroscopic, cross-section library. This example arbitrarily redefined mixture 6 to be mixture 500 so a user would recognize that cell-weighted cross sections were used in the geometry. MIXTURES 1, 2, AND 3 SHOULD NOT BE USED IN THE KENO V.a GEOMETRY DATA BECAUSE THEY HAVE BEEN FLUX-WEIGHTED OVER THE CELL TO CREATE A CELL-WEIGHTED CROSS SECTION THAT ALLOWS THE HETEROGENEOUS SYSTEM TO BE REPRESENTED AS A HOMOGENEOUS MATERIAL.

In Fig C4.5.8, mixtures 1, 2, and 3 have been removed because they should not be used in the KENO V.a geometry description.

MIXING TABLE FROM STANDARD COMPOSITIONS DATA

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY | N.I.T. POINTER |
|---|---|---|---|
| 1 | 1092235 | 7.76007E-04 | 274 |
| 1 | 1092238 | 2.24522E-02 | 276 |
| 1 | 1008016 | 4.64564E-02 | 21 |
| 3 | 3008016 | 2.39971E-02 | 21 |
| 5 | 5008016 | 2.39971E-02 | 21 |
| 2 | 2040302 | 4.25156E-02 | 95 |
| 4 | 4040302 | 4.25156E-02 | 95 |
| 3 | 3001001 | 4.79943E-02 | 4 |
| 5 | 5001001 | 4.79943E-02 | 4 |

COMPLETE MIXING TABLE

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY |
|---|---|---|
| 1 | 1092235 | 7.76007E-04 |
| 6 | 1092235 | 2.57581E-04 |
| 1 | 1092238 | 2.24522E-02 |
| 6 | 1092238 | 7.45257E-03 |
| 1 | 1008016 | 4.64564E-02 |
| 6 | 1008016 | 1.54203E-02 |
| 3 | 3008016 | 2.39971E-02 |
| 6 | 3008016 | 1.34078E-02 |
| 5 | 5008016 | 2.82692E-02 |
| 2 | 2040302 | 4.25156E-02 |
| 6 | 2040302 | 4.07230E-03 |
| 4 | 4040302 | 4.25156E-02 |
| 3 | 3001001 | 4.79943E-02 |
| 6 | 3001001 | 2.68156E-02 |
| 5 | 5001001 | 5.65385E-02 |

Figure C4.5.6 Mixing tables from CSASIX

```
READ MIXT       SCT=1
  MIX=1         1092235         7.76007E-04
  MIX=500       1092235         2.57581E-04
  MIX=1         1092238         2.24522E-02
  MIX=500       1092238         7.45257E-03
  MIX=1         1008016         4.64564E-02
  MIX=500       1008016         1.54203E-02
  MIX=3         3008016         2.39971E-02
  MIX=500       3008016         1.34078E-02
  MIX=5         5008016         2.82692E-02
  MIX=2         2040302         4.25156E-02
  MIX=500       2040302         4.07230E-03
  MIX=4         4040302         4.25156E-02
  MIX=3         3001001         4.79943E-02
  MIX=500       3001001         2.68156E-02
  MIX=5         5001001         5.65385E-02
END MIXT
```

Figure C4.5.7 KENO V.a mixing table using all CSASIX entries

```
READ MIXT       SCT=1
  MIX=500       1092235         2.57581E-04
  MIX=500       1092238         7.45257E-03
  MIX=500       1008016         1.54203E-02
  MIX=500       3008016         1.34078E-02
  MIX=5         5008016         2.82692E-02
  MIX=500       2040302         4.07230E-03
  MIX=4         4040302         4.25156E-02
  MIX=500       3001001         2.68156E-02
  MIX=5         5001001         5.65385E-02
END MIXT
```

Figure C4.5.8 KENO V.a mixing table using only mixtures 4, 5 and 6 from CSASIX

```
READ MIXT       SCT=1
 MIX=4          4040302         4.25156E-02
 MIX=5          5008016         2.82692E-02
                5001001         5.65385E-02
 MIX=500        1092235         2.57581E-04
                1092238         7.45257E-03
                1008016         1.54203E-02
                3008016         1.34078E-02
                2040302         4.07230E-03
                3001001         2.68156E-02
END MIXT
```

Figure C4.5.9 Condensed KENO V.a mixing table using only mixtures 4, 5, and 6 from CSASIX

The mixing table shown in Fig. C4.5.9 has been abbreviated by reordering the data to group all of the components of a mixture together and specifying the mixture number only once for each mixture.

### C4.5.3.3 Create a Macroscopic, AMPX Working Format, Cross-Section Library and a Macroscopic, MORSE/KENO, Cross-Section Library

Example 1 illustrates the use of CSASI to produce a macroscopic, AMPX working format, cross-section library on unit 2 for use by KENO V.a. The ID numbers in this library are the mixtures specified in the standard composition data. To utilize this library in KENO V.a, LIB=2 must be entered in the parameter data, and a mixing table must be provided. The mixing table nuclide ID numbers are the mixture numbers used in the CSASI problem. The number density is a multiplication factor applied to the macroscopic cross sections. Entering a number density of 1.0 causes the cross sections to remain unchanged. Consider a CSASI problem that defines three mixtures and mixture 2 is full-density water. If KENO V.a is to use these cross sections and needs to use half-density water in addition to all of the other defined mixtures, mixture 4 can be defined as half-density water, and the mixing table for KENO V.a would be:

```
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 MIX=4 2 0.5 END MIXT
```

LIB=2 would be entered in the parameter data.

EXAMPLE 1

Consider a problem that consists of four 20.96-kg, 93.2% enriched uranium metal cylinders, density 18.76 g/cc, and four 5-l Plexiglas bottles filled with highly enriched uranyl nitrate solution at 415 g/l, a specific gravity of 1.555, and 0.39 mg of excess nitrate per gram of solution. The isotopic content of the uranium metal is 93.2% wt $^{235}U$, 5.6% wt $^{238}U$, 1.0% wt $^{234}U$, and 0.2% wt $^{236}U$. The uranium isotopic content of the nitrate solution is 92.6% wt $^{235}U$, 5.9% wt $^{238}U$, 1.0% wt $^{234}U$, and 0.5% wt $^{236}U$. Each metal unit has a diameter of 11.496 cm, a height of 10.765 cm, and an edge-to-edge spacing of 1.684 cm horizontally and 1.685 cm vertically. The uranyl nitrate solution is contained in a Plexiglas cylinder having an ID of 19.05 cm and an inside height of 17.78 cm. The Plexiglas walls are 0.635 cm thick. The solution units have an

edge-to-edge spacing of 1.43 cm horizontally and vertically. The center-to-center spacing between the metal and solution units is 17.465 cm.

CSASI creates a macroscopic, AMPX working format library on unit 2 and a macroscopic, MORSE/KENO library on unit 14. To use the macroscopic, AMPX working format library in KENO V.a, enter LIB=2 in the parameter data, and enter the following mixing table:

```
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 END MIXT
```

The total input data to run CSASI, followed by KENO V.a are given below:

```
=CSASI
SET UP 4AQUEOUS 4 METAL IN CSAS25
HANSEN-ROACH  INFHOMMEDIUM
URANIUM       1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
SOLNUO2(NO3)2  2 415 9.783-3 SPG=1.555 1.0 293 92235 92.6 92238 5.9
                                    92234 1.0  92236 0.5 END
PLEXIGLAS    3 END
END COMP
END
=KENOV
KENO VA  4 AQUEOUS 4 METAL MIXED UNITS USING CSASI MACRO WORKING LIBR
READ PARAM  FLX=YES FDN=YES  TME=2.0 LIB=2
END PARAM
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 END MIXT
READ GEOM
BOX TYPE 1
CYLINDER  2 1 9.525 8.89 -8.89
CYLINDER  3 1 10.16 9.525 -9.525
CUBOID  0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
BOX TYPE  2
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 6.59 -15.16 6.225 -14.255
BOX TYPE  3
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 15.16 -6.59 6.225 -14.255
BOX TYPE  4
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 6.59 -15.16 14.255 -6.225
BOX TYPE  5
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 15.16 -6.59 14.255 -6.225
END GEOM  READ ARRAY  NUX=2 NUY=2 NUZ=2  LOOP
1 3R2 1 2 1 1 2 1  2 9R1  3 3R1 2 2 1 3R1  4 6R1 2 2 1  5 3R1 2 2 1 2 2 1
END ARRAY
END DATA
END
```

Example 2 illustrates the use of CSASI to produce a macroscopic, MORSE/KENO format, cross-section library on unit 14 for use by KENO V.a. The ID numbers in this library are the mixtures specified in the standard composition data. To utilize this library in KENO V.a, XSC=14 must be entered in the parameter data, and a mixing table is not provided. The mixtures used in the KENO V.a geometry must correspond to the mixtures defined in the CSASI data.

EXAMPLE 2

Consider the problem described in example 1. Produce a macroscopic, MORSE/KENO format, cross-section library on unit 14 for use by KENO V.a. To utilize this library in KENO V.a, XSC=14 must be entered in the parameter data, and a mixing table is not provided. The mixture numbers used in the KENO V.a geometry must correspond to the mixtures defined in the standard composition data of CSASI. The input data required to run this problem are the following:

```
=CSASI
SET UP 4AQUEOUS 4 METAL IN CSAS25
HANSEN-ROACH  INFHOMMEDIUM
URANIUM      1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
SOLNUO2(NO3)2  2 415 9.783-3 SPG=1.555 1.0 293 92235 92.6 92238 5.9
                                    92234 1.0  92236 0.5 END
PLEXIGLAS    3 END
END COMP
END
=KENOV
KENO VA  4 AQUEOUS 4 METAL MIXED UNITS USING CSASI MACRO WORKING LIBR
READ PARAM  FLX=YES FDN=YES  TME=2.0 XSC=14
END PARAM
READ GEOM
BOX TYPE 1
CYLINDER  2 1 9.525 8.89 -8.89
CYLINDER  3 1 10.16 9.525 -9.525
CUBOID  0 1 10.875 -10.875 10.875 -10.875 10.24 -10.24
BOX TYPE  2
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 6.59 -15.16 6.225 -14.255
BOX TYPE  3
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 15.16 -6.59 6.225 -14.255
BOX TYPE  4
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 6.59 -15.16 14.255 -6.225
BOX TYPE  5
CYLINDER  1 1 5.748 5.3825 -5.3825
CUBOID  0 1 6.59 -15.16 15.16 -6.59 14.255 -6.225
END GEOM  READ ARRAY  NUX=2 NUY=2 NUZ=2  LOOP
1 3R2 1 2 1 1 2 1  2 9R1  3 3R1 2 2 1 3R1  4 6R1 2 2 1  5 3R1 2 2 1 2 2 1
```

END ARRAY
END DATA
END

The cross-section libraries created by CSASI are

1. a macroscopic AMPX working format library on unit 2,
2. a macroscopic MORSE/KENO format library on unit 14, and
3. a microscopic AMPX working format library on unit 4.

### C4.5.3.4. Create a Macroscopic, Cell-Weighted, AMPX Working Format Library and a Macroscopic, Cell-Weighted, MORSE/KENO Library

This section illustrates the use of CSASIX to produce cell-weighted, macroscopic libraries for use with a subsequent code. The problem-dependent cross-section libraries created by CSASIX are:

1. A macroscopic, cell-weighted, AMPX working format library on unit 2.
   To use this library in KENO V.a, set LIB=2 in the parameter data and enter a mixing table using the mixture numbers as the nuclide ID's and number densities of 1.0.

2. A microscopic, cell-weighted, AMPX working format library on unit 3.
   To use this library in KENO V.a, set LIB=3 in the parameter data and enter a mixing table using the appropriate nuclide IDs and number densities. If the cell-weighted mixture is to be used, its component nuclides and correct number densities must be included in the mixing table.

3. A microscopic, AMPX working format library on unit 4.
   To use this library in KENO V.a, set LIB=4 in the parameter data and enter the mixing table data. None of the cross sections in this library are cell weighted.

4. A macroscopic cell-weighted MORSE/KENO format library on unit 14.
   To use this library in KENO V.a, set XSC=14 in the parameter data and omit LIB. Do not provide a mixing table. The cell-weighted cross section is mixture 500.

Example 1 shows the use of the macroscopic, cell-weighted, AMPX working format library. Example 2 shows the use of the macroscopic, cell-weighted, MORSE/KENO library.

EXAMPLE 1

This example illustrates the use of CSASIX to create a cell-weighted macroscopic library for use in KENO V.a. The CSASIX data define the materials used to describe a flooded 3 × 3 array of PWR fuel assemblies. The center-to-center spacing between the assemblies is 12 cm. Each fuel assembly is a 17 × 17 array of fuel pins contained in a 0.29-cm-thick boral sheath. The fuel is 3.2% enriched $UO_2$, density = 10.41 g/cc. The fuel OD is 0.819 cm, the gap OD is 0.836 cm, and the clad OD is 9.50 cm. The fuel pin pitch is 1.26 cm. The fuel pin length is 375 cm.

Note that the cell-weighted macroscopic library is specified in KENO V.a by entering LIB=2 in the parameter

data and entering mixing table data. The KENO V.a geometry uses mixture 500 to represent the homogenized pin cells inside each fuel assembly. Unit 1 in the geometry describes the fuel assembly and the spacing cuboid that define the center-to-center spacing between assemblies. Unit 2 defines the water-reflected array of fuel assemblies. Because mixture 3 was used in the unit cell definition in CSASIX, another water, mixture 5, was defined to be used in the KENO V.a geometry because mixtures 1, 2, and 3 have been cell-weighted and cannot be used in the geometry. The input data are the following:

```
=CSASIX
FLOODED PWR FUEL ASSEMBLY (17x17) AT ROOM TEMP
27GROUPNDF4  LATTICECELL
UO2  1  DEN=10.41 1.0 293. 92235 3.2 92238 96.8 END
ZIRCALLOY  2  1.0  END
H2O  3  1.0  END
ARBMTL-B4C 2.64  2  1  1  0  5000  4  6012  1  4  0.015  END
ARBMTL-AL  2.64  1  0  0  0  13027  100.0  4  0.985  END
H2O   5  1.0  END
END COMP
SQUAREPITCH  1.26 0.819 1 3 0.950 2 0.836 0 END
END
=KENOV
3 X 3 ARRAY OF FLOODED FUEL ASSEMBLIES
READ PARAM LIB=2  TME=5.0  END PARAM
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 MIX=4 4 1.0 MIX=5 5 1.0
END MIXT
READ GEOM
UNIT 1
COM='HOMOGENIZED FUEL ASSEMBLY'
CUBOID  500 1 4P10.71 375.0 0.0
CUBOID   4 1 4P11.0  375.29 -0.29
CUBOID   5 1 4P12.0  375.29 -0.29
GLOBAL UNIT 2
COM='WATER REFLECTED 3X3 ARRAY OF FUEL ASSEMBLIES'
ARRAY 1 3*0.0
REPLICATE 5 2 6P3.0 10
END GEOM
READ ARRAY  ARA=1 NUX=3 NUY=3 NUZ=1 FILL F1 END FILL END ARRAY
READ BIAS ID=500 2 11 END BIAS
END
```

EXAMPLE 2

This example is the same as example 1 except the KENO V.a problem will use the macroscopic, MORSE/KENO library. The input data for CSASIX and KENO V.a are given below. Note that XSC=14 is specified in the KENO V.a data, and a mixing table is not entered. However, the mixture numbers used in the geometry correspond to the mixture numbers defined in the standard composition library. Because water, mixture 3, was used in the unit cell definition in CSASIX, another water, mixture 5, has to be defined to be used in the

KENO V.a geometry because mixture 3 has been cell-weighted along with mixtures 1 and 2. The input data follow:

```
=CSASIX
FLOODED PWR FUEL ASSEMBLY (17x17) AT ROOM TEMP
27GROUPNDF4  LATTICECELL
UO2  1  DEN=10.41 1.0 293. 92235 3.2 92238 96.8 END
ZIRCALLOY  2  1.0  END
H2O  3  1.0  END
ARBMTL-B4C  2.64  2  1  1  0  5000  4  6012  1  4  0.015  END
ARBMTL-AL  2.64  1  0  0  0  13027  100.0  4  0.985  END
H2O   5  1.0  END
END COMP
SQUAREPITCH  1.26 0.819 1 3 0.950 2 0.836 0 END
END
=KENOV
3 X 3 ARRAY OF FLOODED FUEL ASSEMBLIES USING MORSE/KENO LIBRARY
READ PARAM XSC=14 TME=5.0  END PARAM
READ GEOM
UNIT 1
COM='HOMOGENIZED FUEL ASSEMBLY'
CUBOID  500 1 4P10.71 375.0 0.0
CUBOID   4 1 4P11.0  375.29 -0.29
CUBOID   5 1 4P12.0  375.29 -0.29
GLOBAL UNIT 2
COM='WATER REFLECTED 3X3 ARRAY OF FUEL ASSEMBLIES'
ARRAY 1 3*0.0
REPLICATE 5 2 6R3.0 10
END GEOM

READ ARRAY  ARA=1 NUX=3 NUY=3 NUZ=1 FILL F1 END FILL END ARRAY
READ BIAS ID=500 2 11 END BIAS
END
```

### C4.5.4 CREATE AND USE TWO CELL-WEIGHTED MIXTURES IN A PROBLEM

Consider a problem in which a stainless steel cylinder with an ID of 56 cm and an inside height of 91 cm is filled with pellets of $UO_2$ in borated water. The steel is 0.125 cm thick. The spherical 2.57%-enriched $UO_2$ pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.13 cm. The spherical 2.96%-enriched $UO_2$ pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.12 cm. The cylindrical tank is filled half full of the 2.96% pellets in borated water, and the remainder is filled with the 2.57%-enriched pellets in borated water.

Example 1 demonstrates how this problem can be calculated by running a CSASIX case for each of the pellet arrays, combining the macroscopic, AMPX working format, cell-weighted libraries using the AMPX module WAX to combine the two libraries. The combined AMPX working format library can then be used to run the KENO V.a problem. It should be noted that this procedure cannot be stacked together to run in a single SCALE execution because the second CSASIX would destroy the library created by the first CSASIX. Example

2 demonstrates how to add an additional WAX step to move the library created by the first CSASIX so that the steps can be stacked together to run in a single SCALE execution.

EXAMPLE 1

STEP 1. Run CSASIX to obtain a macroscopic, cell-weighted, AMPX working format library on unit 2. The mixture 500 on this library represents the 2.57%- enriched pellets in borated water. The input data for this step are given below. The user must save the library that is created on unit 2. The input data are

```
=CSASIX
2.57% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.57 92238 97.43 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
END COMP
SPHTRIANGP 1.13 1.07  1  2 END
END
```

STEP 2. Run CSASIX to obtain a macroscopic, cell-weighted, AMPX working format library on unit 2. The mixture 500 on this library represents the 2.96%-enriched pellets in borated water. The input data for this step are given below. The user must save the library that is created on unit 2. Note that the problem that is to be run by KENO V.a must have stainless steel available to define the cylindrical container. Therefore, stainless steel must be defined in one of the CSASIX problems. The input data are the following:

```
=CSASIX
2.96% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.96 92238 97.04 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
SS304 3 1.0 283 END
END COMP
SPHTRIANGP 1.12   1.07  1  2 END
END
```

STEP 3. Run WAX to combine the macroscopic, cell-weighted, AMPX working format libraries and rename the mixture 500s to distinguish between them. In this case, mixture 500 for the 2.57%-enriched pellets has been renamed to 501 and mixture 500 for the 2.96%-enriched pellets has been renamed to 502. The 2.57% library is mounted on unit 33, the 2.96% library is mounted on unit 34, and the combined library is written out on unit 4. It is not necessary to rename both of the mixture 500s. One of them can be carried through as 500; the other one, as some other number. WAX can be used to rename all of the mixtures on the library. The cell-weighted mixture created by the CSAS modules

is always mixture 500, but WAX can be used to rename it to any desired number. The mixture numbers on the WAX library are the nuclide ID numbers that must be used in the KENO V.a mixing table when multiple cell-weighted mixtures are used.
The user must save the library that is created on unit 4 so that it will be available for use in KENO V.a. The input data follow:

```
=WAX
0$$ 4 34
1$$ 2  1T
2$$ 33 1  2T
3$$ 500
4$$ 501   3T
2$$ 34 2  2T
3$$ 3 500
4$$ 3 502  3T
END
```

STEP 4. Use the combined macroscopic, AMPX working format library to run KENO V.a. Use mixture 501 for the homogenized, cell-weighted, 2.57%-enriched pellets, and use mixture 502 for the homogenized, cell-weighted, 2.96%-enriched pellets. The KENO V.a data are:

```
=KENOV
PROBLEM USING TWO CELL_WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=501 501 1.0  MIX=502 502 1.0 END MIXT
READ GEOM
CYLINDER 502 1 38.0 45.5 0.0
CYLINDER 501 1 38.0 91.0 0.0
CYLINDER  3 1 38.125 91.0 -0.125
END GEOM
END DATA
END
```

Alternatively, KENO V.a could have been set up to define mixtures 1 and 2 as the cell-weighted mixtures as follows:

```
=KENOV
PROBLEM USING TWO CELL_WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=1 501 1.0  MIX=2 502 1.0 END MIXT
READ GEOM
CYLINDER  2 1 38.0 45.5 0.0
CYLINDER  1 1 38.0 91.0 0.0
CYLINDER  3 1 38.125 91.0 -0.125
END GEOM
END DATA
END
```

EXAMPLE 2

This example is the same as example 1 except a WAX step has been added to move the library created by the first CSASIX from unit 2 to unit 12 so that the steps can be stacked together and run in a single SCALE execution. Because all of the steps are stacked together, unit 12 is used instead of unit 33 because unit 12 is defined in the SCALE procedure and will be automatically passed to the next step. The SCALE input data are:

```
=CSASIX
2.57% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.57 92238 97.43 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
END COMP
SPHTRIANGP 1.13 1.07  1  2 END
END
=WAX
0$$ 12 2
1$$ 1   1T
2$$ 2  1  2T
3$$ 500
4$$ 500    3T
END
=CSASIX
2.96% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.96 92238 97.04 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
SS304 3 1.0 283 END
END COMP
SPHTRIANGP 1.12   1.07  1  2 END
END
=WAX
0$$ 4  2
1$$ 2   1T
'RENAME MIXTURE 500 TO MIXTURE 501
2$$ 12 1  2T
3$$ 500
4$$ 501    3T
'RENAME MIXTURE 500 TO MIXTURE 502.  MIXTURE 3 REMAINS MIXTURE 3
2$$ 2  2  2T
3$$ 3 500
4$$ 3 502   3T
END
```

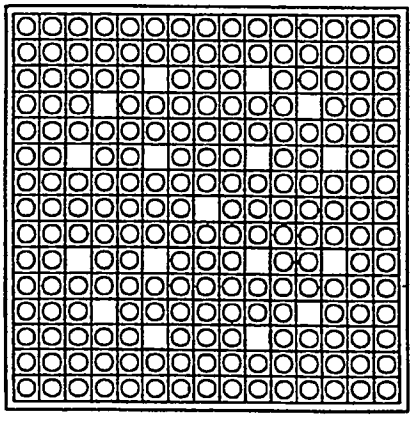
```
=KENOV
PROBLEM USING TWO CELL-WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=501 501 1.0  MIX=502 502 1.0 END MIXT
READ GEOM
CYLINDER 502 1 38.0 45.5 0.0
CYLINDER 501 1 38.0 91.0 0.0
CYLINDER  3 1 38.125 91.0 -0.125
END GEOM
END DATA
END
```

## C4.5.5. CREATE A CELL-WEIGHTED MIXTURE REPRESENTING A NONHOMOGENEOUS FUEL ASSEMBLY

Calculate the k-effective of a 3 × 3 × 3 array of the fuel assembly shown in Fig. C4.5.10. The spacing between the assemblies is 2 cm and is filled with water. The array is reflected by 31 cm of water on all faces. Each assembly is a 15 × 15 array of fuel pins with 16 water holes and a central instrumentation hole. The fuel pins are 1.98%-enriched $UO_2$ at 10.138 g/cc and are 0.94 cm in diameter. The active fuel length is 365.76 cm. The zircaloy-4 clad is 0.545 cm OR × 0.4875 cm IR. The fuel rod pitch is 1.44 cm.

The fuel assembly shrouds are 0.16-cm-thick 304 stainless steel. The IDs are 21.6 cm in cross section. The interstices between the rods are filled with water.

To obtain the correct cell-weighted cross sections for this assembly, run CSASIX to define the cell-weighted fuel pin cell. Then run XSDRNPM to account for the water holes.



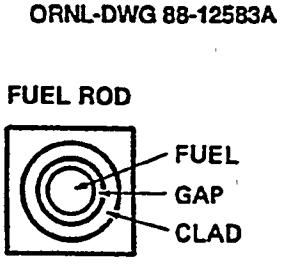



Figure C4.5.10 1.98% enriched $UO_2$ fuel assembly

The input data for CSASIX are given below. Mixtures 1, 2, and 3 are used in the unit cell description, causing them to be cell-weighted and therefore unacceptable for use by a subsequent code. The cross sections for the stainless steel shroud and water holes must be defined so they will be available for use by a subsequent code, XSDRNPM in this case. Mixture 4 is the stainless steel for the fuel assembly shroud, and mixture 5 is the water for the instrumentation hole and water holes. XSDRNPM will create a cell-weighted cross section representing the interior of the fuel assembly. This procedure will use the cell-weighted cross section from CSASIX and the mixture 5 water. Therefore mixture 5 will be cell-weighted and cannot be used in a subsequent code. Since water is to be used in the KENO V.a calculation that uses the cell-weighted cross sections from XSDRNPM, a mixture 6 that is identical to mixtures 3 and 5 must be added to the CSASIX data. The input data follow:

```
=CSASIX
CREATE CELL WEIGHTED CROSS SECTION FOR FUEL PIN CELL
27GROUPNDF4  LATTICECELL
UO2      1 .925   293 92235 1.98 92238 98.02 END
ZIRCALLOY  2 1.0    293 END
H2O      3 1.0    293 END
SS304     4 1.0    293 END
H2O      5 1.0    293 END
H2O      6 1.0    293 END
END COMP
SQUAREPITCH  1.44 .94 1 3 1.09 2 .975 0 END
END
```

The input data for XSDRNPM are given below. Each assembly contains 225 units, 17 of which are water holes (the instrumentation hole is treated as a water hole). The geometry for XSDRNPM is an infinite cylinder of water having an area equal to the area of the central instrumentation hole. The ratio of fuel to water in the assembly is (225-17)/17; so the area of the fuel region surrounding the water is (225-17)/17 times the area of the water region. The OR of the water hole is 0.81243 cm, and the OR of the fuel region is 2.95566 cm. XSDRNPM will use the macroscopic, cell-weighted, cross-section library that was created by CSASIX on unit 2 and will create a new macroscopic cell-weighted cross-section library on unit 3 as specified in the 0$$ array. Note that XSDRNPM will process only those mixtures that are used in the problem. All mixtures that are to be passed through XSDRNPM for use by a subsequent code must be included in every mixture used in XSDRNPM and should have a very small number density (1.0–20). Therefore, mixtures 4 and 6 must be added to both the water (mixture 5 from CSASIX) and the cell-weighted pin cell mixture (mixture 500 from CSASIX) with a negligible number density. The macroscopic, cell-weighted, cross-section library from CSASIX is read from unit 2, and the resultant cell-weighted cross-section library is written on unit 3. The 3$$ array must specify that a weighting calculation is to be done.

```
=XSDRNPM
USE CELL WTD PIN CELL XSECS AND CREATE CELL WTD FUEL ASSMBLY XSECS
0$$ A3 2 E
1$$  2 2 20 1 3 4 6 8 3 1 E
2$$  A7 -1 E
3$$  1 E
4$$  -1 27 E  1T
13$$  1  2  1 2 1 2
```

```
14$$  5 500  4 4 6 6
15** 1.0 1.0  4R1.-20  2T
33## F1.0 4T
35** 9I0 9I0.812433 2.95566
36$$ 10R1 10R2
51$$ 25I1 27 5T
END
```

The input data for KENO V.a are given below. The cell-weighted cross-section library is read from unit 3. This library contains mixtures 5, 500, 4, and 6. Mixtures 5 and 500 have been cell-weighted to represent the heterogeneous fuel assembly. To create a cell-weighted mixture that represents the fuel assembly, the number densities associated with mixtures 5 and 500 are their respective volume fractions in the fuel assembly. In this problem, mixture 1 is the homogenized fuel assembly mixture. Mixture 4 is the stainless steel shroud, and mixture 6 is water. The number density is 1.0 for mixtures 4 and 6.

```
=KENOVA
WATER REFLECTED 3X3X3 ARRAY OF HOMOGENIZED FUEL ASSEMBLIES
READ PARAM LIB=3 TME=10 END PARAM
READ MIXT MIX=1 5 0.075556 500 0.924444  MIX=4 4 1.0 MIX=6 6 1.0 END MIXT
READ GEOM
UNIT 1
COM='CELL-WEIGHTED HOMOGENIZED FUEL ASSEMBLY WITH ZIRCALLOY SHROUD
AND H2O'
CUBOID 1 1 4P10.8 365.76 0.0
REPLICATE 4 1 6R0.16 1
REPLICATE 6 1 6R1.0 1
GLOBAL UNIT 2
ARRAY 1 3*0.0
REPLICATE 6 2 6R3.0 10
END GEOM
READ ARRAY NUX=3 NUY=3 NUZ=3 FILL F1 END FILL END ARRAY
READ BIAS ID=500 2 11 END BIAS
END DATA
END
```

### C4.5.6 SEARCH DATA

Search data must be entered for CSAS4 and CSAS4X. The search data enable the code to perform a search according to the instructions specified by the user. The code begins reading search data when it encounters the words READ SEARCH and continues reading search data until it encounters the words END SEARCH. Search data consist of the search type specification and auxiliary search commands.

#### C4.5.6.1. Search Type Specification

These data are used to define the type of search and to set the parameters that provide limits for the search. The search type specification data consist of (a) a search descriptor, (b) the search type, and (c) optional search parameters as described below.

**C4.5.6.1a** SEARCH DESCRIPTOR is used to define the search mode.
Use OPTIMUM if the maximum value of k-effective is to be determined.
Use CRITICAL if a specified value of k-effective is to be obtained.
Use MINIMUM if the minimum value of k-effective is to be determined.

**C4.5.6.1b** SEARCH TYPE is used to specify the variable that is to be changed during the search procedure.
Use PITCH to alter the center-to-center spacing between the units at the lowest array level. By default only the spacing in the X and Y directions will be altered.
Use DIMENSION to alter the dimensions of one or more geometry regions.
Use CONCENTRATION to alter the concentration of one or more standard compositions in one or more mixtures.

The combination of the search descriptor and the search type defines the search method. An OPTIMUM PITCH search will search for the pitch that gives the maximum value of k-effective. By default only the X and Y spacing will be altered. Auxiliary search commands can be used to instruct the code to also vary the z spacing. An OPTIMUM DIMENSION search will alter the dimensions of one or more geometry regions in accordance with the specified auxiliary search commands and determine the maximum value of k-effective. An OPTIMUM CONCENTRATION search alters the concentration of standard compositions in mixtures in accordance with specified search commands to determine the maximum value of k-effective. A CRITICAL PITCH search alters the spacing between units in the same manner as an optimum pitch search to achieve the specified value of k-effective. A CRITICAL DIMENSION search alters the dimensions of one or more geometry regions in accordance with the specified auxiliary search commands to achieve the specified value of k-effective. A CRITICAL CONCENTRATION search alters the concentration of standard compositions in mixtures in accordance with the specified auxiliary search commands to achieve the specified value of k-effective. A MINIMUM PITCH search will search for the pitch that gives the minimum value of k-effective. A MINIMUM DIMENSION search alters the dimensions of one or more geometry regions in accordance with the specified auxiliary search commands to achieve the minimum value of k-effective. A MINIMUM CONCENTRATION search alters the concentration of standard compositions in mixtures according to the specified auxiliary search commands to determine the miniimum value of k-effective. Only one combination of SEARCH DESCRIPTOR and SEARCH TYPE is allowed in a problem.

**C4.5.6.1c** OPTIONAL SEARCH PARAMETERS are entered after the SEARCH DESCRIPTOR AND SEARCH TYPE and are used to alter the default values of the optional search parameters. Only one set of optional search parameters can be entered for a problem. The optional search parameters are listed below.

PAS=nn is used to set the maximum number of times the search will calculate k-effective. The first pass calculates the k-effective corresponding to the initial geometry dimensions. The second pass calculates the k-effective corresponding to one of the constraints, and the third pass often corresponds to the other constraint. The default value of nn is 10.

NPM=nn is used to set the number of search parameters. The default value is 1 and should not be overridden.

EPS=ff is used to set the search convergence tolerance (the amount by which k-effective is allowed to vary from the desired k-effective). An optimum or minimum search is terminated when the calculated k-effective is within EPS of the optimum or minimum value as indicated by the mathematical fit to

the calculated points. A critical search is terminated when the calculated k-effective is within EPS of the specified k-effective. The default value of ff is 0.005.

KEF=ff is used only for a critical search and is used to specify the desired value of k-effective. The default value of ff is 1.000.

MAXPITCH=ff is allowed ONLY for a PITCH search. It is used to specify the maximum allowed pitch (center-to-center spacing in the x and y directions) between units in an array. The search will terminate if the specified pitch is exceeded. The default value of ff is the pitch corresponding to -5 times the parameter that corresponds to the minimum pitch. It is much easier to specify a maximum allowed pitch than to calculate the appropriate value of the maximum constraint.

MINPITCH=ff is allowed ONLY for a PITCH search. It is used to specify the minimum allowed pitch (center-to-center spacing in the x and y directions) between the units in an array. The search will terminate if the pitch becomes smaller than the specified minimum pitch. The default value of ff is the pitch at which the region immediately inside the spacing region touches the spacing region. It is much easier to specify the minimum allowed pitch than to calculate the appropriate value of the minimum constraint.

MORE is used to terminate the optional search parameters and initiate the auxiliary search commands. Do not enter MORE unless auxiliary search commands are to be entered.

### C4.5.6.2 Auxiliary Search Commands

Auxiliary search commands are entered ONLY if the word MORE was entered in the search type specification data. These data are used to define the method the search will use to alter the geometry data and to set the constraints for the parameter search. The auxiliary search commands consist of (1) INDIVIDUAL SEARCH COMMANDS and (2) SEARCH PARAMETER CONSTRAINTS.

**C4.5.6.2a** An INDIVIDUAL SEARCH COMMAND consists of (1) a command definition, (2) the geometry unit number, (3) the region(s) to be altered, and (4) search constants for each dimension that is to be altered in the specified region(s). A series of individual search commands can be entered for a problem to govern the search process. A new search command is initiated whenever a command definition, item (1) above, is encountered.

The COMMAND DEFINITION tells the code what action is to be taken for a specified geometry region or range of regions. ALTER, CHANGE, or MODIFY are used to cause the specified geometry region(s) to be modified. MAINTAIN is used to maintain the thickness of the specified geometry region(s). KEEP or HOLD are used to cause the specified geometry region(s) to be reset to its original value. Terse input allows the user to truncate words (ALTER, KEEP, MODIFY, MAINTAIN, KEEP, HOLD) as desired. However, sufficient characters must be specified to uniquely define it. For example, an M is not sufficient to distinguish between MODIFY and MAINTAIN. The available command definitions are listed below.

**ALTER**
**CHANGE**
**MODIFY** The list of commands to the left are command definitions, item (1) above. These commands are used to instruct the code to make changes to the specified geometry region(s). The regions that are to be changed are specified using items (2) and (3) above. The amount of change is governed by the search constants, item (4) above.

**MAINTAIN** The list of commands to the left are command definitions, item (1) above. These commands are used to instruct the code to maintain the thickness of the specified region(s) noted in items (2) and (3) above. The search constants, item (4) above, must be nonzero to cause the thickness of the region(s) to be preserved. It should be noted that the thickness of a region can be maintained using this command ONLY when the interior boundary of the region is being altered. If the exterior boundary is being altered, the code is unable to maintain the thickness between it and the boundary interior to it. For example, consider a spherical stainless steel tank with a wall thickness of 1/4 in., filled with a fissile solution. Region 1 is a sphere of solution, and region 2 is the stainless steel. If the search data specifies that region 1 is to be altered and region 2 is to be maintained, the search will maintain a tight-fitting, 1/4-in.-thick, stainless steel region around the fissile solution sphere as it changes sizes. However, if the search data specify only that region 2 is to be altered, the size of the solution sphere will be unchanged and the thickness of the stainless steel spherical shell will vary. If the search data specify that region 1 is to be maintained and region 2 is to be altered, the size of the solution sphere will remain unchanged and the thickness of the stainless-steel container will vary. If the search data specify only that region 2 is to be maintained, no changes will be made and neither region will be changed.

**KEEP**
**HOLD** The list of commands to the left are command definitions, item (1) above. These commands are used to instruct the code to preserve the **original** dimensions of the specified region(s) noted in items (2) and (3) above. The search constants, item (4) above, must be nonzero in order to preserve the original dimensions of the region(s).

THE GEOMETRY UNIT NUMBER, item (2) above, is used to define the unit or box type that contains the geometry region(s) to which the command definition pertains. The geometry unit number is specified by entering the keyword UNIT= followed by the desired unit number.

**UNIT=nn** is used to define the geometry unit to which the previously entered command definition is to be applied. The keyword **UNIT=** is entered, followed by the unit number, nn. There is no default value of nn.

THE REGION(S) TO BE ALTERED, item (3) above, defines the region(s) in unit nn to which the command definition pertains. A single geometry region is specified by entering the keyword REG=, followed by the desired region number, nn. A range of regions is specified by entering the keyword REG=, followed by the first region to which the search commands pertain. The last region in the range is specified by entering the keyword TO and the last region of the range.

**REG=nn** is used to define the first geometry region to which the previously entered command definition is to be applied. The keyword REG= is entered, followed by the region number, nn. There is no default value of nn.

**TO** mm is used to define the final geometry region of the range to which the previously entered command definition applies. The keyword TO is entered, followed by the last region number of the range, mm. There is no default value of mm.

For example, if region 3 of unit 2 is to be altered, items 1, 2, and 3 would be entered as:

ALTER UNIT=2 REG=3

If regions 3, 4, 5, and 6 of unit 2 are to be altered, items 1, 2, and 3 would be entered as:

ALTER UNIT=2 REG=3 TO 6

THE SEARCH CONSTANTS, item (4) above, are proportionality factors that are applied to the specified dimensions of the specified region(s) to which the command definition applies. The search constants define both the dimension to which the search command pertains and the value of the corresponding search constant. Search constants are entered by using a keyword to define the appropriate dimension, followed by a proportionality constant, pc. The default value of pc is 0.0. A nonzero value of pc causes the dimension(s) specified by the keyword to have the search command applied to it. A list of search constant keywords follows.

**ALL=** is used to define the search constant for all of the dimensions of the region or range of regions defined by items (2) and (3) above. ALL= changes the +X, -X, +Y, -Y, +Z, and -Z dimensions of a cuboid; the radius and +Height and -Height of a cylinder; and the radius of a sphere.

**+X=** is used to define the search constant for the positive X dimension of a cuboid.

**–X=** is used to define the search constant for the negative X dimension of a cuboid.

**+Y=** is used to define the search constant for the positive Y dimension of a cuboid.

**–Y=** is used to define the search constant for the negative Y dimension of a cuboid.

**+Z=** is used to define the search constant for the positive Z dimension of a cuboid.

**–Z=** is used to define the search constant for the negative Z dimension of a cuboid.

**R=** is used to define the search constant for the radius of a sphere or a cylinder.

**+H=** is used to define the search constant for the positive height of a cylinder.

**–H=** is used to define the search constant for the negative height of a cylinder.

**CH=** is used to define the search constant for the chord of a hemisphere or hemicylinder.

ENTERING AUXILIARY SEARCH COMMANDS IN THE SEARCH DATA

EXAMPLE 1

Consider an example in which region 2 of unit 1 is a cuboid and all of the dimensions of the cuboid are to be altered. The search data could be entered as:

```
ALTER UNIT=1 REG=2 ALL=1.0
```
or
```
ALTER UNIT=1 REG=2 +X=1.0 -X=1.0 +Y=1.0 -Y=1.0 +Z=1.0 -Z=1.0
```

Because all of the search constants are nonzero, all of the dimensions will be changed. Because the search constants are identical, the original relationship between the dimensions will be preserved as they are altered.

For example, if region 2 of unit 1 is a cylinder and if all the dimensions of the cylinder are to be altered, the search data could be entered as:

```
ALTER UNIT=1 REG=2 ALL=1.0
```
or
```
ALTER UNIT=1 REG=2 R=1.0 +H=1.0 -H=1.0
```

Because all of the search constants are nonzero, all of the dimensions will be changed. Because the search constants are identical, the original relationship between the dimensions will be preserved as they are altered. If the original height-to-diameter ratio is 1.5, that ratio will be preserved throughout the search only if the search constants for the radius and + and - height are identical.

Search constants can be entered sequentially with each new entry overriding only identical previous entries. For example, if region 2 of unit 1 is a cuboid and if all of the dimensions except the -Z dimension are to be altered, the search data could be entered as:

```
ALTER UNIT=1 REG=2 +X=1.0 -X=1.0 +Y=1.0 -Y=1.0 +Z=1.0
```
or
```
ALTER UNIT=1 REG=2 ALL=1.0 -Z=0.0
```

In the second example, all of the search constants corresponding to the cuboid's dimensions are set to 1.0 by using the ALL= command. This includes the -Z dimension. To reset the search constant for the -Z dimension to zero, -Z=0.0 is added following the ALL=1.0 command. If the search commands are reversed,

-Z=0.0 ALL=1.0, the -Z dimension will also be altered because the -Z portion of the ALL= command will override the previously entered -Z=0.0 command.

EXAMPLE 2

A search command must be entered for each unit and region specification. Consider a problem having units 1, 2, and 3. Unit 1 consists of 3 concentric spheres in a cuboid. Unit 2 consists of a single sphere in a cuboid, and unit 3 contains 3 concentric cuboids. A search is to be made that changes the exterior dimensions of the 3 units, the inner sphere of unit 1, and the sphere of unit 2. The thicknesses of the outer spheres of unit 1 are to be maintained, and the two inner cuboids of unit 3 are to remain unchanged. The search data for this problem can be entered as follows:

ALTER UNIT=1 REG=1 ALL=1.0
ALTER UNIT=1 REG=4 ALL=1.0
ALTER UNIT=2 REG=1 TO 2 ALL=1.0
ALTER UNIT=3 REG=3 ALL=1.0
MAINTAIN UNIT=1 REG=2 TO 3 ALL=1.0

**C4.5.6.2b** SEARCH PARAMETER CONSTRAINTS set the parameter limits for the search. The minimum constraint is the minimum value of the parameter allowed in the search. The maximum constraint is the maximum value of the parameter allowed in the search. The initial geometry configuration corresponds to a parameter value of 0.0. A physical limit occurs when the value of the parameter causes geometry intersections. Constraints should be entered for a DIMENSION search.

For a DIMENSION search, the constraints are given by Eqs. (C4.5.1) and (C4.5.2):

$$C_{min} = ((D_{min}/D_i) - 1.0)/SC \qquad \text{(C4.5.1)}$$

$$C_{max} = ((D_{max}/D_i) - 1.0)/SC\,, \qquad \text{(C4.5.2)}$$

where

$C_{min}$ is the minimum constraint for the search
$C_{max}$ is the maximum constraint for the search
$D_{min}$ is the minimum allowed dimension for the search [For a chord, $D_{min} = (Radius_{min} + Chord_{min})/2\ Radius_{min}$]
$D_{max}$ is the maximum allowed dimension for the search [For a chord, $D_{max} = (Radius_{max} + Chord_{max})/2\ Radius_{max}$]
$D_i$ is the initial dimension [For a chord, $D_i = (Radius_{initial} + Chord_{initial})/2\ Radius_{initial}$]
SC is the search constant for that dimension

For example, the initial radius of a sphere is 6 cm, and a search is to be conducted to determine the radius at which the sphere is critical. The minimum radius the user wishes to allow is 3 cm, and the maximum radius to be allowed is 9 cm. A nonzero search constant must be entered to cause the radius to be changed. A search constant of 1.0 will be used for the radius. The constraints calculated from Eqs. (C4.5.1) and (C4.5.2) are:

$C_{min} = ((3.0/6.0)-1.0)/1.0 = -0.5$

$C_{max} = ((9.0/6.0)-1.0)/1.0 = 0.5$

The constraints would be entered in the problem by entering the following data:

-CON=-0.5 +CON=0.5

For a PITCH search, the minimum constraint defines the limit for shrinking the system, and the maximum constraint defines the limit for expanding the system:

$$C_{min} = \underset{i=1}{\overset{i=\text{number of faces}}{MAX}}\left[D_+, D_-\right] \text{ for shrinking}$$

$$C_{max} = \underset{i=1}{\overset{i=\text{number of faces}}{MIN}}\left[D_+, D_-\right] \text{ for expanding}$$

where

$C_{min}$ is the minimum constraint,
$C_{max}$ is the maximum constraint,
$D_+$ are face constraints for the positive dimensions,
$D_-$ are face constraints for the negative dimensions.

Face constraints must be calculated for each face using Eqs. (C4.5.3) and (C4.5.4):

$$D_+ = \frac{(X_{d+} - X_{i+})(SC_+ \times del_+ + SC_- \times del_-)}{(X_{i+} - X_{i-})(SC_+ **2 \times del_+)}, \qquad \text{(C4.5.3)}$$

$$D_- = \frac{-(X_{d-} - X_{i-})(SC_+ \times del_+ + SC_- \times del_-)}{(X_{i+} - X_{i-})(SC_- **2 \times del_-)}, \qquad \text{(C4.5.4)}$$

where

$X_{d+}$ is the desired limit of the positive dimension of the spacing cuboid in that direction (positive dimension of X, Y, or Z, whichever dimension is under consideration)

$X_{i+}$ is the initial positive dimension of the spacing cuboid in that direction (positive dimension of X, Y, or Z, whichever dimension is under consideration)

$X_{d-}$ is the desired limit of the negative dimension of the spacing cuboid in that direction (negative dimension of X, Y, or Z, whichever dimension is under consideration)

$X_{i-}$ is the initial negative dimension of the spacing cuboid in that direction (negative dimension of X, Y, or Z, whichever dimension is under consideration)

$SC_+$ is the search constant for the positive dimension of the spacing cuboid in that direction (positive dimension of X, Y, or Z, whichever dimension is under consideration)

$SC_-$ is the search constant for the negative dimension of the spacing cuboid in that direction (negative dimension of X, Y, or Z, whichever dimension is under consideration)
NOTE: Using a search constant of 1.0 simplifies the determination of $C_{max}$ and $C_{min}$ when the dimensions are to change proportionately.

$del_+$ is the initial distance from the spacing cuboid to the closest interior region in the positive direction (positive dimension of X, Y, or Z, whichever dimension is under consideration)

$del_-$ is the initial distance from the spacing cuboid to the closest interior region in the negative direction (negative dimension of X, Y, or Z, whichever dimension is under consideration)

The search parameter constraints are entered using the following keywords:

–CON=pp is used to set the minimum constraint for the current parameter. The value of pp is defaulted to -10E10 for a dimension search. The value of pp is defaulted to a value that allows geometry regions to touch for a pitch search unless a value was entered for MINPITCH, in which case the parameter corresponding to that pitch is calculated and used.

+CON=rr is used to set the maximum constraint for the current parameter. The value of rr is defaulted to +10E10 for a dimension search and to –5 x pp for a pitch search.

NOTE: A search will reset a constraint (entered using the keyword +CON= or –CON=) that falls outside the default range to the default value. If a PITCH search is specified and if a value has been entered for MAXPITCH and/or MINPITCH, values should not be entered for the constraints. If values are entered for +CON= and/or –CON= for a PITCH search and if MAXPITCH and/or MINPITCH were specified in the optional search parameters, the maximum and minimum constraints will be set to the values corresponding to MAXPITCH and MINPITCH, even though the value of MINPITCH may result in an intersection.

**C4.5.6.2.c** INDIVIDUAL SEARCH COMMANDS for CONCENTRATION SEARCHES

An individual search command for a concentration search consists of (1) a command definition, (2) the mixture number of the mixture to be altered, (3) the name of the standard composition to be altered, and (4) a factor. A series of individual search commands can be entered to govern the search process. A new search command is initiated whenever a command definition, item (1) above, is encountered.

The COMMAND DEFINITION defines the action to be taken for the specified mixture and standard composition component. ALTER, CHANGE, or MODIFY are used to cause the concentration (number densities) of the specified standard composition in the specified mixture to be modified. MAINTAIN, HOLD, and KEEP have not been implemented for a concentration search.

**ALTER**
**CHANGE** The commands to the left are command definitions, item (1) above. These commands instruct the code to modify the specified concentration data.

**MAINTAIN** This command definition is not implemented for a concentration search.

**KEEP**
**HOLD** These command definitions are not implemented for a concentration search.

The MIXTURE NUMBER, item (2) above, defines the mixture that contains the standard composition whose concentration is to be varied during the search. The keyword **MIX=** is entered, followed by the mixture number, nn.

**MIX=nn** is used to define the mixture number associated with the component that is to be changed. The keyword MIX= is entered, followed by the mixture number, nn. There is no default value of nn.

The STANDARD COMPOSITION NAME, item (3) above, defines the standard composition whose concentration will be changed in the defined mixture. Only standard compositions listed in Sect. M8.2 can be entered. Standard compositions beginning with SOLN and ARB cannot be altered directly, but their components can be altered.

**SCNAME=mm** is used to specify the standard composition name of the component that is to be altered. The keyword SCNAME= is entered, followed by the mixture number, mm. There is no default value for mm.

NOTE:
If the standard composition component name specified in the Material Information Data (item 1 of the Standard Composition Specification Data) is a solution, for example, SOLN UO2(NO3)2, and the Concentration Search Data specifies SCNAME=UO2(NO3)2, the amount of UO2(NO3)2 in the solution will be altered, but the amount of water and nitric acid will not be changed. Thus, the resultant mixture may no longer meet the criteria associated with the SOLN specification.

The SEARCH CONSTANT for a concentration search, item (4) above, is a proportional factor that applies to the standard composition being altered. The keyword FACTOR= followed by a proportionality factor is used to specify the search constant for a concentration search.

**FACTOR=pc** is used to specify the search constant. The keyword FACTOR= is entered, followed by the value of the search constant or proportionality factor, pc.

ENTERING AUXILIARY SEARCH COMMANDS IN THE CONCENTRATION SEARCH DATA

EXAMPLE 1

Consider an example in which the density of water is to be varied. Full density water, mixture 1, was specified in the standard composition specification data as: H2O 1 END The auxiliary search data could be entered as follows:

```
ALTER MIX=1 SCNAME=H2O FACTOR=1.0
      or
ALTER MIX=1 SCNA=H2O   FAC=1.0
      or
ALTER M=1   S=H2O      F=1.0
```

Note that terse input allows truncation of the keywords.

EXAMPLE 2

Consider an example in which the density of $UO_2F_2$, is to be varied in mixture 2, a uranyl fluoride solution. The uranyl fluoride solution was specified as:

```
SOLNUO2F2  2  300  0  1  293  92235  5  92238  95  END
```

The auxiliary search data could be entered as follows:

```
ALTER MIX=2  SCNAME=UO2F2  FACTOR=1
      or
ALTER M=2    S=UO2F2       F=1
```

The terse input option allows truncation of the keywords.

**C4.5.6.2.d** SEARCH PARAMETER CONSTRAINTS for CONCENTRATION SEARCHES

The search parameter constraints for concentration searches set the parameter limits for the search. The minimum constraint is the minimum value of the parameter allowed in the search. The maximum constraint is the maximum value of the parameter allowed in the search. The initial concentration corresponds to a parameter value of 0.0. A physical limit occurs when the value of the parameter causes the density of the specified standard composition to become negative. The search can produce an unrealistically high density. Users should manually eliminate those results or set constraints to avoid them.

For a CONCENTRATION search, the constraints are given by Eqs. (C4.5.3) and (C4.5.4). The maximum constraint must be larger than the minimum constraint.

$$C_{min} = \left( \left( \frac{D_{min}}{D_i} \right) - 1 \right) \Big/ \text{FACTOR} \qquad \text{(C4.5.3)}$$

$$C_{max} = \left( \left( \frac{D_{max}}{D_i} \right) - 1 \right) \Big/ \text{FACTOR} \qquad \text{(C4.5.4)}$$

where

| | |
|---|---|
| $C_{min}$ | is the minimum constraint for the search, |
| $C_{max}$ | is the maximum constraint for the search, |
| $D_{min}$ | is the minimum allowed density for the specified standard composition, |
| $D_{max}$ | is the maximum allowed density for the specified standard composition, |
| $D_i$ | is the initial density of the specified standard composition, |
| FACTOR | is the search constant for the standard composition that is being varied. |

Default search constraints are calculated if +CON and -CON are not entered. The default concentration search constraints are calculated as follows:

$$+CON = \begin{Bmatrix} \min(-1/FACTOR), \text{ if any } FACTOR < 0 \\ -S \times (-CON), \text{ if all } FACTOR > 0 \end{Bmatrix}$$

$$-CON = \begin{Bmatrix} -S \times (+CON), \text{ if all } FACTOR < 0 \\ \max(-1/FACTOR), \text{ if any } FACTOR > 0 \end{Bmatrix}$$

### C4.5.7 SEARCH CONSIDERATIONS

Both DIMENSION searches and PITCH searches can be performed using CSAS4 and CSAS4X. A DIMENSION search alters only those regions specified in the search data. A PITCH search alters the center-to-center spacing of units in an array.

By default, a pitch search is performed at the lowest array level, and only the spacing in the x and y directions will be changed. For example, if an array of fuel assemblies is described in the geometry, the lowest array level is the array of fuel pins comprising an assembly. Therefore, an optimum pitch search would alter the spacing between the fuel pins within the assembly. The spacing can be expanded until the array intersects the first region external to it. The exterior size of the fuel pin array would grow or shrink within the confines of the exterior region (fuel assembly shroud). If the external regions are described using replicate regions, the array can grow or shrink within the confines of the maximum and minimum constraints.

If replicate regions are used outside an altered region or an array whose spacing units are altered, the dimensions of the replicate regions are recalculated (maintaining the thickness) at each search pass without having to enter search data for those regions. However, geometry regions specified by a geometry shape (sphere, cuboid, cylinder, etc.) that exist outside an altered region or an array whose spacing units are being altered will remain unchanged unless search data are provided for them. In other words, REPLICATE regions will grow and shrink in response to changes in the dimensions of the interior region, but other geometry shapes will not.

Some of the limitations applicable to a search are:

1. A pitch search is performed only at the lowest array level unless search commands are entered to keep the lowest array level unit unchanged and other commands are entered to cause other units to be altered.

2. A pitch search alters the +X, –X, +Y, and –Y dimensions by default. Appropriate search commands can be entered to alter the +Z and –Z dimensions if it is desirable to do so. Entering a search constant of zero for +X, –X, +Y, and/or –Y will keep the corresponding dimension from being altered.

3. A pitch search alters only the outer region of the unit(s) used in the array at the lowest array level unless the search data specifying otherwise is input.

4. A search cannot alter a region whose boundaries are set by the code (i.e., an ARRAY, CORE BOUNDARY, or REPLICATE following an ARRAY). If the dimensions of a replicate region are to be altered, the dimensions of the region interior to it must be explicitly defined. For example, the interior region can be a standard geometry shape (sphere, cylinder, cuboid, etc.), but cannot be a replicate following an array or core boundary.

5. All searches allow auxiliary search data as described in Sect. C4.4.10.2.

## C4.5.8 OPTIMUM PITCH SEARCH

An optimum pitch search searches for the pitch that yields the highest value of k-effective. An optimum pitch search is activated by entering "OPTIMUM PITCH" in the search data. By default, the search is performed at the lowest array level, and only the spacing in the x and y directions (the x and y dimensions of the outermost region of the unit(s) used in the array at the lowest array level) will be changed. The search constants are defaulted to 1.0 for the +X, –X, +Y, and –Y dimensions of the region. The dimensions of other geometry regions will not be changed (their search constants are defaulted to 0.0) unless additional search data containing appropriate instructions are supplied.

The limits for an optimum pitch search can be set using either the MAXPITCH=/MINPITCH= option in the optional search parameters (Sect. C4.5.6.1c) or the +CON=/–CON= in the search parameter constraints of the auxiliary search commands (Sect. C4.5.6.2b). MAXPITCH=/MINPITCH= are used to enter a value of the maximum allowed pitch and minimum allowed pitch respectively. +CON=/–CON= are used to enter values for the parameter constraints (i.e., the maximum and minimum allowed value of the search parameter). Typically, it is easier to set the maximum allowed pitch and minimum allowed pitch using MAXPITCH=/MINPITCH= than it is to calculate the value of the parameter corresponding to those pitches. The default minimum constraint corresponds to MINPITCH, the pitch at which the largest interior region of a unit used in the array is in contact with the spacing cuboid of that unit. The maximum constraint is defaulted to –5 times the minimum constraint. The code calculates k-effective for the initial geometry first. Then the dimensions corresponding to the minimum constraint and maximum constraint are calculated.

A simple optimum pitch search is demonstrated in example 1:

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a "square" pitch. The uranium spheres are 2 cm in radius, and the center-to-center spacing is 8 cm. The uranium spheres and their associated spacing are defined to be unit 1, and the 10 × 10 × 10 array is defined to be array 1. Search data and results for some optimum pitch searches using this example are given in methods 1 through 6 below.

### DEFAULT OPTIMUM PITCH SEARCH

The search data for a default optimum pitch search are as follows:

READ SEARCH OPTIMUM PITCH END SEARCH

These data will cause the code to alter the +X, –X, +Y, and –Y dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The +Z and –Z dimensions will remain unchanged, thereby preserving the original spacing between the z layers. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y dimensions of the cuboid is preserved throughout the search). Because this is an optimum pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 1

Define unit 1 so the sphere is centered in the unit. The geometry region data and search data for this example are:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  OPTIMUM PITCH  END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
SECOND PASS..k-effective=0.9127 ± 0.0045   Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

THIRD PASS...k-effective=0.2861 ± 0.0025  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P14.0 2P4.0
```

METHOD 2

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case follow:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 4P4.0
READ SEARCH  OPTIMUM PITCH  END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5017 ± 0.0036   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 6.0 –2.0 4P4.0


SECOND PASS..k-effective=0.9127 ± 0.0045   Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0
```

THIRD PASS...k-effective=0.2837 ± 0.0028  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 2P14.0 2P4.0

METHOD 3

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are:

UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 6.0 -2.0 6.0 -2.0
READ SEARCH  OPTIMUM PITCH  END SEARCH

A summary of the code results follows:

FIRST PASS...k-effective=0.5054 ± 0.0036  Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 6.0 -2.0 6.0 -2.0

SECOND PASS..k-effective=0.9094 ± 0.0044  Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

THIRD PASS...k-effective=0.2825 ± 0.0031  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 -2.0 26.0 -2.0 6.0 -2.0

**OPTIMUM PITCH SEARCH IN ALL THREE DIMENSIONS**

Search data for an optimum pitch search that alters the Z dimensions as well as the X and Y dimensions are given below.

READ SEARCH
OPTIMUM PITCH MORE ALTER UNIT=1 REG=2 +Z=1.0 -Z=1.0
END SEARCH

These data will cause the code to alter the +X, –X, +Y, –Y, +Z, and –Z dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The original relationship between the dimensions is preserved (i.e., original ratio of the X to Y to Z dimensions of the cuboid is preserved throughout the search). Because this is an optimum pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 4

Define unit 1 so the sphere is centered in the unit. The geometry region data and search data for this example are:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH
OPTIMUM PITCH MORE ALTER UNIT=1 REG=2 +Z=1.0 –Z=1.0
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033  Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0

SECOND PASS..k-effective=1.2771 ± 0.0048  Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2609 ± 0.0020  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P14.0
```

METHOD 5

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 4P4.0
READ SEARCH
OPTIMUM PITCH MORE ALTER UNIT=1 REG=2 +Z=1.0 -Z=1.0
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5017 ± 0.0036  Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 4P4.0
```

SECOND PASS..k-effective=1.2771 ± 0.0048 Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2570 ± 0.0024 Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 -2.0 4P14.0

METHOD 6

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are the following:

UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 6.0 -2.0 6.0 -2.0
READ SEARCH
OPTIMUM PITCH MORE ALTER UNIT=1 REG=2 +Z=1.0 -Z=1.0
END SEARCH

A summary of the code results follows:

FIRST PASS...k-effective=0.5054 ± 0.0036 Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 -2.0 6.0 -2.0 6.0 -2.0

SECOND PASS..k-effective=1.2771 ± 0.0048 Parameter=-0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2596 ± 0.0021 Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 -2.0 26.0 -2.0 26.0 -2.0

### C4.5.9 OPTIMUM DIMENSION SEARCH

An optimum dimension search searches for the geometry dimensions that yield the highest value of k-effective. An optimum dimension search is activated by entering "OPTIMUM DIMENSION" in the search data. There are no defaulted search data in a dimension search. The user must specify the dimensions to be changed and the manner in which they will be changed as described in the auxiliary search commands (Sect. C4.5.6.2).

A dimension search is performed by altering the regions having nonzero search constants specified in the auxiliary search commands portion of the search data. By default, the search constants for every dimension in the problem is zero. Only those dimensions having a nonzero search constant are altered by the code.

By default, a dimension search sets the minimum constraint to -10E10 and the maximum constraint to +10E10. The relationship between the constraints and the search constants are given in Eqs. (C4.5.1) and (C4.5.2). If the default values of the constraints are used, appropriate search constants must be calculated using these equations. It may be simpler to set the search constants to 1.0 and calculate the corresponding maximum and minimum constraints.

A simple optimum dimension search is demonstrated in example 1.

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a "square" pitch. The uranium spheres are 2 cm in radius, and the center-to-center spacing is 8 cm. The uranium spheres and their associated spacing are defined to be unit 1, and the 10 × 10 × 10 array is defined to be array 1. Search data and results for some optimum dimension searches using this example are given in methods 1 through 6 below.

**OPTIMUM DIMENSION SEARCH IN X AND Y**

The search data for an optimum dimension search that alters only the x and y dimensions of region 2 of unit 1 are given as follows. The constraints are set to be consistent with those of the optimum pitch search above.

```
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

These data will cause the code to alter the +X, –X, +Y, and –Y dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The +Z and –Z dimensions will remain unchanged, thereby preserving the original spacing between the z layers. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y dimensions of the cuboid is preserved throughout the search). Because this is an optimum pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 1

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example are:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE   0 1 2P4.0

SECOND PASS..k-effective=0.9027 ± 0.0043   Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

THIRD PASS...k-effective=0.2861 ± 0.0025   Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P14.0 2P4.0
```

METHOD 2

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are the following:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5017 ± 0.0036   Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS..k-effective=0.9032 ± 0.0042   Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

THIRD PASS...k-effective=0.2837 ± 0.0028   Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 2P14.0 2P4.0
```

METHOD 3

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are the following:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5054 ± 0.0036  Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS...k-effective=0.9009 ± 0.0044  Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

THIRD PASS...k-effective=0.2596 ± 0.0021  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 2P14.0 2P4.0
```

**OPTIMUM DIMENSION SEARCH IN ALL THREE DIMENSIONS**

Search data for an optimum dimension search that alters the X, Y, and Z dimensions follow. The constraints are set to be consistent with the optimum pitch search above.

```
REACH SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

These data will cause the code to alter the +X, –X, +Y, –Y, +Z, and –Z dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y to Z dimensions of the cuboid is preserved throughout the search). Because this is an optimum dimension search, the sphere dimensions will not be altered unless search data are entered for region 1 of unit 1. For this particular example, results for only three passes are given.

METHOD 4

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example are:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033  Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0

SECOND PASS...k-effective=1.2743 ± 0.0053  Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2609 ± 0.0020  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P14.0
```

METHOD 5

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH  OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

FIRST PASS...k-effective=0.5017 ± 0.0036 Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS...k-effective=1.2854 ± 0.0047 Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2570 ± 0.0024 Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 4P14.0

METHOD 6

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are given below:

UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0
READ SEARCH OPTIMUM DIMENSION
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH

A summary of the code results follows:

FIRST PASS...k-effective=0.5054 ± 0.0036 Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS..k-effective=1.2778 ± 0.0045 Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

THIRD PASS...k-effective=0.2596 ± 0.0021 Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 26.0 –2.0 26.0 –2.0

## C4.5.10 CRITICAL PITCH SEARCH

A critical pitch search alters the outer region of the unit or units at the lowest array level in search of a specified value of k-effective. A critical pitch search is activated by entering "CRITICAL PITCH" in the search data. By default, the search is performed at the lowest array level, and only the spacing in the x and y directions ( the x and y dimensions of the outermost region of the unit(s) used in the array at the lowest array level) will be changed. The search constants are defaulted to 1.0 for the +X, -X, +Y, and -Y dimensions of the region. The dimensions of other geometry regions will not be changed (their search constants are defaulted to 0.0) unless additional search data containing appropriate instructions are supplied.

The limits for a critical pitch search can be set using either the MAXPITCH=/MINPITCH= option in the optional search parameters (Sect. C4.5.6.1c) or the +CON=/–CON= in the search parameter constraints of the auxiliary search commands (Sect. C4.5.6.2b). MAXPITCH+/MINPITCH= are used to enter a value of the maximum allowed pitch and minimum allowed pitch respectively. +CON=/–CON= are used to enter values for the parameter constraints (i.e., the maximum and minimum allowed value of the search parameter). Typically, it is easier to set the maximum allowed pitch and minimum allowed pitch using MAXPITCH=/MINPITCH= than it is to calculate the value of the parameter corresponding to those pitches. The default minimum constraint corresponds to MINPITCH, the pitch at which the largest interior region of a unit used in the array is in contact with the spacing cuboid of that unit. The maximum constraint is defaulted to -5 times the minimum constraint. The code calculates k-effective for the initial geometry first. Then the dimensions corresponding to the minimum constraint and maximum constraint are calculated.

A simple critical pitch search is demonstrated in example 1.

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a "square" pitch. The uranium spheres are 2 cm in radius, and the center-to-center spacing is 8 cm. The uranium spheres and their associated spacing are defined to be unit 1, and the 10 × 10 × 10 array is defined to be array 1. Search data and results for some critical pitch searches using this example are given in methods 1 through 6.

### DEFAULT CRITICAL PITCH SEARCH

The search data for a critical pitch search that alters the +X, –X, +Y, and –Y dimensions in search of a k-effective of 0.93 are:

READ SEARCH CRITICAL PITCH KEF=0.93 END SEARCH

These data will cause the code to alter the +X, –X, +Y, and –Y dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The +Z and –Z dimensions will remain unchanged, thereby preserving the original spacing between the z layers. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y dimensions of the cuboid is preserved throughout the search). Because this is a critical pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 1

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example follow:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  CRITICAL PITCH KEF=0.93  END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0

SECOND PASS..k-effective=0.2861 ± 0.0025   Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P14.0 2P4.0

THIRD PASS...k-effective=0.9127 ± 0.0045   Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0
```

After the third pass, the code recognizes that the desired k-effective of 0.93 lies outside the range of the parameter constraints. Messages CS-85 and CS-83 are printed and execution is terminated.

METHOD 2

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH  CRITICAL PITCH KEF=0.93  END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5017 ± 0.0036   Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS..k-effective=0.2837 ± 0.0028   Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 2P14.0 2P4.0
```

THIRD PASS..k-effective=0.9127 ± 0.0045  Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

After the third pass, the code recognizes that the desired k-effective of 0.93 lies outside the range of the parameter constraints. Messages CS-85 and CS-83 are printed and execution is terminated.

METHOD 3

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are:

UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0 READ SEARCH
CRITICAL PITCH KEF=0.93  END SEARCH

A summary of the code results follows:

FIRST PASS...k-effective=0.5054 ± 0.0036  Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS..k-effective=0.2825 ± 0.0031  Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 26.0 –2.0 6.0 –2.0

THIRD PASS...k-effective=0.9094 ± 0.0044  Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 6.0 –2.0

After the third pass, the code recognizes that the desired k-effective of 0.93 lies outside the range of the parameter constraints. Messages CS-85 and CS-83 are printed and execution is terminated.

**CRITICAL PITCH SEARCH IN ALL THREE DIMENSIONS**

Search data for a critical pitch search that alters the Z dimensions as well as the X and Y dimensions follow:

READ SEARCH
CRITICAL PITCH MORE KEF=0.93 ALTER UNIT=1 REG=2 +Z=1.0 –Z=1.0
END SEARCH

These data will cause the code to alter the +X, –X, +Y, –Y, +Z, and –Z dimensions of the outer region of unit 1 and search for the dimensions that give a k-effective of 0.93. The original relationship between the dimensions is preserved (i.e., original ratio of the X to Y to Z dimensions of the cuboid is preserved throughout the search). Because this is an optimum pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 4

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example are given below:

```
UNIT 1
SPHERE 1 1 2.0
CUBE   0 1 2P4.0
READ SEARCH
CRITICAL PITCH MORE KEF=0.93 ALTER UNIT=1 REG=2 +Z=1.0 –Z=1.0
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE   0 1 2P4.0

SECOND PASS..k-effective=0.2609 ± 0.0020   Parameter=2.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P14.0

THIRD PASS...k-effective=1.2771 ± 0.0048   Parameter=–0.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

FOURTH PASS..k-effective=0.7643 ± 0.0047   Parameter=–0.2769
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.8922

FIFTH PASS...k-effective=0.9226 ± 0.0041   Parameter=–0.3643
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.5426
```

METHOD 5

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are given below:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH
CRITICAL PITCH MORE KEF=0.93 ALTER UNIT=1 REG=2 +Z=1.0 –Z=1.0
END SEARCH
```

The code results are summarized below.

```
FIRST PASS...k-effective=0.5017 ± 0.0036   Parameter=0.0000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS..k-effective=0.2570 ± 0.0024   Parameter=2.5000
UNIT 1

SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 4P14.0

THIRD PASS...k-effective=1.2771 ± 0.0048   Parameter=–0.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

FOURTH PASS..k-effective=0.7616 ± 0.0040   Parameter=–0.2762
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.7906 –2.0 4P2.8953

FIFTH PASS...k-effective=0.9221 ± 0.0042  Parameter=–0.3650
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.0801 –2.0 4P2.54
```

METHOD 6

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case follow:

```
UNIT 1
SPHERE 1 1 2.0
```

CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0
CRITICAL PITCH MORE KEF=0.93 ALTER UNIT=1 REG=2 +Z=1.0 –Z=1.0

A summary of the code results follows:

FIRST PASS...k-effective=0.5054 ± 0.0036 Parameter=0.0000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS...k-effective=0.2596 ± 0.0021 Parameter=2.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 26.0 –2.0 26.0 –2.0

THIRD PASS...k-effective=1.2771 ± 0.0048 Parameter=–0.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

FOURTH PASS...k-effective=0.7598 ± 0.0042 Parameter=–0.2751
UNIT 1

SPHERE 1 1 2.0
CUBOID 0 1 3.7992 –2.0 3.7992 –2.0 3.7992 –2.0

FIFTH PASS...k-effective=0.9183 ± 0.0047 Parameter=–0.3652
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.0783 –2.0 3.0783 –2.0 3.0783 –2.0

### C4.5.11 CRITICAL DIMENSION SEARCH

A critical dimension search searches for the geometry dimensions that yield a specified value of k-effective. A critical dimension search is activated by entering "CRITICAL DIMENSION" in the search data. No defaulted search data are in a dimension search. The user must specify the dimensions to be changed and the manner in which they will be changed.

A dimension search is performed by altering the regions having nonzero search constants specified in the individual search commands portion of the search data. By default, the search constants for every dimension in the problem is zero. Only those dimensions having a nonzero search constant are altered by the code.

By default, a dimension search sets the minimum constraint to -10E10 and the maximum constraint to +10E10. The relationship between the constraints and the search constants is given in Eqs. (C4.5.1) and (C4.5.2). If the default values of the constraints are used, appropriate search constants must be calculated using these equations. It may be simpler to set the search constants to 1.0 and calculate the corresponding maximum and minimum constraints.

A simple critical dimension search is demonstrated using example 1 with methods 1 through 6. A more complicated critical dimension search is given in example 2.

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a "square" pitch. The uranium spheres are 2 cm in radius, and the center-to-center spacing is 8 cm. The uranium spheres and their associated spacing are defined to be unit 1, and the 10 × 10 × 10 array is defined to be array 1. Search data and results for some optimum dimension searches using this example are given in methods 1 through 6.

**CRITICAL DIMENSION SEARCH IN X AND Y**

The search data for a critical dimension search that alter only the x and y dimensions of region 2 of unit 1 follow. The constraints are set to be consistent with those of the optimum pitch search above.

```
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

These data will cause the code to alter the +X, –X, +Y, and –Y dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The +Z and –Z dimensions will remain unchanged, thereby preserving the original spacing between the z layers. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y dimensions of the cuboid is preserved throughout the search). Because this is an optimum pitch search, the sphere dimensions will not be changed. For this particular example, results for only three passes are given.

METHOD 1

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example are:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0

SECOND PASS..k-effective=0.2861 ± 0.0025.  Parameter=2.500
UNIT 1
```

SPHERE 1 1 2.0
CUBOID 0 1 4P14.0 2P4.0

THIRD PASS...k-effective=0.9028 ± 0.0043 Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

After the third pass, the code recognizes that the desired k-effective of 0.93 lies outside the range of the parameter constraints. Messages CS-85 and CS-83 are printed and execution is terminated.

METHOD 2

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are the following:

UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH CRITICAL DIMENSION KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH

A summary of the code results follows:

FIRST PASS...k-effective=0.5017 ± 0.0036 Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS..k-effective=0.2837 ± 0.0028 Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 2P14.0 2P4.0

THIRD PASS...k-effective=0.9032 ± 0.0042 Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 2P4.0

After the third pass, the code recognizes that the desired k-effective of 0.93 lies outside the range of the parameter constraints. Messages CS-85 and CS-83 are printed and execution is terminated.

METHOD 3

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case are given below.

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5054 ± 0.0036   Parameter=0.000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS..k-effective=0.2825 ± 0.0031   Parameter=2.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 26.0 –2.0 6.0 –2.0

THIRD PASS...k-effective=0.9009 ± 0.0044   Parameter=–0.500
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 4P2.0 6.0 –2.0
```

**CRITICAL DIMENSION SEARCH IN ALL THREE DIMENSIONS**

Search data for a critical dimension search that alters the X, Y, and Z dimensions follow. The constraints are set to be consistent with the critical pitch search above.

```
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

These data will cause the code to alter the +X, –X, +Y, –Y, +Z, and –Z dimensions of the outer region of unit 1 and search for the dimensions that give the maximum k-effective. The original relationship between the dimensions is preserved (i.e., the original ratio of the X to Y to Z dimensions of the cuboid is preserved throughout the search). Because this is a critical dimension search, the sphere dimensions will not be altered unless search data are entered for region 1 of unit 1. For this particular example results for only three passes are given.

METHOD 4

Define unit 1 so that the sphere is centered in the unit. The geometry region data and search data for this example follow:

```
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.4990 ± 0.0033   Parameter=0.0000
UNIT 1
SPHERE 1 1 2.0
CUBE  0 1 2P4.0

SECOND PASS..k-effective=0.2609 ± 0.0020   Parameter=2.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P14.0

THIRD PASS...k-effective=1.2742 ± 0.0054   Parameter=–0.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

FOURTH PASS..k-effective=0.7687 ± 0.0043   Parameter=–0.2780
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.8881

FIFTH PASS...k-effective=0.9211 ± 0.0039   Parameter=–0.3637
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.5454
```

METHOD 5

Define unit 1 so that the sphere is offset in the unit in the X direction but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case follow:

```
UNIT 1
SPHERE 1 1 2.0
```

```
CUBOID 0 1 6.0 –2.0 4P4.0
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5017 ± 0.0036   Parameter=0.0000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 4P4.0

SECOND PASS..k-effective=0.2569 ± 0.0024   Parameter=2.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 4P14.0

THIRD PASS...k-effective=0.7557 ± 0.0042   Parameter=–0.2733
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.8139 –2.0 4P2.9069

FOURTH PASS..k-effective=0.9142 ± 0.0042   Parameter=–0.3644
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.0851 –2.0 4P2.54

FIFTH PASS...k-effective=0.9228 ± 0.0041   Parameter=–0.3665
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0
```

METHOD 6

Define unit 1 so that the sphere is offset in the unit in X, Y, and Z but the center-to-center spacing is retained at 8.0 cm. The geometry region data for this case follow:

```
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0
READ SEARCH  CRITICAL DIMENSION  KEF=0.93
MORE ALTER UNIT=1 REG=2 +X=1.0 –X=1.0 +Y=1.0 –Y=1.0 +Z=1.0 –Z=1.0
–CON=–0.5 +CON=2.5
END SEARCH
```

A summary of the code results follows:

```
FIRST PASS...k-effective=0.5054 ± 0.0036   Parameter=0.0000
SPHERE 1 1 2.0
CUBOID 0 1 6.0 –2.0 6.0 –2.0 6.0 –2.0

SECOND PASS..k-effective=0.2596 ± 0.0021   Parameter=2.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 26.0 –2.0 26.0 –2.0 26.0 –2.0

THIRD PASS...k-effective=1.2778 ± 0.0045   Parameter=–0.5000
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 6P2.0

FOURTH PASS..k-effective=0.7556 ± 0.0042   Parameter=–0.2748
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.8012 –2.0 3.8012 –2.0 3.8012 –2.0

FIFTH PASS...k-effective=0.9228 ± 0.0041   Parameter=–0.3665
UNIT 1
SPHERE 1 1 2.0
CUBOID 0 1 3.0678 –2.0 3.0678 –2.0 3.0678 –2.0
```

EXAMPLE 2

Consider a problem having units 1, 2, and 3. Unit 1 consists of three concentric spheres in a cuboid. Unit 2 consists of a single sphere in a cuboid, and Unit 3 contains three concentric cuboids. A search is to be made that changes the exterior dimensions of the three units, the inner sphere of unit 1, and the sphere of unit 2. The thicknesses of the outer spheres of unit 1 are to be maintained, and the two inner cuboids of unit 3 are to remain unchanged. The search data for this problem are the following:

```
READ SEARCH CRITICAL DIMENSION
KEF=1.000 EPS=0.005
MORE
ALTER UNIT=1 REG=1 ALL=1.0
ALTER UNIT=1 REG=4 ALL=1.0
ALTER UNIT=2 REG=1 TO 2 ALL=1.0
ALTER UNIT=3 REG=3 ALL=1.0
MAINTAIN UNIT=1 REG=2 TO 3 ALL=1.0
+CON=2.0  –CON=–0.5
END SEARCH
```

The complete input data for this problem are:

```
=CSAS4
SEARCH DEMONSTRATION OF CRITICAL DIMENSION SEARCH
HANSEN-ROACH  INFHOMMEDIUM
```

```
URANIUM  1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
PLEXIGLASS  2 END
SS304   3 END
END COMP
SEARCH DEMONSTRATION OF CRITICAL DIMENSION SEARCH
READ PARAMETERS  TME=1.0 GEN=53
END PARAMETERS
READ GEOMETRY
UNIT 1
SPHERE 1 1 3.0
SPHERE 0 1 3.25
SPHERE 2 1 3.5
CUBE   0 1 2P5.0
UNIT 2
SPHERE 1 1 3.0
CUBE   0 1 2P5.0
UNIT 3
CUBE   0 1 2P4.9
CUBE   3 1 2P5.0
CUBE   0 1 2P5.0
END GEOMETRY
READ ARRAY ARA=1 NUX=3 NUY=3 NUZ=3 FILL
1 3 2 3 2 3 2 3 1
3 2 3 1 3 2 3 1 3
2 3 1 3 1 3 1 3 2
END FILL END ARRAY
END DATA
READ SEARCH   CRITICAL DIMENSION
KEF=1.000  EPS=0.005
MORE
ALTER UNIT=1 REG=1 ALL=1.0
ALTER UNIT=1 REG=4 ALL=1.0
ALTER UNIT=2 REG=1 TO 2 ALL=1.0
ALTER UNIT=3 REG=3 ALL=1.0
MAINTAIN UNIT=1 REG=2 TO 3 ALL=1.0
+CON=2.0  –CON=–0.5
END SEARCH
END
```

The results of this search are

```
FIRST PASS...k-effective=0.4075    Parameter=0.000
UNIT 1
SPHERE 1 1 3.0
SPHERE 0 1 3.25
SPHERE 2 1 3.5
CUBE   0 1 2P5.0
UNIT 2
```

```
SPHERE 1 1 3.0
CUBE  0 1 2P5.0
UNIT 3
CUBE  0 1 2P4.9
CUBE  3 1 2P5.0
CUBE  0 1 2P5.0

SECOND PASS..k-effective=1.0744   Parameter=2.000
UNIT 1
SPHERE 1 1 9.0
SPHERE 0 1 9.25
SPHERE 2 1 9.5
CUBE  0 1 2P15.0
UNIT 2
SPHERE 1 1 9.0
CUBE  0 1 2P15.0
UNIT 3
CUBE  0 1 2P4.9
CUBE  3 1 2P5.0
CUBE  0 1 2P15.0

THIRD PASS...k-effective=1.0159   Parameter=1.777
UNIT 1
SPHERE 1 1 8.3307
SPHERE 0 1 8.5807
SPHERE 2 1 3.8307
CUBE  0 1 2P13.885
UNIT 2
SPHERE 1 1 8.33017
CUBE  0 1 2P13.8856
UNIT 3
CUBE  0 1 2P4.9
CUBE  3 1 2P5.0
CUBE  0 1 2P13.885

FOURTH PASS..k-effective=0.9952   Parameter=1.716
UNIT 1
SPHERE 1 1 8.1490
SPHERE 0 1 8.3990
SPHERE 2 1 8.6490
CUBE  0 1 2P13.583
UNIT 2
SPHERE 1 1 8.1490
CUBE  0 1 2P13.582
UNIT 3
CUBE  0 1 2P4.9
CUBE  3 1 2P5.0
CUBE  0 1 2P13.582
```

Convergence was achieved on the fourth pass. Based on the preceding data, the best estimate of the parameter is 1.730 with the following corresponding geometry:

```
UNIT 1
SPHERE 1 1 8.1901
SPHERE 0 1 8.4401
SPHERE 2 1 8.6901
CUBE  0 1 2P13.650
UNIT 2
SPHERE 1 1 8.1901
CUBE  0 1 2P13.650
UNIT 3
CUBE  0 1 2P4.9
CUBE  3 1 2P5.0
CUBE  0 1 2P13.650
```

In the search data of the above example, note that the search command, ALTER, must be entered each time. If it were entered only once regions 1 and 2 of unit 3 would be altered; no changes would be made to unit 1 or unit 2:

```
ALTER UNIT=1 REG=1 ALL=1.0
   UNIT=1 REG=4 ALL=1.0
   UNIT=2 REG=1 TO 2 ALL=1.0
   UNIT=3 REG=1 TO 2 ALL=1.0
MAINTAIN UNIT=1 REG=2 TO 3 ALL=1.0
```

The following data are equivalent to the above data:

```
ALTER UNIT=3 REG=1 TO 2 ALL=1.0
```

## C4.5.12 CONCENTRATION SEARCH CONSIDERATIONS

Concentration searches can be performed using CSAS4 or CSAS4X. A concentration search alters only those standard compositions specified in the search data. Standard compositions beginning with SOLN and ARB cannot be altered directly. However, one or more of their components can be altered. For example, if the standard composition component name specified in the material information data is SOLN (SOLNUO2F2) and the concentration search data specifies SCNAME=UO2F2, the amount of uranyl fluoride salt, UO2F2, in the solution will be altered, but the amount of water and hydrofluoric acid in the solution will remain unchanged. Therefore, the resultant mixture may no longer meet the criteria for a solution (SOLN) specification.

Physically, the concentration can vary from zero to some upper limit. The code will prevent the concentration from falling below zero but the user is responsible for setting constraints that prevent the concentration from exceeding reasonable values. The theoretical density is a reasonable upper limit.

**WARNING:** Cell-weighted mixtures cannot be modified during search problems.

## C4.5.13 OPTIMUM CONCENTRATION SEARCH

An optimum concentration search alters the concentration of the specified standard composition in the specified mixture to determine the highest value of k-effective. The limits for an optimum concentration search are governed by the values entered for the parameter constraints (see Sect. C4.5.6.2.4).

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a square pitch. The uranium spheres are 2 cm in radius and the center-to-center spacing is 8 cm. The spheres are moderated by ½ density water. Determine the water concentration yielding the maximum value of k-effective. The maximum allowed density is full density water. The minimum allowed density is 0.05 density water.

The input data for this problem specifies mixture 2 to be H2O with a density multiplier of 0.5 (H2O 2 0.5 END). The maximum constraint is calculated according to:

$$+\text{CON} = \left( \frac{D_{max}}{D_{initial}} - 1 \right) / \text{FACTOR} \quad +\text{CON} = \left( \frac{1.0}{0.5} - 1 \right) / 1 = 1.0.$$

The minimum constraint is calculated according to:

$$-\text{CON} = \left( \frac{D_{min}}{D_{initial}} - 1 \right) / \text{FACTOR} \quad -\text{CON} = \left( \frac{0.05}{0.5} - 1 \right) / 1 = -0.9.$$

The search data with constraints to limit the water density range from 5% density to full density follows.

```
READ SEARCH  OPTIMUM CONCENTRATION  MORE
ALTER  MIX=2  SCNAME=H2O  FACTOR=1
+CON=1.0  –CON=–0.9  END SEARCH
```

The code results follow:

FIRST PASS...k-effective = 1.22307±0.0049  Parameter=0.000

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.33757-2 |
| | 2 | 2008016 | 1.66879-2 |

SECOND PASS...k-effective=0.576665±0.0039  Parameter= –0.900

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.33757-3 |
| | 2 | 2008016 | 1.66879-3 |

THIRD PASS...k-effective=1.16362±0.0044 Parameter=1.000

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 6.67514-2 |
| | 2 | 2008016 | 3.33757-2 |

FOURTH PASS...k-effective=1.16362±0.0044 Parameter=0.500

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 5.00636-2 |
| | 2 | 2008016 | 2.50318-2 |

FIFTH PASS...k-effective=1.23497±0.0048 Parameter=0.080

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.60458-2 |
| | 2 | 2008016 | 1.80229-2 |

For this particular problem, convergence was achieved on search pass number five. Since the initial pass was set up for half density water and the converged results occurred for 0.54 density water, it is not necessary to resubmit the problem using the converged water concentration.

## C4.5.14 CRITICAL CONCENTRATION SEARCH

A critical concentration search alters the concentration of the specified standard composition in the specified mixture to obtain a specified value of k-effective. The limits for a critical concentration search are governed by the values entered for the parameter constraints (see Sect. C4.5.6.2.4).

EXAMPLE 1

Consider a 10 × 10 × 10 array of uranium spheres arranged in an array having a square pitch. The uranium spheres are 2 cm in radius and the center-to-center spacing is 8 cm. The spheres are moderated by ½ density water. Determine the water concentration yielding a k-effective of 0.90. The maximum allowed density is full density water. The minimum allowed density is 0.05% density water.

The input data for this problem specifies mixture 2 to be H2O with a density multiplier of 0.5 (H2O 2 0.5 END). The maximum constraint is calculated according to

$$+\text{CON} = \left( \frac{D_{max}}{D_{initial}} - 1 \right) / \text{FACTOR} \quad +\text{CON} = \left( \frac{1.0}{0.5} - 1 \right) / 1 = 1.0.$$

The minimum constraint is calculated according to:

$$-\text{CON} = \left( \frac{D_{min}}{D_{initial}} - 1 \right) / \text{FACTOR} \quad -\text{CON} = \left( \frac{0.0005}{0.5} - 1 \right) /1 = -0.999.$$

The search data with constraints to limit the water density range from 0.05% density to full density follows.

```
READ SEARCH  CRITICAL CONCENTRATION  KEF=0.90  MORE
ALTER  MIX=2  SCNAME=H2O  FACTOR=1
+CON=1.0  -CON=-0.999  END SEARCH
```

The code results follow:

FIRST PASS...k-effective=1.2344±0.0032  Parameter=0.000

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.33757-2 |
| | 2 | 2008016 | 1.66872-2 |

SECOND PASS...k-effective=0.5063±0.0024  Parameter=-0.999

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.33749-5 |
| | 2 | 2008016 | 1.66874-5 |

THIRD PASS...k-effective=1.0628±0.0037  Parameter=-0.459

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 1.80613-2 |
| | 2 | 2008016 | 9.03063-3 |

FOURTH PASS...k-effective=0.5804±0.0027  Parameter=-0.894

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 3.52600-3 |
| | 2 | 2008016 | 1.76300-3 |

FIFTH PASS...k-effective=0.9255±0.0036  Parameter=-0.601

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 1.33055-2 |
| | 2 | 2008016 | 6.65275-3 |

SIXTH PASS...k-effective=0.8988±0.0035  Parameter=-0.625

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 2 | 2001001 | 1.25267-2 |
| | 2 | 2008016 | 6.26336-3 |

Convergence was achieved on the sixth pass. Based on the data, the best estimate of the parameter is -0.624 and the corresponding mixing table for water is:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 2 | 2001001 | 1.25630-2 |
| 2 | 2008016 | 6.28149-3 |

The averaged results and the best estimate correspond to a converged density multiplier of 0.188 for the parameter = -0.625. Since the code does resonance processing through BONAMI and NITAWL prior to each run, this is a good result.

EXAMPLE 2

Consider the 10x10x10 array of uranium spheres described in Example 1. The spheres are moderated by water having a density of 0.25 g/cc. The search limits are specified to allow the minimum water density to be 0.15 g/cc and the maximum water density to be 0.33 g/cc. Determine the water density corresponding to a k-effective of 0.90.

The input data for this problem specifies mixture 1 as water with a density of 0.25 g/cc (H2O 1 DEN=0.25 END). The search data is:

```
READ SEARCH CRIT CONC KEF=0.90 MORE ALTER MIX=1
SC=H2O FACTOR=1 +CON=0.32 -CON=-0.40 END SEARCH
```

The code results follow:

FIRST PASS...k-effective=1.02415±0.0036 Parameter=0.000

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1001001 | 1.67179-2 |
| | 1 | 1008016 | 8.35897-3 |

SECOND PASS...k-effective=-.8072±0.0034 Parameter=-0.400

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1001001 | 1.00308-2 |
| | 1 | 1008016 | 5.01538-3 |

THIRD PASS...k-effective=0.9175±0.0037 Parameter=-0.229

| mixing table entries for water: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1001001 | 1.28920-2 |
| | 1 | 1008016 | 6.44598-3 |

FOURTH PASS...k-effective=-.8956±0.0036 Parameter=-0.256

mixing table entries for water:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1001001 | 1.24389-2 |
| 1 | 1008016 | 6.21943-3 |

Convergence was achieved on the fourth pass. Based on the data, the best estimate of the parameter is -0.250 and the corresponding mixing table for water is:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1001001 | 1.25303-2 |
| 1 | 1008016 | 6.26514-3 |

This parameter corresponds to a water density of 0.1875 g/cc (H2O 1 DEN=0.1875 END).

EXAMPLE 3

Consider a spherical tank of 304 stainless steel filled with 5% enriched uranyl fluoride with a density of 300 gU/ℓ and no excess acid. The inside radius of the tank is 30 cm and the wall is 0.25 cm thick. Determine the concentration of uranyl fluoride yielding a k-effective of 1.0

The input data for this problem specifies mixture 1 to be a uranyl fluoride solution as shown below.

```
SOLNUO2F2 1 300 0 1 293 92235 5 92238 95 END
```

The search data, including constraints to limit the density from 75 gU/ℓ to 750 gU/ℓ follows:

```
READ SEARCH CRITICAL CONCEN KEF=1 MORE
ALTER MIX=1 SCNA=UO2F2 FAC=1 -CON=-0.75 +CON=1.5
END SEARCH
```

The code results for the first search attempt follow:

FIRST PASS...k-effective=-.8009±0.0030 Parameter=0.000

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 3.84319-5 |
| 1 | 1092238 | 7.20984-4 |
| 1 | 1008016 | 3.33408-2 |
| 1 | 1009019 | 1.51883-3 |
| 1 | 1001001 | 6.36440-2 |

SECOND PASS...k-effective=1.0306±0.0039 Parameter=1.500

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 9.60799-5 |
| 1 | 1092238 | 1.80246-3 |
| 1 | 1008016 | 3.56191-2 |
| 1 | 1009019 | 3.79708-3 |
| 1 | 1001001 | 6.36440-2 |

THIRD PASS...k-effective=1.0230±0.0034 Parameter=1.300

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 8.84014-5 |
| 1 | 1092238 | 1.65841-3 |
| 1 | 1008016 | 3.53156-2 |
| 1 | 1009019 | 3.49363-3 |
| 1 | 1001001 | 6.36440-2 |

FOURTH PASS...k-effective=0.9536±0.0038 Parameter=0.695

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 6.51516-5 |
| 1 | 1092238 | 1.22225-3 |
| 1 | 1008016 | 3.43968-2 |
| 1 | 1009019 | 2.57479-3 |
| 1 | 1001001 | 6.36440-2 |

FIFTH PASS...k-effective=1.0019±0.0041 Parameter=1.028

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 7.79365-5 |
| 1 | 1092238 | 1.46209-3 |
| 1 | 1008016 | 3.49020-2 |
| 1 | 1009019 | 3.08005-3 |
| 1 | 1001001 | 6.36440-2 |

Convergence was achieved on the fifth pass. Based on the preceding data, the best estimate of the parameter is 1.0108. The corresponding mixing table for mixture 1 follows.

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 7.72789-5 |
| 1 | 1092238 | 1.44975-3 |
| 1 | 1008016 | 3.48761-2 |
| 1 | 1009019 | 3.05406-3 |
| 1 | 1001001 | 6.36440-2 |

The best estimate of the parameter, 1.0108, corresponds to 603 gU/ℓ. Because the current implementation of the concentration search varies only the UO2F2 portion of the solution and does not alter the acid and water, the

converted results do not accurately represent the true solution. To test the best-estimate results, the problem was rerun with the uranyl fluoride solution specified as

```
SOLNUO2F2 1 603 0 1 293 92235 5 92238 95 END
```

The search data were specified to limit the range of the search to vary from 500 gU/ℓ to 700 gU/ℓ as shown below.

```
READ SEARCH CRITICAL CONCEN KEF=1 MORE
ALTER MIX=1 SCNA=UO2F2 FAC=1 -CON=-0.17 +CON=0.16 END SEARCH
```

The code results for the second search attempt follow.

FIRST PASS...k-effective=1.0111±0.0035 Parameter=0.000

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 7.72482-5 |
| | 1 | 1092238 | 1.44918-3 |
| | 1 | 1008016 | 3.31955-2 |
| | 1 | 1009019 | 3.05285-3 |
| | 1 | 1001001 | 6.02853-2 |

SECOND PASS...k-effective=0.96817±0.0034 Parameter=-0.17

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 6.41160-5 |
| | 1 | 1092238 | 1.20282-3 |
| | 1 | 1008016 | 3.26765-2 |
| | 1 | 1009019 | 2.53387-3 |
| | 1 | 1001001 | 6.02853-2 |

+THIRD PASS...k-effective=1.0017±0.0040 Parameter=-0.044

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 7.38460-5 |
| | 1 | 1092238 | 1.38535-3 |
| | 1 | 1008016 | 3.30611-2 |
| | 1 | 1009019 | 2.98140-3 |
| | 1 | 1001001 | 6.02853-2 |

Convergence was achieved on the third pass, and the best estimate of the parameter was nearly identical to the parameter from the third pass. A parameter of -0.044 corresponds to a density of 576 gU/ℓ. Respecify the problem to run using this density and limiting the density range from 500 gU/ℓ (–CON=-0.132) to 625 gU/ℓ (+CON=0.085).

The code results for the third search attempt follow.

FIRST PASS...k-effective=0.9879±0.0041 Parameter=0.000

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 7.3789-5 |
| | 1 | 1092238 | 1.38429-3 |
| | 1 | 1008016 | 3.32085-2 |
| | 1 | 1009019 | 2.91616-3 |
| | 1 | 1001001 | 6.05846-2 |

SECOND PASS...convergence achieved, k-effective=1.0011±0.0039 Parameter=0.085

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 8.00614-5 |
| | 1 | 1092238 | 1.50195-3 |
| | 1 | 1008016 | 3.34563-2 |
| | 1 | 1009019 | 3.16403-3 |
| | 1 | 1001001 | 6.05846-2 |

A parameter of 0.085 corresponds to a density of 625 gU/ℓ. Respecify the problem to run using a density of 625 gU/ℓ, and limiting the range from 550 gU/ℓ (-CON=-0.12) to 700 gU/ℓ (+CON=0.12).

The code results for the fourth search attempt follow.

FIRST PASS...k-effective=1.0168±0.0043 Parameter=0.000

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 8.00666-5 |
| | 1 | 1092238 | 1.50205-3 |
| | 1 | 1008016 | 3.31850-2 |
| | 1 | 1009019 | 3.16423-3 |
| | 1 | 1001001 | 6.00415-2 |

SECOND PASS...k-effective=0.9818±0.0040 Parameter=-0.12

| mixing table entries for mixture 1: | MIXTURE | NUCLIDE | DENSITY |
|---|---|---|---|
| | 1 | 1092235 | 7.04586-5 |
| | 1 | 1092238 | 1.32180-3 |
| | 1 | 1008016 | 3.28053-2 |
| | 1 | 1009019 | 2.78452-3 |
| | 1 | 1001001 | 6.00415-2 |

THIRD PASS...convergence achieved, k-effective=1.0032±0.0037 Parameter=-0.0575

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 7.54619-5 |
| 1 | 1092238 | 1.41567-3 |
| 1 | 1008016 | 3.30030-2 |
| 1 | 1009019 | 2.98226-3 |
| 1 | 1001001 | 6.00415-2 |

A parameter of -0.0575 corresponds to a density of 589 gU/ℓ.

Respecify the problem to use a density of 589 gU/ℓ and limiting the search range from 550 gU/ℓ (–CON=-0.066) to 625 gU/ℓ (+CON=0.061), the following results were obtained for the fifth search pass.

FIRST PASS...convergence achieved k-effective=1.0078±0.0039 Parameter=0.000

mixing table entries for mixture 1:

| MIXTURE | NUCLIDE | DENSITY |
|---|---|---|
| 1 | 1092235 | 7.54547-5 |
| 1 | 1092238 | 1.41553-3 |
| 1 | 1008016 | 3.32022-2 |
| 1 | 1009019 | 2.98197-3 |
| 1 | 1001001 | 6.04405-2 |

This type of concentration search did not function as well as desired because only the UO2F2 portion of the solution was altered. Therefore, the H/X for a converged search was not the true H/X for a solution having the density determined by the search. A summary of the search results for example 3 follows.

EXAMPLE 3 SEARCH RESULTS

| Search number | Initial density (gU/ℓ) | Initial H/U-235 | Converged parameter | Converged density (gU/ℓ) | Converged H/U-235 |
|---|---|---|---|---|---|
| 1 | 300 | 1,656 | 1.0108 | 603 | 824 |
| 2 | 603 | 780 | -0.044 | 576 | 818 |
| 3 | 576 | 821 | 0.085 | 625 | 757 |
| 4 | 625 | 750 | -0.0575 | 589 | 796 |
| 5 | 589 | 801 | 0.000 | 589 | 801 |

## C4.6 DESCRIPTION OF OUTPUT

This section describes the output from the CSAS control sequences. The output files that are created when the CSAS control sequences are executed consist of (1) output from the control module, (2) output from the functional modules invoked by the control module, and (3) a listing of the input data. The order in which these output files appear is system dependent. If PARM=CHECK is specified on the analytical sequence indicator, there will be no output from the functional modules. The output files from the functional mudules are described in the section pertaining to the module (see Table C4.6.1).

Table C4.6.1 Location of functional module output

| Functional module | Section |
|---|---|
| BONAMI | F1.6 |
| NITAWL-II | F2.8.2 |
| XSDRN | F3.7 |
| ICE | F8.5 |
| KENO V.a | F11.6 |

### C4.6.1 CONTROL MODULE HEADER PAGE

The control module header page, shown in Fig. C4.6.1, prints the name of the control module in block letters. The job name from the job control language, the date (month/day.year), and the time at execution (hour:minute:second) are also printed. This time is given in terms of a 24-h clock, with midnight being 2400 h.

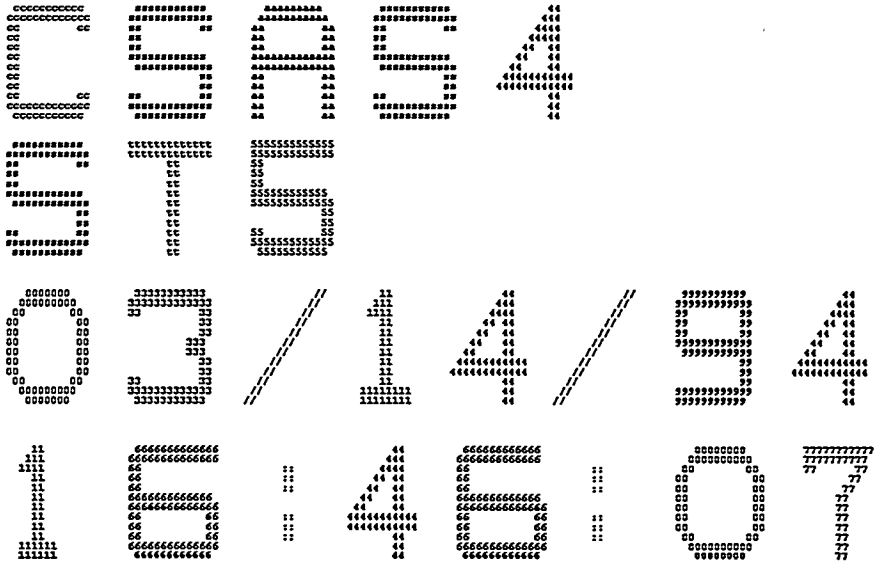


Figure C4.6.1 Sample control module header page

## C4.6.2 CONTROL MODULE PROGRAM VERIFICATION TABLE

The control module program verification information table, shown in Fig. C4.6.2, is printed after the header page. It lists the name of the control module that is being executed, the date the load module was created, the library that contains the load module, the computer code name from the configuration control table, and the revision number. Some computers do not store load modules in a library, in which case the library entry is left blank. The job name, date, and time of execution are also printed. This information may be useful for quality assurance purposes.

```
 ssssssssss     cccccccccc      aaaaaaaaa     ll             eeeeeeeeeeee
ssssssssssss   cccccccccccc    aaaaaaaaaa     ll             eeeeeeeeeeee
ss         ss  cc          cc  aa         aa  ll             ee
ss             cc              aa         aa  ll             ee
ss             cc              aa         aa  ll             ee
ssssssssssss   cc              aaaaaaaaaaaaa  ll             eeeeeeeee
 ssssssssssss  cc              aaaaaaaaaaaaa  ll             eeeeeeeee
          ss   cc              aa         aa  ll             ee
          ss   cc              aa         aa  ll             ee
ss        ss   cc          cc  aa         aa  ll             ee
ssssssssssss   cccccccccccc    aa         aa  llllllllllll   eeeeeeeeeeee
 ssssssssss     cccccccccc     aa         aa  llllllllllll   eeeeeeeeeeee

********************************************************************************
********************************************************************************
********************************************************************************
*****                                                                      *****
*****                  program verification information                 *****
*****                                                                      *****
*****                 code system:    scale  version:    4.2            *****
*****                                                                      *****
********************************************************************************
********************************************************************************
*****                                                                      *****
*****                                                                      *****
*****              program:  csas                                          *****
*****                                                                      *****
*****        creation date:  12/01/93                                      *****
*****                                                                      *****
*****              library:  /scale4.2p/exe                                *****
*****                                                                      *****
*****                                                                      *****
*****      production code:  csas                                          *****
*****                                                                      *****
*****              version:  2.2                                           *****
*****                                                                      *****
*****              jobname:  st5                                           *****
*****                                                                      *****
*****    date of execution:  03/14/94                                      *****
*****                                                                      *****
*****    time of execution:  16:46:07                                      *****
*****                                                                      *****
********************************************************************************
********************************************************************************
********************************************************************************
```

Figure C4.6.2 Sample control module program verification table

## C4.6.3 MATERIAL INFORMATION PROCESSOR OUTPUT

The Material Information Processor output is printed immediately after the program verification table. This output is discussed in detail in Sects. M7.6.1 through M7.6.6. An example of this output is given in Figs. C4.6.3 and C4.6.4.

```
sample problem 4  set up 2c8 in csas25

**** problem parameters ****

lib hansen-roach library
mxx            1 mixtures
msc            1 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           0 fuel solutions

**** problem composition description ****

sc  uranium      standard composition
mx             1 mixture no.
vf         .9850 volume fraction
roth     19.0500 theoretical density
temp       293.0 deg kelvin
           92235  93.20%
           92238   5.60%
           92234   1.00%
           92236    .20%
end

**** problem geometry ****

**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

Figure C4.6.3 Part 1 of Material Information Processor printout

```
*** sample problem 4  set up 2c8 in csas25 ***

*** ********** data library information ********** ***

***  unit number   data set name                                volume name   unit function                ***
***  ------------  ------------                                ------------  -------------                ***

***  89  /scale4.2p/datalib/scale.rev05.sclib                                standard composition library ***
***  81  /scale4.2p/datalib/scale.rev02.xn16                                  cross section library        ***
***  11  ft11f001                                                            short cross section library  ***

*** standard composition library data ***

***  unit number  : 89
***  dataset name : /scale4.2p/datalib/scale.rev05.sclib
***  library title: scale-4 standard composition library
***                 386 standard compositions, 332 nuclides
***                 10 elements with variable isotopic distrbutions.
***  creation date:  1/25/94

*** cross section library data ***

***  unit number  : 81
***  dataset name : /scale4.2p/datalib/scale.rev02.xn16
***  library title: scale 4 - 16 neutron group criticality safety library
***                   hansen-roach data with knight modifications and some endf/b 4 data
***                     compiled for nrc       1/27/89
***                       last updated       10/12/89
***                         l.m.petrie       ornl
```

Figure C4.6.4 Part 2 of Material Information Processor printout

## C4.6.4 KENO V.a OUTPUT

Output from KENO V.a is printed if the control sequence expects to execute KENO V.a and PARM=CHECK is specified on the analytical sequence indicator. See Sects. F11.6.3 through F11.6.20 for an explanation of the KENO V.a output.

If the control sequence expects to execute KENO V.a and if PARM=CHECK is not specified on the analytical sequence indicator, the KENO V.a output consists of a set of cryptic statements indicating the number of I/Os used to process the KENO V.a data as shown in Fig. C4.6.5.

```
 ........    0 io's were used before reading keno v data    ........
 ........    0 io's were used reading the keno v parameter data    ........
*************** data reading completed ***************
 ........    0 io's were used preparing the keno v input data    ........
 ........    0 io's were used loading the keno v data    ........
 ........    0 io's were used loading the data    ........
 ........    0 io's were used checking the keno v geometry data    ........
 *****  restart data has been written on unit 95  *****
 ........    0 io's were used writing the keno v - csas data    ........
 ........    0 io's were used before reading search data    ........
 ........    0 io's were used processing csas input data    ........
```

Figure C4.6.5 Summary of KENO V.a I/O usage by CSAS control module

The first line of Fig. C4.6.5, the number of I/Os used before reading KENO V.a data, is always printed for the analytical sequences contained in the CSAS family of control modules. It is the only line of Fig. C4.6.5 that is printed for CSASN, CSAS1X, CSASI, and CSASIX. The next to the last line of Fig. C4.6.5 is printed only for control modules that expect to perform a search by executing the control module MODIFY (i.e., CSAS4 and CSAS4X). All other lines of Fig. C4.6.5 are printed by control modules that expect to execute KENO V.a (i.e., CSAS25, CSAS2X, CSAS4, and CSAS4X).

## C4.6.5 PROGRAM VERIFICATION TABLE FOR THE CONTROL MODULE MODIFY

If a search is to be performed, the control module MODIFY is invoked to alter the KENO V.a geometry data and execute KENO V.a using the new data. This procedure is repeated until the problem achieves convergence, runs out of time or I/Os, or completes the specified number of search passes.

The program verification information table for MODIFY is shown in Fig. C4.6.6. It lists the name of the control module, the date the load module was created, the library that contains the load module, the computer code name from the configuration control table, and the revision number. Some computers do not store load modules in a library, in which case the library entry is left blank. The job name, date, and time of execution are also printed. This information may be useful for quality assurance purposes.

```
 ssssssssss     ccccccccccc     aaaaaaaaa     ll             eeeeeeeeeeee
ssssssssssss   ccccccccccccc    aaaaaaaaaaa    ll             eeeeeeeeeeee
ss          ss cc          cc  aa          aa  ll             ee
ss             cc              aa          aa  ll             ee
ss             cc              aa          aa  ll             ee
sssssssssssss  cc              aaaaaaaaaaaaa  ll             eeeeeeeee
 sssssssssssss cc              aaaaaaaaaaaaa  ll             eeeeeeeee
          ss   cc              aa          aa  ll             ee
          ss   cc              aa          aa  ll             ee
ss        ss   cc          cc  aa          aa  ll             ee
ssssssssssssss ccccccccccccc   aa          aa  lllllllllllll  eeeeeeeeeeee
 ssssssssssss   ccccccccccc    aa          aa  lllllllllllll  eeeeeeeeeeee

**********************************************************************************
**********************************************************************************
**********************************************************************************
*****                                                                        *****
*****                  program verification information                   *****
*****                                                                        *****
*****                 code system:    scale  version:    4.2              *****
*****                                                                        *****
**********************************************************************************
**********************************************************************************
*****                                                                        *****
*****                                                                        *****
*****              program:  modify                                          *****
*****                                                                        *****
*****        creation date:  10/25/93                                        *****
*****                                                                        *****
*****              library:  /scale4.2p/exe                                  *****
*****                                                                        *****
*****                                                                        *****
*****      production code:  modify                                          *****
*****                                                                        *****
*****              version:  2.1                                             *****
*****                                                                        *****
*****              jobname:  st5                                             *****
*****                                                                        *****
*****    date of execution:  03/14/94                                        *****
*****                                                                        *****
*****    time of execution:  16:46:26                                        *****
*****                                                                        *****
*****                                                                        *****
**********************************************************************************
**********************************************************************************
**********************************************************************************
```

Figure C4.6.6 Program verification table for MODIFY

## C4.6.6 TABLE OF SEARCH DATA

The table of search data, shown in Fig. C4.6.7, lists the search data for the problem. The KENO V.a title is printed at the top of the table. The search parameters are printed in the top portion of the table after the KENO V.a title and below the heading "SEARCH DATA." The search type is the first entry in the search parameters and defines the type of search to be performed. In this example, a critical pitch search is to be performed. PAS is the number of search passes to be performed. The default value is 10. NPM, the number of search parameters, is defaulted to 1 and should not be changed. EPS is the search convergence tolerance and is defaulted to 0.005. If a critical pitch or critical dimension search is to be performed, KEF, the desired value of k-effective, is printed. It is omitted for "optimum" and "minimum" searches. –CON is the value of the minimum constraint and +CON is the value of the maximum constraint. If a "pitch" search is performed and if values are entered for the

```
*** critical pitch search for case 2c8 bare ***

                    ******  search data  ******

                    search type                    critical pitch
pas                 number of search passes              10
npm                 number of search parameters           1
eps                 search convergence tolerance        .0050
kef                 desired k-effective                1.0000
-con                minimum constraint                 -.2070
+con                maximum constraint                  .0693

                  unit   first  last
      command     number region region  ----------  search constants  ----------
                  unit   reg=__ to __   +x or r  -x or +h  +y or -h   -y     +z     -z
 alter/change/modify    1      2      2    1.00      1.00      1.00     1.00   .00    .00
 alter/change/modify    1      2      2     .00       .00       .00      .00  1.00   1.00
```

Figure C4.6.7 Table of search data

minimum allowed pitch, MINPITCH, and the maximum allowed pitch, MAXPITCH, they are printed following +CON.

The bottom portion of the table of search data lists the search command, the unit number, the range of regions in the unit the search command is applicable to, and the search constants for each dimension of those regions. A search constant of 0.0 means the search command should be ignored for that particular dimension. In Fig. C4.6.7, all six dimensions (+X, –X, +Y, –Y, +Z, –Z) of region 2 of unit 1 will be altered.

## C4.6.7 SEARCH PASS ITERATION DATA

This portion of the search output is created as the search progresses. At the end of each KENO V.a calculation, the pass number, k-effective, and deviation are printed, followed by the value of the search parameter. The third line is a statement that modified KENO V.a data have been written on unit 95 to provide data for the next pass to KENO V.a. These three lines are printed for every completed search pass. However, the third line is not printed when convergence is achieved. This output is illustrated in Fig. C4.6.8.

```
**********   search pass  1     keff=  9.62581E-01 + or -  4.66186E-03   **********
                    the parameter was  0.00000E+00
               *****  modified keno v data has been rewritten on unit 95  *****

**********   search pass  2     keff=  9.26968E-01 + or -  4.76267E-03   **********
                    the parameter was  6.92605E-02
               *****  modified keno v data has been rewritten on unit 95  *****

**********   search pass  3     keff=  1.01030E+00 + or -  4.65530E-03   **********
                    the parameter was -7.27719E-02
               *****  modified keno v data has been rewritten on unit 95  *****

**********   search pass  4     keff=  1.00968E+00 + or -  4.18861E-03   **********
                    the parameter was -5.70604E-02
               *****  modified keno v data has been rewritten on unit 95  *****

**********   search pass  5     keff=  9.80129E-01 + or -  5.04216E-03   **********
                    the parameter was -3.87869E-02
               *****  modified keno v data has been rewritten on unit 95  *****

**********   search pass  6     keff=  9.98210E-01 + or -  4.21917E-03   **********
                    the parameter was -5.09593E-02
```

Figure C4.6.8 Search pass iteration data

## C4.6.8 SEARCH CONVERGENCE INFORMATION

If a search achieves convergence, the search convergence information table is printed as shown in Fig. C4.6.9. The KENO V.a title is the first line printed. The pass at which convergence was achieved is printed followed by the equation used in the search to determine k-effective in terms of the parameter, P. The k-effective and deviation from the converged pass are printed, followed by the geometry description from the converged pass.

```
                                critical pitch search for case 2c8 bare
**********************************************************************************************************************************
**********************************************************************************************************************************

                          convergence was achieved on pass  6  the parameter was  -5.09593E-02

                          the equation used in the search was:

                          k-eff = +9.39989E-01 -9.76969E-01*p +7.86001E+00*p**2 +8.33712E+01*p**3

                          k-effective=  9.98210E-01 + or -  4.21917E-03   the corresponding geometry follows;

              media bias      geometry description for those units utilized in this problem
region         num   id

                                                  -----   unit     1   -----

1 cylinder       1   1  radius =  5.7480   +z =  5.3825   -z = -5.3825   centerline is at  x = 0.00000E+00  y = 0.00000E+00

2 cuboid         0   1      +x =  6.8786   -x = -6.8786   +y =  6.8786   -y = -6.8786     +z =  6.5318     -z = -6.5318
**********************************************************************************************************************************
**********************************************************************************************************************************
```

Figure C4.6.9 Search pass iteration data

## C4.6.9 EXTRAPOLATION OF CONVERGED DATA

If a search achieves convergence and has run four or more passes, the data from the search are used to provide an estimate of the parameter that is used to determine the geometry corresponding to the best estimate of the parameter. This last information printed for a search is shown in Fig. C4.6.10.

```
                                critical pitch search for case 2c8 bare

                          based on the preceding data, the best estimate of the parameter is   -5.17118E-02

                          the geometry corresponding to this parameter follows:

            media bias      geometry description for those units utilized in this problem
region       num   id

                                                          -----   unit     1   -----

1 cylinder  1  1  radius = 5.7480  +z = 5.3825  -z = -5.3825  centerline is at  x = 0.00000E+00  y = 0.00000E+00

2 cuboid    0  1      +x = 6.8732  -x = -6.8732 +y =  6.8732  -y = -6.8732     +z =  6.5266     -z = -6.5266

*****************************************************************************************************************
*****************************************************************************************************************
                           *****  modified keno v data has been rewritten on unit 95  *****

*****  convergence has been achieved in the search package.  *****

 control module modify   is complete.
```

Figure C4.6.10 Best estimate of the parameter and geometry

## C4.7 WARNING AND ERROR MESSAGES

CSAS4 contains two types of warning and error messages. The first type of message is from the Material Information Processor which is common to many of the SCALE analytical sequences. These messages are prefixed by MP- followed by a number. The Material Information Processor messages are listed in numerical order in Sect. C4.7.1. The second type of message pertaining to CSAS4 is identified by CS- followed by a number. These messages are listed in numerical order in Sect. C4.7.2. For additional information concerning a message, simply look up the number in this section.

Warning messages appear when a possible error is encountered. It is the responsibility of the user to verify whether or not the data are correct when a warning message is encountered. The functional modules activated by CSAS4 and related sequences will be executed even though a warning message has been generated.

When an error is recognized, an error message is written and an error flag is set so the functional modules will not be activated. The code stops immediately if the error is too severe to allow continuation of input. However, it will continue to read and check the data if it is able. When the data reading is completed, execution is terminated if an error flag was set when the data were being processed. If the error flag has not been set, execution continues. The STOP codes associated with the severe error messages are listed in Sect. C4.7.3.

### C4.7.1 MATERIAL INFORMATION PROCESSOR MESSAGES

The following messages originate in the Material Information Processor, that portion of the analytical sequence that reads the parameter data, standard compositions specification data, and geometry data necessary for preparing the required information for the cross-section processing codes:

MP-1 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold this data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 300 is executed when this message is printed.

MP-2 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold this data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 310 is executed when this message is printed.

MP-3 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 320 is executed when this message is printed.

MP-4 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine SETUPA indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 330 is executed when this message is printed.

MP-5 ***ERROR***SYSTEM GEOMETRY NAME INCORRECT. INSTEAD OF (____), THE USER SHOULD HAVE ENTERED LATTICECELL, MULTIREGION, OR INFHOMMEDIUM AS THE SECOND ITEM ON THE PARAMETER CARD. CHECK SPELLING.

This self-explanatory message is from subroutine SETUPB. The user has misspelled the word defining the type of calculation to be performed, or the parameter data preceding it were entered incorrectly. See Sect. C4.4.3 for assistance in correctly specifying the parameter data.

MP-6 ***ERROR*** _____ NOT FOUND IN STANDARD COMPOSITION LIBRARY. MAKE SURE THE COMPOSITION NAME USED IS IN THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message from subroutine STDCMP indicates that a standard composition name is misspelled, the standard composition specification data were out of order, or the "END" has been omitted from the preceding standard composition specification. Check the standard composition specification data carefully. See Sect. C4.4.4 for assistance with standard composition specification cards. See Sect. M8 for the materials available in the Standard Composition Library.

MP-7 ***ERROR*** THE USER ENTERED A VOLUME FRACTION (VF) OF 0.0 FOR ONE OF THE STANDARD COMPOSITIONS (_____). THIS INDICATES THAT HE INTENDS TO ENTER A NUMBER DENSITY FOR THE STANDARD COMPOSITION. THAT, HOWEVER, CAN ONLY BE DONE WHEN THE STANDARD COMPOSITION NAME REPRESENTS A SINGLE NUCLIDE (LIKE H, O, PB, B-10, U-238, ETC.). CHECK THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message is from subroutine STDCMP. Check the Standard Composition Library, Sect. M8, for the available standard composition names. See Sect. C4.4.4 for the correct method of specifying standard composition specification data.

MP-8 ***ERROR*** CELL TYPE NAME (_____) IS INCORRECT. CHECK THE SPELLING. THE ALLOWED NAMES ARE: SQUAREPITCH, TRIANGPITCH, SPHSQUAREP, SPHTRIANGP, SYMMSLABCELL, ASYMSLABCELL, ASQUAREPITCH, ATRIANGPITCH, ASPHSQUAREP, ASPHTRIANGP, ASQP, ATRP, ASSP, ASTP.

This self-explanatory message is from subroutine LATCEL. The data are misspelled or out of order. Some data items may be missing, or extra ones may have been entered. Errors in the standard composition specification data can cause this message to be printed. See Sect. C4.4.6 for additional assistance.

MP-9 ***ERROR*** THE NUMBER OF ZONES (IZM=_____) IS INCORRECT FOR ASYMSLABCELL GEOMETRY. FOR AN ASYMMETRIC LATTICECELL CALCULATION, ONE SHOULD HAVE IZM=3, 5, OR 7, DEPENDING ON THE MATERIALS PRESENT.

This self-explanatory message is from subroutine LATCEL. See Sect. C4.4.6 for an explanation of the ASYMSLABCELL geometry data. A code error is responsible for this message.

MP-10 ***ERROR*** INSUFFICIENT DATA SUPPLIED ON THE GEOMETRY DESCRIPTION CARD.

This message from subroutine LATCEL can be caused by having an error in the standard composition specification data, thereby causing the reading routines to get out of phase. If no error messages precede this message, it may indicate that a code error exists. The number of material zones (IZM) has been destroyed or altered in some manner.

MP-11 ***ERROR*** MIXTURE _____ IN THE CSAS GEOMETRY DESCRIPTION WAS NOT SPECIFIED IN THE STANDARD COMPOSITION DATA.

This message from subroutine LATCEL indicates that a mixture number used in the geometry description was not defined in the standard composition specification data. Check the input data for consistency. If the specified mixture number is zero and if the problem is a LATTICELL problem using mixture zero for the clad or moderator, it will be necessary to specify a low-density mixture in the standard composition data and use it in the cell specification.

MP-12 ***ERROR*** THE GAP WAS SPECIFIED AS MIXTURE _____ ON THE LATTICECELL. IT SHOULD BE ZERO OR A VALID MIXTURE NUMBER.

This message from subroutine LATCEL indicates that the mixture number used for the GAP was not defined in the standard composition specification data. Check the input data for consistency.

MP-13 **WARNING** NUMBER OF ENTRIES ON THE GEOMETRY DESCRIPTION CARD EXCEEDS THE NUMBER EXPECTED. THIS MESSAGE MAY BE GENERATED IF THE USER FORGOT TO ENTER THE END ON THE GEOMETRY DESCRIPTION CARD.

This message from subroutine LATCEL may result from a code error or from omitting the word "END" from the geometry description card.

MP-14 ***ERROR*** THE COORDINATE SYSTEM NAME (_____) IS INCORRECT. FOR THIS MULTIREGION CALCULATION, THE USER SHOULD HAVE ENTERED SLAB, CYLINDRICAL, SPHERICAL, BUCKLEDSLAB OR BUCKLEDCYL ON THE GEOMETRY DESCRIPTION CARD. CHECK SPELLING.

This self-explanatory message is from subroutine MULTRG.

MP-15 ***ERROR*** ALPHANUMERIC DESCRIPTION OF THE RIGHT-HAND BOUNDARY CONDITION (____) IS INCORRECT. ONLY VACUUM, REFLECTED, PERIODIC, AND WHITE BOUNDARY CONDITIONS ARE ALLOWED. CHECK SPELLING. CHECK GEOMETRY DESCRIPTION CARD INPUT INSTRUCTIONS FOR ADDITIONAL ASSISTANCE.

This self-explanatory message is from subroutine MULTRG. See Sect. C4.4.6 for additional assistance.

MP-16 ***ERROR*** ALPHANUMERIC DESCRIPTION OF THE LEFT-HAND BOUNDARY CONDITION (____) IS INCORRECT. ONLY VACUUM, REFLECTED, PERIODIC, AND WHITE BOUNDARY CONDITIONS ARE ALLOWED IN SLAB GEOMETRY. IN THE CASE OF CYLINDRICAL OR SPHERICAL GEOMETRY, ONLY THE REFLECTED BOUNDARY CONDITION IS ALLOWED ON THE LEFT. CHECK SPELLING. CHECK GEOMETRY DESCRIPTION CARD INPUT INSTRUCTIONS FOR ADDITIONAL INSTRUCTIONS.

This self-explanatory message is from subroutine MULTRG. See Sect. C4.4.6 for additional assistance.

MP-17 ***ERROR*** MIXTURE _____ FROM THE MULTIREGION DATA WAS NOT SPECIFIED IN THE STANDARD COMPOSITION DATA.

This message is from subroutine MULTRG. It indicates that a mixture number specified in the multiregion data was not defined in the standard composition specification data. Check the input data for consistency.

MP-18 ***ERROR*** USER MADE AT LEAST ____ DATA ERRORS. PROBLEM WILL NOT BE RUN. HOPEFULLY, EACH OF THESE ERRORS WILL HAVE GENERATED ITS OWN SELF-EXPLANATORY ERROR MESSAGE. IF NOT, CHECK DATA CAREFULLY BEFORE RESUBMITTING.

This self-explanatory message is from subroutine SETUPB. Check through the problem printout and correct all errors that have occurred.

MP-19 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message is from subroutine SETUPB. It indicates that the allocated computer storage will not hold the data that must be loaded to calculate the information that must be passed to BONAMI. The first number is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that is required to hold these data. The third number is the minimum additional words of storage required to hold these data. Subsequent portions of the program may require even more storage. Increase the region size in the "go step" of the job control language and resubmit the problem. A STOP 390 is executed when this message is printed.

MP-20 ***ERROR*** MIXTURE NUMBER_____ ON A STANDARD COMPOSITION CARD IS LESS THAN 1.

This message is from subroutine SETUPB. A mixture number specified on a standard composition specification card must always be greater than zero. This message is triggered if a MP-6 message is printed. Fixing the MP-6 error should eliminate this message. If the printed mixture number is a positive number, a code error is responsible for the message.

MP-21 ***WARNING*** STANDARD COMPOSITION CARD NUMBER _____ IMPLIES THAT MIXTURE NUMBER____ HAS A TEMPERATURE OF ____ DEGREES KELVIN, WHEREAS STANDARD COMPOSITION SPECIFICATION CARD _____ IMPLIES THAT MIXTURE NUMBER____ HAS A TEMPERATURE OF ____ DEGREES KELVIN. (ONE OF THESE MAY HAVE BEEN THE VALUE SPECIFIED BY DEFAULT.) THE CODE WILL ASSUME THE HIGHER OF THE TWO AND PROCEED. IF THIS IS NOT SATISFACTORY, THE USER SHOULD ENTER THE CORRECT TEMPERATURE ON EACH OF THE STANDARD COMPOSITION SPECIFICATION CARDS INDICATED.

This self-explanatory message is from subroutine SETUPB.

MP-22 **WARNING** STANDARD COMPOSITION SPECIFICATION CARD(S) MISSING FOR MIXTURE NUMBER____.

This message from SETUPB is printed to remind the user that the mixture numbers specified in the standard composition specification data are not contiguous. This may result in wasted storage space, but is not necessarily indicative of an error.

MP-23 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MTABLE indicates that the allocated computer storage will not hold the nuclide ID numbers that will be processed by subroutine IDENTS. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 340 is executed when this message is printed.

MP-24 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 350 is executed when this message is printed.

MP-25 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 360 is executed when this message is printed.

MP-26 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 370 is executed when this message is printed.

MP-27 ***ERROR*** THE NUMBER OF MIXTURES FOUND IN THE MIXING TABLE IS INCONSISTENT WITH THE INPUT DATA. THE NUMBER FOUND IS _____, BUT THE INPUT DATA INDICATED _____.

This message from subroutine REORDR indicates that an error was found in the mixing table during operations to reorder and load it in permanent storage. The number of mixtures found in the mixing table should be the same as the number indicated by the input data. A code error is the likely cause of this message. A STOP 380 is executed when this message is printed.

MP-28 ***ERROR*** NUCLIDE ID _____ SPECIFIED IN THE INPUT DATA WAS NOT FOUND IN THE STANDARD COMPOSITION LIBRARY.

This message is from subroutine IDENTS or subroutine SLNS. A STOP 395 is printed with this message. Check the standard compositions specification data, Sect. C4.4.4, and ensure that the "ZA" ID numbers were entered correctly. This message is also printed if an arbitrary material specifies a value other than 0 or 1 for the MULTIPLE ISOTOPE INDICATOR (IVIS in Sect. C4.4.4).

MP-29 ***ERROR*** AN ARGUMENT OF ____ IS INVALID FOR AN ARCSIN OR ARCCOS.

This message from subroutine ARCHK is indicative of a code error. ARCHK is called from subroutine DAN to ensure that proper arguments are supplied to the ARCSIN and ARCCOS functions in determining cylinder Dancoff factors. This error can be caused by specifying a center-to-center spacing that is smaller than is physically possible.

MP-30 ***ERROR*** THE MATRIX WAS SINGULAR WHEN THE NUMBER FRACTION OF AN ISOTOPE IN AN ELEMENT WAS BEING DETERMINED. MAKE SURE THE SPECIFIED ISOTOPE DISTRIBUTION (WEIGHT PERCENT) SUMS TO 100.0.

This message from subroutine NOFRAC is caused by an isotope distribution that does not sum to 100% for an element. Entry A7 of Sects. C4.4.4 or M7.4.4 illustrates the correct procedure for specifying the weight percent of the various isotopes in an element.

MP-31 ***ERROR*** THE CALCULATED NUMBER FRACTIONS DO NOT REPRODUCE THE WEIGHT PERCENTS ENTERED AS INPUT DATA OR OBTAINED FROM THE STANDARD COMPOSITION LIBRARY.

WTFRAC=nnnnnnnnnn NOFRAC=mmmmmmmmmm MASS=ℓℓℓℓℓℓℓℓℓℓ

This message is printed from subroutine NOFRAC. WTFRAC, NOFRAC, and MASS are printed for each isotope in the standard composition. WTFRAC is the specified input weight percent for the isotope converted to weight fraction instead of weight percent. NOFRAC is the adjusted atom percent converted to atom fraction for the isotope corresponding to a total weight percent of 100% for the isotopes in the standard composition. MASS is the atomic weight of the isotope. The adjusted isotopic weight percents are not printed in MP-31 but can be calculated according to:

$$ADJ_i = \frac{NOFRAC_i \times MASS_i}{\sum_i NOFRAC_i \times MASS_i} \times 100, \qquad \text{(C4.7.1)}$$

where

$ADJ_i$ is the adjusted weight percent of isotope i,

$NOFRAC_i$ is the number fraction of isotope i printed in MP-31,

$MASS_i$ is the atomic weight of isotope i printed in MP-31.

Message MP-31 is printed if the adjusted weight percent of an isotope differs by 0.4% or more from the isotopic weight percent specified for that particular isotope in the standard composition data. An error flag is set, and the

problem will not execute. If the isotopic weight percents specified in the input data do not sum to 100%, the weight percent of the first isotope whose specified weight percent is greater than 0.1% is adjusted to force the total weight percent to sum to 100%. If this adjustment results in a calculated weight percent for that isotope that is less than 0.4% different from that specified in the input data, the problem is run using the adjusted data. If message MP-31 is printed, correct the isotopic weight percents specified in the standard composition input data to sum to 100 or specify the isotope with the largest weight percent first, and resubmit the problem.

MP-32 ***ERROR*** END COMP WAS EXPECTED, BUT _____ WAS READ INSTEAD.

This message from subroutine SETUPA is printed if the data is out of order or the keywords END COMP or END COMPOSITIONS are omitted after the last standard composition specification. An END followed by blanks is also allowed. Check the data and the spelling of the END COMP. See Sect. C4.4.4.

MP-33 ***ERROR*** END OF FILE WAS ENCOUNTERED WHILE READING MULTIREGION ZONE DESCRIPTION. VERIFY THAT THE END ZONE CARD WAS ENTERED.

This self-explanatory message from subroutine SETUPA is printed if the END ZONE is omitted following the MULTIREGION zone description data. See Sect. C4.4.7.

MP-34 ***ERROR*** END ZONE WAS EXPECTED, BUT ____ WAS READ INSTEAD.

This message from subroutine SETUPA is printed if data is out of order or the keywords END ZONE are omitted at the end of the MULTIREGION zone description data. Check the data and the spelling of the END ZONE. See Sect. C4.4.7.

MP-35 ***ERROR*** END OF FILE WAS ENCOUNTERED WHEN THE GEOMETRY DESCRIPTION CARD WAS EXPECTED.

This self-explanatory message from subroutine SETUPA is printed if data are omitted or out of order. See Sect. C4.4.

MP-36 ***ERROR*** ILLEGAL SOLUTION NAME _____. ALLOWED NAMES INCLUDE SOLNUO2F2, SOLNUO2(NO3)2 AND SOLNPU(NO3)4.

This self-explanatory message from subroutines SLNS and SLNAME is printed if a solution name is misspelled or data are out of order. See Sect. C4.4.4 for assistance in specifying solution names correctly.

MP-37 ***ERROR*** ____ IS AN INVALID NAME WHEN READING MORE DATA.

This message is printed from subroutine MOREDT when extra parameter data is to be entered as explained in Sect. C4.4.8. The data are out of order, or a parameter name is misspelled.

MP-38 ***ERROR*** INVALID GEOMETRY WORD _____. ONLY INFHOMMEDIUM, MULTIREGION, OR LATTICECELL ARE ALLOWED.

This self-explanatory message is from subroutine SETUPA. Correct the input data and resubmit the problem. Check the spelling of the geometry word, ensure that the data are in order, and resubmit the problem.

MP-39 ***WARNING*** AN END DATA WAS READ BEFORE IT WAS EXPECTED. AN END WAS PROBABLY OMITTED FROM A STANDARD COMPOSITION.

This message from subroutine SETUPA indicates that an "END" was omitted from a standard composition specification or that the END for a standard composition specification was followed by COMP. The first four characters following the END on a standard composition specification card cannot be COMP. The words END COMP terminate the standard composition specification data. Check the standard composition specification data carefully.

MP-40 ***ERROR*** NUCLIDE ID ____ IN MIXTURE ____ DOES NOT OCCUR IN THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message is from subroutine REORDR. A STOP 385 is executed in conjunction with this message.

MP-41 ***ERROR*** A NUCLIDE ID OF _____ IS INVALID FOR _____ IN THE FOLLOWING STANDARD COMPOSITION SPECIFICATION.

This message from subroutine STDCMP indicates that a nuclide ID is not compatible with a standard composition name. For example: UO2 3 1 300 94240 50 94238 50 END would cause this error because plutonium is invalid for a uranium standard composition specification. If this message is printed for a solution, the specific gravity of the solution may have been entered without the required keyword SPG=.

MP-42 ***ERROR*** AN END WAS ENCOUNTERED BEFORE MORE DATA WAS COMPLETED.

This message from MOREDT indicates incomplete input for MORE DATA. See Sect. C4.4.8 or M7.4.8 for assistance.

MP-43 ***WARNING*** BAL=_____ IS AN INVALID BALANCE TABLE SPECIFICATION. ALL BALANCE TABLES WILL BE PRINTED.

This message from subroutine MOREDT indicates that the value entered following BAL= is incorrect. Available options are NONE, FINE, and ALL. If any other value is entered, this message will be printed.

MP-44 *** ERROR *** NUCLIDE ____ IS NOT A MULTIPLE ISOTOPE NUCLIDE ***ERROR***.

This message from subroutine STDCMP indicates that a value greater than zero was entered for the multiple isotope indicator (IVIS) in an arbitrary material and that the first ID NUMBER (NCZA) is not a multiple isotope nuclide (see Sect. C4.4.4). When this message is printed, an error flag is set to prevent execution and IVIS is set to zero to allow the code to continue checking the remaining input data. If an arbitrary material was not specified, this message indicates that the Standard Composition Library is in error for the standard composition printed immediately following this message.

MP-45 ***** ERROR ***** ____ IS NOT A VALID CALCULATION TYPE. USE INFHOMMEDIUM, LATTICECELL, OR MULTIREGION. EXECUTION IS TERMINATED.

This message from subroutine CSPARM indicates that the TYPE OF CALCULATION (see item 3 of Sect. C4.4.3) was spelled incorrectly, entered out of order, or was improperly specified.

MP-46 *** ERROR *** THE FUELOD CANNOT BE GREATER THAN THE PITCH.

This self-explanatory message from subroutine LATCEL results from an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem (see Sect. C4.4.6). The value of PITCH must be at least as large as the value of FUELOD.

MP-47 *** ERROR *** THE CLADOD CANNOT BE LESS THAN THE FUELOD, NOR GREATER THAN THE PITCH.

This message from subroutine LATCEL indicates an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem. See Sect. C4.4.5 and verify the values of PITCH, FUELOD, and CLADOD.

MP-48 *** ERROR *** THE GAPOD CANNOT BE LESS THAN THE FUELOD, NOR GREATER THAN THE CLADOD.

This message from subroutine LATCEL indicates an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem. See Sect. C4.4.6 and verify the values of FUELOD, CLADOD, and CLADID (note that CLADID is the GAPOD and is equal to FUELOD + twice the gap thickness).

MP-49 *** ERROR *** THE ABOVE RADIUS MUST BE GREATER THAN THE PREVIOUS RADIUS.

This message from subroutine MULTRG indicates an error in the GEOMETRY SPECIFICATION for a MULTIREGION problem. See Sect. C4.4.7 and check to ensure that each value of RZ is greater than the previous value of RZ.

MP-50 *** WARNING *** IN MORE DATA, RESONANCE INFORMATION (RES=) WAS SUPPLIED FOR MIXTURE _____. IT WILL NOT BE USED BECAUSE THAT MIXTURE OCCURS IN THE CELL DESCRIPTION.

This message from subroutine MOREDT is printed if resonance data (RES=) are entered for a mixture in MORE DATA (Sect. C4.4.8) and are also used in the cell description (Sects. C4.4.6 and C4.4.7).

MP-51 *** ERROR *** IN MORE DATA _____ IS AN INVALID GEOMETRY TYPE FOR USE IN SPECIFYING RESONANCE DATA.
CYLINDER, SPHERE, OR SLAB MUST BE ENTERED.

This message from subroutine MOREDT is printed if resonance data (RES=) are entered in MORE DATA (Sect. C4.4.8) and an incorrect shape is specified. The correct specification for a sphere of mixture 3 having a radius of 1.5 cm is: RES=3 SPHERE 1.5

MP-52 *** WARNING *** IN MORE DATA, DANCOFF INFORMATION (DAN( )=) WAS SUPPLIED FOR MIXTURE _____. THIS VALUE WILL BE USED INSTEAD OF CALCULATING THE DANCOFF FROM THE CELL DESCRIPTION.

This message from subroutine MOREDT is printed to alert the user that the Dancoff information calculated from the unit cell (Sects. C4.4.6 and C4.4.7) specification is being overridden by the value that was entered in MORE DATA (Sect. C4.4.8).

MP-54 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the volume fraction data from subroutine VERIFY. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-55 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the mixing table data required for subroutine BONAMI. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-56 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the mixture number, outer radii, temperature, sigma-H, and zone identifiers for each BONAMI zone. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-57 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the BONAMI control parameters. The first number printed is the amount of storage, in words, needed to hold these

data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-58 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the cross-section data required by subroutines EPSIG and XSMESH. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-59 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the data required by subroutine XSDRN. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-60 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the data required by subroutine ICE. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-61 *** ERROR *** THE CROSS-SECTION LIBRARY NAME ENTERED IN THE MATERIAL INFORMATION PROCESSOR DATA FOR THIS CSAS PROBLEM IS INCORRECT. SEE TABLE M4.1.1 OF THE SCALE MANUAL.

This message from subroutine C4DATA indicates that an invalid CROSS-SECTION LIBRARY NAME (see item 2 of Sect. M7.4.3) was entered. The name was misspelled, out of order, or improperly specified. See Table M4.1.1 for a list of valid library names.

MP-62 ***ERROR*** A MIXTURE NUMBER IS OUT OF RANGE. ENTRY _____ IN THE MIXING TABLE IS _____. IT SHOULD FALL BETWEEN 1 AND _____, THE LARGEST NUMBER USED IN THE STANDARD COMPOSITION DATA.

This message is from subroutine EPSIG. Because the largest mixture number is determined by the code as the input data are read, this message may be the result of a code error introduced when changes were made to the code.

MP-63 ***ERROR*** A POINTER INTO THE NUCLIDE DATA IS OUT OF RANGE. ENTRY ___ IN THE MIXING TABLE IS ____. IT SHOULD FALL BETWEEN 1 AND ____, THE NUMBER OF NUCLIDES IN THE STANDARD COMPOSITION LIBRARY.

This message is from subroutine EPSIG. It may result from a code error introduced when changes were made to the code.

MP-64 TAPE COPY NEEDS _____ WORDS, BUT ONLY HAS _____ WORDS.

This message from subroutine SHORTX indicates that the allocated computer storage is insufficient to create the short cross-section library utilized by the Materials Information Processor. Increase the region size for the "go step" and resubmit the problem. A STOP 200 is executed when this message is printed.

MP-65 ***** ERROR ***** THE FOLLOWING NUCLIDES WERE NOT FOUND ON THE MASTER LIBRARY:

This message from subroutine SHORTX indicates that the user specified a standard composition that contains a nuclide that is not available on the specified master library. For example, nuclide ID 25304, which is stainless-steel-weighted manganese, is not available on any of the libraries but is listed in the Standard Composition Library. 25055 could be used instead.

MP-66 ***ERROR*** THE MIX. NO. (MIXZ=_____) ASSIGNED TO ZONE IZ=_____ WAS OUTSIDE THE RANGE 0 TO MXX=_____.

This message from subroutine VERIFY indicates that the mixture number is out of range. A code error is the likely cause of this error.

MP-67 **WARNING** MIXTURE ____ WAS ASSIGNED TO ZONES _____ AND _____. THIS MAY BE CORRECT IF THE MIXTURE HAS NO RESONANCE NUCLIDES AND HAS THERMAL SCATTERING DATA AT ONLY ONE TEMPERATURE. USUALLY, UNIQUE MIXTURE NUMBERS SHOULD BE USED IN EACH ZONE, EVEN THOUGH THE MIXTURE SPECIFICATIONS MAY BE IDENTICAL. THIS ALLOWS BONAMI AND NITAWL TO PRODUCE SEPARATE CROSS SECTIONS BASED ON THE APPROPRIATE NUMBER OF EXTERNAL MODERATORS AND THE ACTUAL MEAN CHORD LENGTH FOR EACH SPECIFIED ZONE AS WELL AS THE ACTUAL TEMPERATURE FOR THAT ZONE.

This self-explanatory message is from subroutine VERIFY. The user should verify that the multiple use of the stated mixture number was intentional and correct.

MP-68 ERRONEOUS MIXING TABLE LENGTH FOR XSDRN.

This message from subroutine XSDRN is indicative of a code error. The computed mixing table length, MS, passed to subroutine XSDRN, does not match the counted length in subroutine XSDRN.

MP-69 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

This message from subroutine XSMESH indicates that the allocated computer storage will not hold the interval boundaries and zone number for each interval. The first number printed is the amount of storage, in words, needed to hold this data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. Specification (by the control module) of too many mesh intervals in a given zone is the most frequent cause of failure. In MULTIREGION calculations where one has a large zone containing a strong absorber (such as a tank of uranyl-fluoride solution), it may be necessary to reduce the number of mesh intervals by increasing the size factor (SZE) in the optional parameter data (Sect. C4.4.8). Making this factor large enough (SZF=10, 15, 20, 30, etc.) will generally allow any problem to run in the space allocated. Once the problem runs, one should check the number density of all strong absorbers to see if they are realistic. Occasionally a user will accidentally specify an unrealistic concentration of a strong absorber in one or more zones. Such a mistake dramatically increases the number of mesh intervals the code thinks it needs.

MP-70 ***ERROR*** MORE THAN ONE ZONE WAS ENCOUNTERED FOR THIS INFINITE HOMOGENEOUS MEDIA PROBLEM.

This self-explanatory message is from subroutine XSMESH. Correct the input data and resubmit the problem.

MP-71 ***WARNING*** THE TEMPERATURE FOR STANDARD COMPOSITION _____ WAS SPECIFIED AS _____ DEGREE CELSIUS.
DENSITY DATA AT 25 DEGREES CELSIUS WILL BE USED FOR THIS SOLUTION.

This message from subroutine SLNS is printed if SOLNUO2F2 or SOLNPU(NO3)4 were specified in the standard composition data with a temperature outside the range of 15 to 30°C. The Standard Composition Library does not contain data for these solutions outside that temperture range.

MP-72 *** ERROR *** KEYWORD SPG= TO SPECIFY THE SPECIFIC GRAVITY OF THE SOLUTION WAS EXPECTED. KEYWORD _____ WAS READ INSTEAD.

This message from subroutine STDCMP indicates an error in the specification of solution data in the standard composition specification. If the specific gravity of a solution is to be entered, it is the fourth entry after the solution name. Check the solution input data carefully.

MP-73 ***ERROR*** EXTERNAL MODERATOR INDEX IS INCORRECT IN ZONE _____ OF THE MULTIREGION DATA. ENTER NOEXTERMOD, ONE EXTERMOD, OR TWOEXTERMOD INSTEAD OF (_____). CHECK SPELLING.

This self-explanatory message is printed from subroutine MULTRG and subroutine RESDA. The user has misspelled the word defining the external moderator index of the specified zone or the MULTIREGION data are out of order. See Sect. M7.4.4.7.

MP-74 *** ERROR *** FIRST MODERATOR HAS NEGATIVE THICKNESS

This message from subroutine DANCOF indicates that the dimensions specified for an ASYMSLABCELL are inconsistent. The array pitch (PITCH) is smaller than the sum of the second moderator thickness (TKMOD2) and *MAX* where *MAX* is the outside diameter of the clad (CLADOD) if entered; otherwise, *MAX* is the outside diameter of the fuel (FUELOD). Correct the data and resubmit the problem. See Sect. M7.4.6 and Fig. M7.5.6.2 for assistance.

MP-75 *** ERROR *** KEYWORD DEN= OR SPG= TO SPECIFY THE DENSITY OF THE BASIC STANDARD COMPOSITION WAS EXPECTED. KEYWORD _____ WAS READ INSTEAD.

This message from subroutine STDCMP indicates an error in the specification of the density in the basic standard composition data. If the density of the material is to be entered, it is the second entry after the standard composition name. The density is optional data. If the density is entered, it is multiplied by the volume fraction to obtain the density actually used for this composition.

MP-76 *** ERROR *** THE LOWER BOUND OF XSDRN INTERVAL _____ IS _____WHICH MUST BE LESS THAN THE UPPER BOUND OF _____.

This message from subroutine XSMESH indicates that the specified XSDRN mesh interval is incorrect. Each mesh interval must have a finite positive thickness. This message can be caused by an error in the unit cell data. Either a zero-thick region was specified or the unit cell dimensions were incorrectly specified. Carefully check the unit cell dimensions. If a LATTICECELL annular cell is specified, verify that the FUELOD is at least as large as TKMOP2+2(CLADOD-FUELOD).

MP-77 A NEGATIVE DIMENSION WAS ENCOUNTERED IN THE UNIT CELL DATA. CHECK AND CORRECT THE GEOMETRY DIMENSION.

This message from subroutine LATCEL is printed if a negative number is read for a cell dimension. Check the printed problem geometry and correct the input data (Sect. M7.4.6).

MP-78 THE FUEL THICKNESS IS NEGATIVE. THE ANNULAR CELL DIMENSIONS FOR TKMOD2, CLADOD, AND FUELOD ARE INCONSISTENT.

This message from subroutine LATCEL indicates an error in the description of a cylindrical or spherical annular cell. The cell description must provide a finite fuel thickness. The ID of the fuel is calculated using the diameter of the second moderator, TKMOD2, and the OD of the clad, CLADOD. If the calculated ID of the fuel is greater than or equal to the specified OD of the fuel, there is no fuel in the cell and this message is printed. The

thickness of the gap and clad is ½(CLADOD–FUELOD), so the ID of the fuel is TKMOD2+CLADOD–FUELOD. See Sect. M7.4.6 for data input instructions.

MP-79 *** ERROR *** THE ORIGIN MUST BE 0.0 FOR SPHERICAL OR CYLINDRICAL GEOMETRY.

This message from subroutine MULTRG is printed if the origin, ORGN of Table M7.4.8, Sect. M.4.7, is nonzero for a cylinder or sphere. Either the MULTIREGION unit cell data are out of order or a nonzero number has been entered for ORGN. Correct the data and resubmit the problem.

MP-80 *** ERROR *** NOT ENOUGH STORAGE IN CLAPSE – FIRST NUMBER IS THE AMOUNT NEEDED, SECOND IS THE AMOUNT AVAILABLE.

PERTINENT CONSTANTS

This message from subroutine CLAPSE indicates that the allocated computer storage is insufficient to allow collapsing the thermal cross sections to one group for shielding calculations. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated. A STOP 201 is executed when this message is printed.

MP-81 *** ERROR *** DANCOFF FACTOR MUST BE BETWEEN 0.0 AND 1.0.

This message from subroutine MOREDT is printed if DAN(n)= is specified in MORE DATA with a value greater than 1.0 or less than 0.0. Dancoff factors must lie between 0 and 1. The problem will not be run if message MP-81 is printed. Correct the data and resubmit the problem. See Sect. M7.4.7 for assistance.

MP-82 THE FIRST MODERATOR THICKNESS IS NEGATIVE. THE ASYMMETRIC SLAB CELL DIMENSIONS FOR TKMOD2, CLADOD, AND PITCH ARE INCONSISTENT

This message from subroutine LATCEL indicates an error in the description of an asymmetric slab cell. The thickness of the first moderator is not specified in the input data and is calculated as PITCH–TKMOD2/2–CLADOD. If the thickness of the first moderator is negative, this message is printed.

MP-83 *** ERROR *** AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE MATERIAL INFORMATION PROCESSOR DATA ENTERED FOR THIS PROBLEM.

This message from subroutine C4DATA is printed if the subroutine library routine LRDERR returns a value of "TRUE," indicating that a reading error was encountered in the Material Information Processor input data. Locate the unnumbered message stating "***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****." Correct the data and resubmit the problem.

MP-84 *** WARNING *** FOR MIXTURE ____ THE DANCOFF FACTOR OF ____ IS OUTSIDE THE RANGE OF 0 TO 1. IT WILL BE RESET TO THE CLOSER BOUND.

This message from subroutine RESDA is printed when the DANCOFF factor is less than $10^{-4}$ outside the allowed range of 0 to 1. This is usually due to roundoff in calculating the Dancoff factor.

MP-85 *** ERROR *** FOR MIXTURE _____ THE DANCOFF FACTOR OF _____ IS OUTSIDE THE RANGE OF 0 TO 1. IT WILL BE RESET TO THE CLOSER BOUND. THE PROBLEM WILL NOT BE EXECUTED.

This message from subroutine RESDA is printed when the Dancoff factor is more than $10^{-4}$ outside the allowed range of 0 to 1. Check the input data to be sure that the specified Dancoff factor is between 0 and 1. See Sect. M7.4.8 or C4.4.8 for input specifications for the Dancoff factor. If the Dancoff factor was not specified in the optional parameter data, the message is due to a code error.

MP-86 ***ERROR*** THE FUEL DENSITY FOR SOLUTION ____ IN MIXTURE ____ SPECIFIES A FUEL DENSITY OF _____ G/L, WHICH EXCEEDS THE FUEL DENSITY OF A HYDRATED CRYSTAL, AND WILL LEAD TO ERRONEOUS DENSITIES WHEN USED AS A SOLUTION

This message from subroutine SLNS is printed only for solution standard compositions (those that begin with SOLN) that are incorrectly specified. Each solution standard composition is composed of (1) a fuel-salt, fs, (2) water, $\mathbf{H_2O}$, and (3) an acid. A fully hydrated fuel-salt is defined as $fs \cdot nH_2O$ where **n** defines the number of water molecules associated with the fuel-salt crystal (water of hydration). The values of **n** are:

| **n** | Solution | Hydrated Salt |
|---|---|---|
| 2 | $UO_2F_2$ | $UO_2F_2 \cdot 2H_2O$ |
| 6 | $UO_2(NO_3)_2$ | $UO_2(NO_3)_2 \cdot 6H_2O$ |
| 5 | $Pu(NO_3)_4$ | $Pu(NO_3)_4 \cdot 5H_2O$ |

If the fuel density is so large that there is not enough water in the solution to form fully hydrated crystals of fuel-salt, this message will be printed. The solution name, mixture number, and specified fuel density are printed in the error message. The problem will not run if this message is printed. Correct the input data for this solution standard composition in the specified mixture and resubmit the problem.

MP-87 ***WARNING*** THE FUEL DENSITY FOR SOLUTION ____ IN MIXTURE ____SPECIFIES A FUEL DENSITY OF ___ G/L. THIS MAY BE GREATER THAN THE SATURATED SOLUTION FUEL DENSITY, AND MAY GIVE INCORRECT DENSITIES AS A SOLUTION. CHECK THE DENSITIES CAREFULLY.

This message from subroutine SLNS is printed because the fuel density for the specified solution standard composition may be high enough to allow precipitation of the fuel-salt. Each solution standard composition is composed of (1) a fuel-salt (**fs**), (2) water ($\mathbf{H_2O}$), and (3) an acid. The water/fuel-salt ratio, **n**, is the ratio of the moles of water to moles of fuel-salt. The minimum threshold values of **n** used to trigger this message are:

| **n** | Solution |
|---|---|
| 8 | $UO_2F_2$ |
| 17 | $UO_2(NO_3)_2$ |
| 17 | $Pu(NO_3)_4$ |

The solution name, mixture number, and specified fuel density are printed in the error message. The problem will run and it is the responsibility of the user to verify that the resulting solution is correct.

MP-88 ***ERROR*** THE LIBRARY SPECIFIED IS A PRE-NITAWL-II LIBRARY

This error message from subroutine SETUPA indicates the specified library was created to be used with a version of NITAWL prior to NITAWL-II. See Table M4.1.1 for a list of acceptable cross-section libraries. If the library is correctly specified, its format is not consistent with the NITAWL-II format.

MP-89 ***ERROR*** THE MATERIAL SPECIFIED IN THE _____TH STANDARD COMPOSITION SPECIFICATION DOES NOT CONTAIN ANY MULTIPLE ISOTOPE NUCLIDES. ISOTOPIC SPECIFICATIONS ARE NOT ALLOWED.

This error message from subroutine SCDATA indicates that the nth standard composition specification does not contain multiple isotopic nuclides but isotopic data (NCZA and ATPM) were included. It could possibly indicate a missing END statement. When determining the nth specification, arbm and basic standard composition specifications count as one each and solution specifications count as three each (fuel salt, water, and an acid).

MP-90 ***ERROR*** TOO MANY ISOTOPES WERE SPECIFIED FOR THE _____TH ELEMENT OF THE _____TH STANDARD COMPOSITION SPECIFICATION.

This error message from subroutine SCDATA indicates the nth element of the mth standard composition specification contains more isotopes than are allowed. Reduce the number of isotopes used to describe the material. When determining the mth standard composition specification, arbitrary materials and basic standard composition specifications count as one each and solution specifications count as three each (fuel salt, water, and an acid).

MP-91 ***ERROR*** THE USER SPECIFIED A NEGATIVE VOLUME FRACTION FOR _____.

This self-explanatory error message from subroutine STDCMP indicates a negative density multiplier or volume fraction (VF). It was entered for the standard composition named in the message.

MP-92 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0 MIXTURE NUMBER FOR ___.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero mixture number (MX) was entered for the standard composition named in the message.

MP-93 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0 DENSITY FOR _____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero density or specific gravity (SPGR or ROTH) was specified for the standard composition named in the message.

MP-94 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0.0 FUEL DENSITY FOR SOLUTION _____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero fuel density (FD) was specified for the solution standard composition named in the message.

MP-95 ***ERROR*** THE USER SPECIFIED A NEGATIVE ACID MOLARITY FOR SOLUTION _____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero acid molarity (AML) was specified for the solution standard composition named in the message.

### C4.7.2 MESSAGES FROM CRITICALITY SAFETY SEQUENCE NO. 4

The following messages originate in the part of CSAS4 that reads, checks, and prepares data for KENO V.a and the search module MODIFY.

CS-16 ***WARNING*** READ FLAG NOT FOUND. ASSUME KENO V PARAMETER DATA FOLLOWS.

This message from subroutine CPARAM indicates that the word READ is not the first word of KENO V.a data following the KENO V.a title. If parameter data is to be entered, the code expects the words READ PARAMETERS to precede the parameter input data. If the word READ is not the first word, the code assumes the data are parameter input data.

CS-21 A UNIT NUMBER WAS ENTERED FOR THE CROSS-SECTION LIBRARY. (LIB= IN PARAMETER DATA.) THE DEFAULT VALUE SHOULD BE USED IN ORDER TO UTILIZE THE CROSS SECTIONS GENERATED BY CSAS. MAKE CERTAIN THE CORRECT CROSS-SECTION LIBRARY IS BEING USED.

This message is from subroutine CPARAM. It indicates that a value has been entered for the cross-section library in the KENO V.a parameter data. The cross-section library created by the analytical sequence should be used. MAKE CERTAIN THAT THE CORRECT CROSS SECTIONS ARE BEING USED.

CS-50 *** ERROR *** SEARCH COMMAND NUMBER ____ IS UNABLE TO PERFORM A PITCH SEARCH BECAUSE THE DIMENSIONS OF REGION _____ OF UNIT _____ ARE NOT EXPLICITLY DEFINED.

This message from subroutine PCHSRH indicates that the specified search command is not valid for the specified region. An ARRAY or CORE region cannot be altered; nor can a REPLICATE or REFLECTOR region immediately following an ARRAY or CORE region.

CS-55 *** ERRORS WERE ENCOUNTERED IN PROCESSING THE CSAS-KENO5 DATA. EXECUTION IS IMPOSSIBLE. ***

This message from subroutine SASSY is printed if errors were found in the KENO V.a input data for CSAS. If a search is being made, data reading will continue until all the data have been entered or a fatal error terminates the data reading. When the data reading and checking have been completed, the problem will terminate without executing. Check the printout to locate the errors responsible for this message.

CS-62 *** ERROR *** MIXTURE _____ IN THE GEOMETRY WAS NOT CREATED IN THE STANDARD COMPOSITIONS SPECIFICATION DATA.

This message from subroutine MIXCHK indicates that a mixture specified in the KENO V.a geometry was not created in the standard composition data. If mixture 500 (a cell-weighted, homogenized mixture) is to be used in the KENO V.a geometry data, CSAS2X or CSAS4X must be specified to create it.

CS-65 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. C4DATA WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the material input processor and data preprocessor. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 207 is executed when this message is printed.

CS-66 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. SASSY WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the KENO V.a input processor. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 208 is executed when this message is printed.

CS-67 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. RDOPT WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the processor for the search data. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 209 is executed when this message is printed.

CS-68 *** ERROR *** AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE ___ DATA ENTERED FOR THIS PROBLEM.

This message from the main program, CSAS, is printed if the subroutine library routine LRDERR returns a value of "TRUE," indicating that a reading error has been encountered in the "KENO PARAMETER" data or the CSAS "SEARCH" data. The appropriate data type is printed in the message.

Locate the unnumbered message stating "***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****." Correct the data and resubmit the problem.

CS-69 ***ERROR*** MIXTURE ____ IS AN INAPPROPRIATE MIXTURE NUMBER FOR USE IN THE KENO GEOMETRY DATA BECAUSE IT IS A COMPONENT OF THE CELL-WEIGHTED MIXTURE CREATED BY XSDRNPM.

This message from subroutine CMXCHK indicates that a mixture that is a component of a cell-weighted mixture has been used in the KENO V.a geometry data.

CS-70 ***** ERROR ***** SEARCH OR OPTIMIZATION DATA MUST BE ENTERED FOR CSAS4, CSAS4X, AND CSAS3. NO SEARCH DATA WAS ENTERED.

This message from subroutine RDOPT is self-explanatory. If the user does not desire to run a search, another sequence such as CSAS25, CSAS2X, or CSAS1X should be chosen.

CS-71 *** ERROR *** _____ IS NOT A VALID SEARCH TYPE.

This message is from subroutine RDOPT. The allowed search types include OPTIMIZATION, DIMENSION, and CONCENTRATION. The first four characters of the search data after the words READ SEARCH must be OPTI, OPT, OPTM, DIM, DIME, DMSN, CON, or CONC. The data may be misspelled or out of order.

CS-72 *** ERROR *** THE SEARCH TYPE IS INVALID. I=_____.

This message is from subroutine RDOPT. The numerical index, I, should be 1 for an optimum pitch search, 2 for a dimension search, and 3 for a concentration search. If it is none of these, the search type has been incorrectly specified or a code error has been introduced.

CS-73 ***** AN END OF FILE WAS ENCOUNTERED BEFORE ALL THE SEARCH DATA WAS READ.

This self-explanatory message is from subroutine RDOPT. Check the input data to be sure nothing was omitted or misspelled.

CS-74 ***** AN END SEARCH FLAG WAS READ BEFORE ALL THE SEARCH DATA WAS READ.

This self-explanatory message is from subroutine RDOPT. Check the input data for omissions and correct order.

CS-75 *** ERROR *** READ SEARCH FLAG WAS NOT FOUND. _____ WAS READ INSTEAD.

This self-explanatory message is from subroutine RDOPT. READ SEARCH was expected but was not found. Check the input data for omissions and correct order.

CS-76 *** ERROR *** END SEARCH FLAG WAS NOT FOUND. _____ WAS READ INSTEAD.

This self-explanatory message from subroutine RDOPT indicates that an end of file was encountered when looking for READ SEARCH. Check the input data for omissions, correct order, and spelling.

CS-77 *** ERROR *** AN END OF FILE WAS FOUND WHEN THE READ SEARCH FLAG WAS EXPECTED.

This self-explanatory message is from subroutine RDOPT. Check the input data for omissions, correct order, and spelling.

CS-78 *** ERROR *** _____ IS NOT A VALID OPTIMIZATION SEARCH TYPE.

This message is from subroutines FIXTYP and OPTYPE. It indicates that an invalid search type was read. The valid search names include OPTIMUM PITCH, OPTIMUM CONCENTRATION, and OPTIMUM DIMENSION. Either the data was entered improperly or a code error has been introduced. A STOP 215 is executed when this message is printed.

CS-79 *** ERROR *** THE SEARCH TYPE IS INVALID. I=____.

This message from subroutines FIXTYP and OPTYPE indicates that the search type was misspelled or a code error has been introduced. A STOP 216 is executed when this message is printed.

CS-80 *** ERROR *** SEARCH DATA HAS BEEN DESTROYED. I=____ ICMND=____ IPNUM= ____ II=____ IGEOM=____

This message from subroutines CNCSRH or DIMSRH indicates that the search data cannot be interpreted. This is probably due to a code error.

CS-81 THE SEARCH WAS PREMATURELY TERMINATED BECAUSE ONLY ____ GENERATIONS WERE COMPLETED IN THE LAST KENO PASS. AT LEAST ____GENERATIONS (NSK=, IN THE KENO PARAMETER DATA) ARE REQUIRED TO SUSTAIN A SEARCH.

This self-explanatory message is from subroutine LODATA. Alter the data and/or the job control language to specify enough time and I/Os to allow completion of a sufficient number of generations in each pass. A STOP 212 is executed when this message is printed.

CS-82 *** AN ERROR WAS ENCOUNTERED IN ONE OF THE FUNCTIONAL MODULES.

This message from MODIFY, the main program of the search package, indicates that an error was encountered during execution of BONAMI, NITAWL, XSDRN, or KENO V.a. Check the printout to locate and correct the error.

CS-83 ***** NO FEASIBLE SOLUTION WAS FOUND IN THE SEARCH PACKAGE. EXECUTION IS TERMINATED. *****

This message from MODIFY, the main program of the search package, is self-explanatory. The code was unable to reach a feasible solution for the requested type of search.

CS-84 **** ERROR **** POINTER INDEX IS OUT OF RANGE. MODIFIED KENO V DATA CANNOT BE WRITTEN. NDX=____.

This message from subroutine RERITE is probably due to a code error. A STOP 213 is executed when this message is printed.

CS-85 *** ERROR *** ALL OF THE ROOTS FOR K=_____ LIE OUTSIDE THE PARAMETER CONSTRAINTS.

This self-explanatory message from subroutine LODATA indicates that a solution does not exist for the specified desired value of k-effective.

CS-86 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutines FIXTYP and OPTYP indicates that the allocated computer storage will not hold the information required for a search. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage necessary to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 211 is executed in conjunction with this message.

CS-87 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. LODATA WAS NOT CALLED.

This message from subroutine MODIFY indicates that the allocated computer storage is not large enough to load the search data. The message is accompanied by a STOP 220. Increase the requested computer storage in the job control language.

CS-88 *** NOT ENOUGH STORAGE TO LOAD THE KENO V MIXING TABLE.

This message from subroutine MXTBLE is printed if the allocated computer storage will not accommodate the KENO V.a mixing table data. Increase the requested computer storage in the job control language.

CS-89 *** ERROR *** _____ IS AN INVALID DIMENSION SEARCH COMMAND.

This message from subroutine DMSN indicates that the dimension search data are out of order, a search command is spelled incorrectly, or the search data are specified incorrectly.

CS-90 *** ERROR *** _____ IS AN INVALID SEARCH PARAMETER.

This message printed from subroutines FIXCNC, FIXDIM, FIXPCH, OPTCNC, OPTDMN, or OPTPCH indicates that a parameter entered in the search type specification data is not valid (see Sect. C4.4.9.1 and Table C4.4.15). The data could be misspelled or out of order. Omission of the keyword MORE before entering the individual search commands (Sect. C4.4.9.2) can cause this error.

CS-91 *** ERROR *** _____ IS AN INVALID CONCENTRATION SEARCH COMMAND.

This message from subroutine CNCN is caused by a misspelled or illegal search command when attempting to do a concentration search.

CS-92 *** ERROR *** ITY IS NOT PITCH(1), DIMENSION(2), OR CONCENTRATION(3). ITY IS PRINTED BELOW.

PERTINENT CONSTANTS

This message from subroutine LODATA indicates that the search data was incorrect when it was reloaded. This error is probably due to a code error.

CS-93 *** ERROR *** NOPT IS NOT MINIMUM (-1), CRITICAL(0), OR MAXIMUM(1). NOPT IS PRINTED BELOW.

PERTINENT CONSTANTS

This message from subroutine LODATA indicates that the search data were incorrect when they were reloaded. This error is probably due to a code error.

CS-94 *** ERROR *** REG=____ IS AN INVALID SEARCH DATA ENTRY. THE REGION NUMBER MUST BE GREATER THAN ZERO AND NO LARGER THAN THE NUMBER OF REGIONS IN THE UNIT.

This message from subroutine DMSN indicates that the region to be altered is incorrectly specified. For example, if the unit being altered contains five geometry regions, the value specified for REG= can be as small as 1 and as large as 5.

CS-95 *** ERROR *** AN ERROR WAS ENCOUNTERED IN THE SEARCH DATA. THE LAST REGION NUMBER MUST BE AT LEAST AS LARGE AS THE FIRST REGION NUMBER. CHECK THE SEARCH DATA PRINTED BELOW.
keyword UNIT ___ REGIONS ___ TO ___ PARAMETER=____
SEARCH CONSTANTS ARE _____ _____ _____ _____ _____ _____

This message from subroutine DMSN indicates that the search data specified an invalid region number for the final region to be altered. Check the printed data and correct as appropriate.

CS-96 *** ERROR *** AN ERROR WAS ENCOUNTERED IN THE SEARCH DATA. THE REGION NUMBERS MUST BE GREATER THAN ZERO AND NO LARGER THAN THE NUMBER OF REGIONS IN THE UNIT. CHECK THE SEARCH DATA PRINTED BELOW.
keyword UNIT ____ REGIONS ____ TO ____ PARAMETER=____
SEARCH CONSTANTS ARE _____ _____ _____ _____ _____ _____

This message from subroutine DMSN indicates that one of the specified region numbers is incorrect. Check the printed data and correct as appropriate.

CS-97 *** ERROR *** NO VALID SEARCH COMMANDS WERE FOUND IN THE DATA.

This message is accompanied by a STOP 235 and is printed from subroutines PCHSRH, CNCSRH, and DIMSRH. A common cause of this error is the omission of the MORE command before the individual search commands are entered. See Sects. C4.4.10.1 and C4.4.10.2.

CS-99 THIS PROBLEM WILL NOT BE RUN BECAUSE PARM=CHECK WAS ENTERED IN THE ANALYTICAL SEQUENCE SPECIFICATION.

This message from subroutine CSAS indicates that the problem data were read and checked and no errors were found. To execute the problem, remove the PARM=CHECK or PARM=CHK from the analytical sequence indicator data entry.

CS-100 THIS PROBLEM WILL NOT BE RUN BECAUSE ERRORS WERE ENCOUNTERED IN THE INPUT DATA.

This message from subroutine CSAS is self-explanatory. Examine the printout to locate the error or errors in the input data. Correct them and resubmit the problem.

CS-101 THE CONSTRAINTS ARE NOT VALID FOR PARAMETER SET. –CON=____ +CON=_____

This message is printed from either subroutine CNCN or DMSN if the parameter set is invalid. For a parameter set to be valid, –CON must be less than 0.0 and +CON must be greater than 0.0. They can be explicitly set or calculated using default or provided data.

CS-103 CANNOT MODIFY STANDARD COMPOSITION IN CELL-WEIGHTED MIXTURE. THE FOLLOWING COMPOSITIONS ARE IN THE CELL-WEIGHTED MIXTURE.

This message from subroutine CELCHK indicates that searches cannot be performed on compositions contained in the unit cell data used to make a cell-weighted mixture.

### C4.7.3. STOP CODES

The STOP codes that are utilized in CSAS3 and CSAS4 are listed in tabular form, indicating the subroutine where they occur and the associated error message. A STOP is executed when a fatal error is recognized. Locate the associated message number to determine the appropriate corrective measures. A traceback may be generated whenever subroutine SOTP is called to print a message. If no traceback is indicated in the STOP CODE table, a STOP is printed at the completion of the associated message.

| Stop No. | Subroutine | Traceback | Associated Message |
|---|---|---|---|
| 20 | | | See Sect. C4.7.5 |
| 200 | SHORTX | No | MP-64 |
| 201 | CLAPSE | No | MP-80 |
| 203 | MXTBLE | No | CS-88 |
| 207 | CSAS | Yes | CS-65 |
| 208 | CSAS | Yes | CS-66 |
| 209 | CSAS | Yes | CS-67 |
| 211 | OPTYP | No | CS-86 |
| | FIXTYP | No | CS-86 |
| 212 | LODATA | No | CS-81 |
| 213 | RERITE | No | CS-84 |
| 214 | CSAS | Yes | CS-68 |
| 215 | OPTYP | No | CS-78 |
| | FIXTYP | No | CS-78 |
| 216 | OPTYP | No | CS-79 |
| | FIXTYP | No | CS-79 |
| 220 | MODIFY | No | CS-87 |
| 225 | LODATA | Yes | CS-92 |
| 230 | LODATA | Yes | CS-93 |
| 235 | DIMSRH | Yes | CS-97 |
| | OPTSRH | Yes | CS-97 |
| | PCHSRH | Yes | CS-97 |
| 300 | KNIGHT | Yes | MP-1 |
| 310 | KNIGHT | Yes | MP-2 |
| 320 | KNIGHT | Yes | MP-3 |
| 330 | SETUPA | Yes | MP-4 |
| 340 | MTABLE | Yes | MP-23 |
| 350 | MIXTAB | Yes | MP-24 |
| 360 | MIXTAB | Yes | MP-25 |
| 370 | MIXTAB | Yes | MP-26 |
| 380 | REORDR | No | MP-27 |
| 385 | REORDR | No | MP-40 |
| 390 | SETUPB | Yes | MP-19 |
| 395 | IDENTS | No | MP-28 |
| | SLNS | No | MP-28 |
| 396 | CSPARM | No | MP-45 |

### C4.7.4 MESSAGES ASSOCIATED WITH STOP 20 IN CSAS

The following messages from Subroutine Library routines will result in a STOP 20 in the material information processor and CSAS:

LMP001 DA ERROR – INVALID UNIT NUMBER. THE LOGICAL UNIT NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that an invalid unit number was specified as a direct-access device. In CSAS this message is indicative of a code error.

LMP002 DA ERROR – FORTRAN USING THIS UNIT. THE LOGICAL UNIT NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that the specified unit number is open as a sequential dataset rather than a direct-access dataset. In CSAS, this error may be caused by entering a direct-access unit number for FRD= or FWR= in MORE DATA.

LMP003 DA ERROR – DCB NOT OPEN. THE LOGICAL UNIT NUMBER IS _____.

This message from the Subroutine Library direct-access routines indicates that the program attempted to read or write on a direct-access device but that the data control block was not open. In CSAS this message is indicative of a code error.

LMP004 DA ERROR – UNABLE TO OPEN DCB. THE LOGICAL UNIT NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that the program is unable to open the data control block for the direct-access device. This message indicates that the job control language did not include proper specification of the above-named unit.

LMP005 DA ERROR – RELATIVE BLOCK NOT IN DATA SET. RELATIVE BLOCK NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that the number of direct-access blocks is too small for the problem. Increase the number of direct-access blocks in MORE DATA by entering the parameter DAB=nnn where nnn is larger. For example, if the relative block number is 201, try increasing the number of direct-access blocks to 300 or more. For CSAS25, CSAS2X, CSAS4, and CSAS4X, it may also be necessary to set NB8=nnn in the KENO parameter data.

LMP006 DA ERROR – INVALID BLOCK LENGTH. THE BLOCK LENGTH IS _____.

This message from the Subroutine Library direct-access routines indicates that the length of the direct-access blocks is invalid. A valid block length must be positive. In CSAS this message is indicative of a code error.

LMP007 DA ERROR – DCB ALREADY OPEN. THE LOGICAL UNIT NUMBER IS _____.

This message from the Subroutine Library direct-access routines indicates that the data control block for the above-named unit was previously opened and not closed.

LMP008 DA ERROR – PERMANENT I/O ERROR.

This message from the Subroutine Library direct-access routines indicates that a permanent I/O error has occurred.

### C4.7.5 UNNUMBERED MESSAGES

Occasionally the code may terminate without printing a message that can be located via the message number. Some of these messages are:

***** END OF FILE READ ON UNIT n *****

This message is from subroutine ENFILE in the Subroutine Library. Unit n is the device from which data is being read. This message can be caused by lack of a double blank between the END for the last standard composition and the END COMP. It can also be caused by omitting the END COMP or misspelling COMP in the END COMP entry. Other causes include incomplete or missing data. A traceback accompanies this message. Correct the data and resubmit the problem.

***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****
card image is printed here
ON THE ABOVE CARD, CHARACTER NUMBER m (IMAGE=c) IS NOT VALID.

This message is from the Subroutine Library free-form reading routine DREAD. The character number, m, and the character image, c, printed in the message are recognized as being invalid for the type of data being read. For example, character data were encountered when numerical data were expected. This message can be caused by missing or incomplete data and by omitting double blanks after a labeled or unlabeled END. A traceback accompanies this message. Correct the data and resubmit the problem.

## C4.A ALPHABETICAL INDEX OF SUBROUTINES

This section provides a convenient alphabetical index of the subroutines and functions and common blocks used in CSAS.

Table C4.A.1 provides an alphabetical listing of the subroutines and functions that comprise CSAS. The first column lists the subroutine name, the second column lists the subroutine that calls it, and the third column contains a list of the subroutines it calls. KENO V.a subroutines utilized by CSAS are contained in brackets <>; Material Information Processor subroutines used by CSAS are contained in braces {}, and FORTRAN-supplied library routines are contained in parentheses ().

Table C4.A.2 provides an alphabetical listing of the subroutines and functions in the control module MODIFY. MODIFY allows the user to perform searches using KENO V.a.

Table C4.A.3 provides an alphabetical index of the common blocks used in CSAS.

Table C4.A.4 provides an alphabetical index of the common blocks used in MODIFY.

Table C4.A.1 Alphabetical index of CSAS routines

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {ADAN} | {DAN} | (ACOS)<br>(ASIN)<br>(COS)<br>(SIN)<br>(SQRT)<br>{GAUSS}<br>{GDAN} |
| <ADJUST> | <HOLCHK> | MOVE |
| (ACOS) | {ADAN}<br>{DAN}<br>{FINNER}<br>{GDAN}<br>{OUTER}<br>{SPHDAN} | |
| (ASIN) | {ADAN}<br>{GDAN}<br>{OUTER} | |
| (COS) | {ADAN}<br>{DAN}<br>{FINNER}<br>{GDAN}<br>{OUTER} | |
| (EXP) | {DANCOF}<br>{DANDAD}<br>{FINNER} | |
| (SIN) | {ADAN}<br>{FINNER}<br>{GDAN}<br>{OUTER} | |
| (SQRT) | {ADAN}<br>{BALL}<br>{DAN}<br>{FINNER<br>{GDAN}<br>{OUTER}<br>{POKE} | |
| (ASIN) | <VOLUME> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| (EXP) | {NFLUX} | |
| (SQRT) | {BONDA}<br><CORSIZ><br><CRMAX><br><CRMIN><br>{DANCOF}<br>{EPSIG}<br><HOLHOL><br>{NFLUX}<br><RDPLOT><br><SRMAX><br><SRMIN><br><VOLUME><br><XRMIN><br><XXLIM><br><XXMIN> | |
| {AGAUSS} | {DAN}<br>{SPHDAN} | |
| <ALBRD> | <DIFALB> | <ALBUSE> |
| <ALBUSE> | <ALBRD> | INQUIR<br>IO<br>RITE |
| ALLOWC | {MOREDT} | |
| ALOCAT | CSAS | |
| <ARALBA> | <JOMITY><br>SASSY | <PRTLBA> |
| <ARASIZ> | <CORSIZ> | <LSCAN> |
| <ARAYIN> | <DATAIN> | AREAD<br><BOX><br>CLEAR<br><HLFWRD><br>IREAD<br><RCHRS><br><RDBOX><br>STOP<br>YREAD |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {ARCHK} | {DAN} | |
| AREAD | <ARAYIN> | |
| | CNCN | |
| | CPARAM | |
| | {CSPARM} | |
| | <DATAIN> | |
| | DMSN | |
| | FIXCNC | |
| | FIXDIM | |
| | FIXPCH | |
| | FIXTYP | |
| | <KENOG> | |
| | {LATCEL} | |
| | <MIXIT> | |
| | {MOREDT} | |
| | {MULTRG} | |
| | OPTCNC | |
| | OPTDMN | |
| | OPTPCH | |
| | OPTYP | |
| | <RCHRS> | |
| | <RDBIAS> | |
| | RDOPT | |
| | <RDORGN> | |
| | <RDPLOT> | |
| | <RDREF> | |
| | <RDSTRT> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {STDCMP} | |
| {ARRAY3} | {MIXTAB} | |
| {AVGB} | {BOND} | |
| {AVGY} | {TWODYS} | MGCWRD |
| | | STOP |
| {AVG1} | {CLAPSE} | |
| {AVG2} | {TWODXS} | MGCWRD |
| {AXSDRN} | {C4DATA} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {BALL} | {DANDAD} | (SQRT)<br>{SPHDAN} |
| {BINFIL} | {ICE}<br>{XSDRN} | |
| {BINRCF} | {BONAMI}<br>{ICE}<br>{NITAWL}<br>{XSDRN} | |
| {BINRCX} | {BONAMI}<br>{ICE}<br>{NITAWL}<br>{XSDRN} | |
| BIRITE | CPARAM | |
| {BONAMI} | {C4DATA} | {BINRCF}<br>{BINRCX}<br>{LCHKMX} |
| {BOND} | {CLAPSE} | {AVGB}<br>{FOLD}<br>IO<br>{UNFOLD} |
| {BONDA} | {RESDA} | (SQRT) |
| <BOX> | <ARAYIN> | |
| <BOXC> | <LOADIT> | |
| {CALCMS} | {C4DATA} | {LCHKMX} |
| CELLMX | {C4DATA} | {LCHKMX} |
| {CINIT} | {C4DATA} | CLEAR<br>INQUIR<br>OPENDA<br>RITE |
| {CLAPSE} | {C4DATA} | {AVG1}<br>{BOND}<br>CLEAR<br>COPY |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | {DEAN} |
| | | IO |
| | | IO4 |
| | | STOP |
| | | {TWODXS} |
| | | {TWODYS} |
| CLEAR | <ARAYIN> | |
| | {CINIT} | |
| | {CLAPSE} | |
| | CNCN | |
| | CPARAM | |
| | CSAS | |
| | {C4DATA} | |
| | <DATAIN> | |
| | DMSN | |
| | <GEOMIN> | |
| | <HOLE> | |
| | <HUNTER> | |
| | <LODARA> | |
| | <LODWTS> | |
| | <MESH> | |
| | {NITAWL} | |
| | <PRTPLT> | |
| | <RDALB> | |
| | <RDICE> | |
| | RDOPT | |
| | <RDPLOT> | |
| | <RDSTRT> | |
| | <RELATE> | |
| | {RESDA} | |
| | <RGUSED> | |
| | <RTARA> | |
| | <SAVST6> | |
| | {SETUPB} | |
| | <SORTA> | |
| | {TWODXS} | |
| | {TWODYS} | |
| | <UNTCRS> | |
| | <VOLUME> | |
| | {XSDRN} | |
| CLOSDA | CSAS | |
| | {KNIGHT} | |
| | {SHORTX} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| CMXCHK | SASSY | {LCHKMX} |
| CNCN | FIXTYP<br>OPTYP | AREAD<br>CLEAR<br>FREAD<br>INQUIR<br>IREAD<br>RITE |
| CNCSRH | FIXTYP<br>OPTYP | PRTSRH<br>RITE<br>STOP |
| COPY | {CLAPSE}<br>{SHORTX}<br>{TWODXS}<br>{TWODYS} | |
| <CORSIZ> | <JOMITY> | (SQRT)<br><ARASIZ> |
| CPARAM | CSAS | AREAD<br>BIRITE<br>CLEAR<br>FREAD<br>IREAD<br>LIBEQL<br>OPNFIL<br>RNDIN<br>RNDOUT<br>TIMFAC<br>ZREAD |
| CREAD | <RDREF> | |
| <CRMAX> | <HOLEXT><br><JOMCHK> | (SQRT)<br><CRSPRD><br><DOTPRD><br>STOP<br><VECADD><br><VECNRM> |
| <CRMIN> | <HOLHOL> | (SQRT)<br><CRSPRD><br><DOTPRD><br>STOP |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <VECADD> |
| | | <VECNRM> |
| <CRSPRD> | <CRMAX> | |
| | <CRMIN> | |
| | <SRMAX> | |
| | <SRMIN> | |
| CSAS | | ALOCAT |
| | | CLEAR |
| | | CLOSDA |
| | | CPARAM |
| | | {C4DATA} |
| | | GETMS |
| | | INDEX |
| | | IOLEFT |
| | | <IOSDUN> |
| | | OPNFIL |
| | | PARTBL |
| | | RDOPT |
| | | RESETM |
| | | SASSY |
| | | SCANON |
| | | STOP |
| {CSPARM} | {SETUPA} | AREAD |
| | | LCOMPR |
| | | {SMMTBA} |
| {C4DATA} | CSAS | {AXSDRN} |
| | | {BONAMI} |
| | | {CALCMS} |
| | | CELLMX |
| | | {CINIT} |
| | | {CLAPSE} |
| | | CLEAR |
| | | {EPSIG} |
| | | {ICE} |
| | | {KNIGHT} |
| | | {MIPCHK} |
| | | {NITAWL} |
| | | {PRELPS} |
| | | {SHORTX} |
| | | {SMMTBA} |
| | | STOP |
| | | {VERIFY} |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | {XSDRN} |
| | | {XSMESH} |
| {DAN} | {DANDAD} | {ADAN} |
| | | (ACOS) |
| | | (COS) |
| | | (SQRT) |
| | | {AGAUSS} |
| | | {ARCHK} |
| | | ERRSET |
| {DANCOF} | {RESDA} | (EXP) |
| | | (SQRT) |
| | | {DANDAD} |
| | | E3 |
| | | SHELC |
| | | STOP |
| {DANDAD} | {DANCOF} | (EXP) |
| | | {BALL} |
| | | {DAN} |
| | | {SDAN} |
| <DATAIN> | SASSY | <ARAYIN> |
| | | AREAD |
| | | CLEAR |
| | | <EXTRA> |
| | | <FLDATA> |
| | | <GEOMIN> |
| | | <IDX1D> |
| | | IOWRT |
| | | <MIXIT> |
| | | <RDBIAS> |
| | | <RDPLOT> |
| | | <RDREF> |
| | | <RDRST> |
| | | <RDSTRT> |
| | | <RT> |
| | | <RTARA> |
| | | <SAVST6> |
| | | STOP |
| | | <WRTPLT> |
| {DEAN} | {CLAPSE} | {KAUL} |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <DIFALB> | <FLDATA> | <ALBRD> |
| DIMSRH | FIXTYP<br>OPTYP | PRTSRH<br>RITE<br>STOP |
| DMSN | FIXTYP<br>OPTYP | AREAD<br>CLEAR<br>FREAD<br>INQUIR<br>IREAD<br>RITE |
| <DOTPRD> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN><br><XRMIN> | |
| DTASET | {SHORTX} | |
| {EPSIG} | {C4DATA} | (SQRT) |
| ERRSET | {DAN} | |
| <EXTRA> | <DATAIN> | |
| E3 | {DANCOF}<br>{SDAN} | |
| {FAX} | {STDCMP} | FREAD<br>IREAD |
| {FAXPRT} | {STDCMP} | |
| <FINDBX> | <LOCATE> | |
| {FINNER} | {OUTER} | (ACOS)<br>(COS)<br>(EXP)<br>(SIN)<br>(SQRT) |
| {FIX} | {STDCMP} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| FIXCNC | FIXTYP | AREAD<br>FREAD<br>IREAD |
| FIXDIM | FIXTYP | AREAD<br>FREAD<br>IREAD |
| FIXPCH | FIXTYP | AREAD<br>FREAD<br>IREAD<br>REED |
| FIXTYP | RDOPT | AREAD<br>CNCN<br>CNCSRH<br>DIMSRH<br>DMSN<br>FIXCNC<br>FIXDIM<br>FIXPCH<br>PCHSRH<br>REED<br>STOP |
| <FLDATA> | <DATAIN> | <DIFALB><br>REED<br><RGUSED><br>RITE<br><SORTA><br><SORTR><br>STOP<br><WATES> |
| {FOLD} | {BOND} | |
| {FOX} | {STDCMP} | |
| FREAD | CNCN<br>CPARAM<br>DMSN<br>{FAX}<br>FIXCNC<br>FIXDIM<br>FIXPCH<br>{LATCEL} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <MIXIT> | |
| | {MOREDT} | |
| | {MULTRG} | |
| | OPTCNC | |
| | OPTDMN | |
| | OPTPCH | |
| | <RDBIAS> | |
| | <RDORGN> | |
| | <RDPLOT> | |
| | <RDSTRT> | |
| | {SETUPA} | |
| | {STDCMP} | |
| | <XXIN> | |
| {GAUSS} | {ADAN} | |
| | {OUTER} | |
| {GDAN} | {ADAN} | (ACOS) |
| | | (ASIN) |
| | | (COS) |
| | | (SIN) |
| | | (SQRT) |
| | | KI3 |
| <GEOMIN> | <DATAIN> | CLEAR |
| | | <KENOG> |
| | | <READGM> |
| | | STOP |
| GETMS | CSAS | |
| GETPTR | <RCHRS> | |
| {GFLUX} | {PRELPS} | |
| <GOCURS> | <VOLUME> | <HUNTER> |
| <GTVOLS> | <VOLUME> | |
| <HLFWRD> | <ARAYIN> | |
| {HOGGIE} | {TWODYS} | STOP |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <HOLCHK> | <JOMCHK> | <ADJUST><br><HOLEXT><br><HOLHOL> |
| <HOLE> | <LODARA> | CLEAR<br>STOP |
| <HOLEXT> | <HOLCHK> | <CRMAX><br><SRMAX><br><XXLIM> |
| <HOLHOL> | <HOLCHK> | (SQRT)<br><CRMIN><br><SRMIN><br><XRMIN><br><XXMIN> |
| <HUNTER> | <GOCURS><br><VOLUME> | CLEAR<br>MOVE |
| {ICE} | {C4DATA} | {BINFIL}<br>{BINRCF}<br>{BINRCX}<br>{VOLFRC} |
| {IDENTS} | {MTABLE} | {ISCAN}<br>{NOFRAC} |
| <IDX1D> | <DATAIN> | IREAD |
| INDEX | CSAS<br><RTADJ> | |
| INQUIR | <ALBUSE><br>{CINIT}<br>CNCN<br>DMSN<br><MIXIT><br>MXTBLE<br><PRTPLT><br><RDALB><br><RDARA><br><RDICE><br>RDOPT<br><RDRST><br><RDWTS> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <RT> | |
| | <RTARA> | |
| | <SAVST6> | |
| | <WAITIN> | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTICE> | |
| | <WRTPLT> | |
| | <WRTRST> | |
| IO | <ALBUSE> | |
| | {BOND} | |
| | {CLAPSE} | |
| | <RDALB> | |
| | <RDARA> | |
| | <RDBIAS> | |
| | <RDICE> | |
| | <RDPLOT> | |
| | <RDRST> | |
| | <RDSTRT> | |
| | <RDWTS> | |
| | <SAVST6> | |
| | {TWODXS} | |
| | {TWODYS} | |
| | <WAITIN> | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTARA> | |
| | <WRTICE> | |
| | <WRTPLT> | |
| | <WRTRST> | |
| | <WRTWTS> | |
| IOLEFT | CSAS | |
| | <IOSDUN> | |
| <IOSDUN> | CSAS | IOLEFT |
| | <JOMITY> | |
| | SASSY | |
| IOWRT | <DATAIN> | |
| | {SHORTX} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| IO4 | {CLAPSE} | |
| | {TWODXS} | |
| | {TWODYS} | |
| IREAD | <ARAYIN> | |
| | CNCN | |
| | CPARAM | |
| | DMSN | |
| | {FAX} | |
| | FIXCNC | |
| | FIXDIM | |
| | FIXPCH | |
| | <IDX1D> | |
| | <KENOG> | |
| | {LATCEL} | |
| | <MIXIT> | |
| | {MOREDT} | |
| | {MULTRG} | |
| | OPTCNC | |
| | OPTDMN | |
| | OPTPCH | |
| | <RDBIAS> | |
| | <RDBOX> | |
| | <RDPLOT> | |
| | <RDSTRT> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {STDCMP} | |
| | {XSMESH} | |
| | <XXIN> | |
| {ISCAN} | {IDENTS} | |
| | {NITAWL} | |
| | {REORDR} | |
| | {SLNS} | |
| | {STDCMP} | |
| <JOMCHK> | <JOMITY> | <CRMAX> |
| | | <HOLCHK> |
| | | <SRMAX> |
| | | <XXLIM> |
| <JOMITY> | SASSY | <ARALBA> |
| | | <CORSIZ> |
| | | <IOSDUN> |
| | | <JOMCHK> |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <LOADIT> |
| | | <PRTJOM> |
| | | <VOLUME> |
| {KAUL} | {DEAN} | |
| <KENOG> | <GEOMIN> | AREAD |
| | | IREAD |
| | | <RCHRS> |
| | | <RDORGN> |
| | | <XXIN> |
| | | XXINA |
| KI3 | {GDAN} | |
| {KNIGHT} | {C4DATA} | CLOSDA |
| | | MESAGE |
| | | OPENDA |
| | | REED |
| | | {SETUPA} |
| | | STOP |
| {LATCEL} | {SETUPB} | AREAD |
| | | FREAD |
| | | IREAD |
| | | LCOMPR |
| | | {MIXERR} |
| | | SCANOF |
| | | SCANON |
| {LCHKMX} | {BONAMI} | |
| | {CALCMS} | |
| | CELLMX | |
| | CMXCHK | |
| | MIXCHK | |
| | {XSDRN} | |
| LCOMPR | {CSPARM} | |
| | {LATCEL} | |
| | {MULTRG} | |
| | {RESDA} | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {SMMTBA} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| LEN | <RCHRS> | |
| LIBEQL | CPARAM | |
| <LOADIT> | <JOMITY> | <BOXC><br><LODALB><br><LODARA><br><PRTARA><br>REED |
| <LOCATE> | <MESH> | <FINDBX><br><LOCBOX><br>MOVE<br><POSIT> |
| <LOCBOX> | <LOCATE><br><LODARA><br><SORTA> | |
| <LODALB> | <LOADIT> | REED |
| <LODARA> | <LOADIT> | CLEAR<br><HOLE><br><LOCBOX><br>REED |
| <LODRGC> | <RGUSED> | |
| <LODWTS> | SASSY | CLEAR<br><PRTWTS><br>RD<br>REED |
| {LOGID} | {SHORTX} | |
| LREAD | <MIXIT><br>{MULTRG}<br>{SETUPA}<br>{STDCMP} | |
| <LSCAN> | <ARASIZ><br><RGUSED> | |
| MESAGE | {KNIGHT} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <MESH> | <PRINT> | CLEAR<br><LOCATE> |
| MGCWRD | {AVGY}<br>{AVG2} | |
| {MIPCHK} | {C4DATA} | WOT8 |
| MIXCHK | MXTBLE | {LCHKMX} |
| {MIXERR} | {LATCEL}<br>{MOREDT}<br>{MULTRG} | |
| <MIXIT> | <DATAIN> | AREAD<br>FREAD<br>INQUIR<br>IREAD<br>LREAD<br><RDMIXT><br>RITE<br>STOP |
| {MIXTAB} | {SETUPB} | {ARRAY3}<br>{MODIFY}<br>{MTABLE}<br>{REORDR}<br>STOP |
| {MODIFY} | {MIXTAB} | |
| {MOREDT} | {SETUPB} | ALLOWC<br>AREAD<br>FREAD<br>IREAD<br>{MIXERR}<br>RESETC<br>SCANOF<br>SCANON |
| MOVE | <ADJUST><br><HUNTER><br><LOCATE><br>SASSY<br><SAVST6><br>{TWODYS} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <VOLUME><br><WATES> | |
| {MTABLE} | {MIXTAB} | {IDENTS}<br>STOP |
| {MULTRG} | {SETUPB} | AREAD<br>FREAD<br>IREAD<br>LCOMPR<br>LREAD<br>{MIXERR}<br>SCANOF<br>SCANON |
| MXTBLE | SASSY | INQUIR<br>MIXCHK<br>RITE<br>STOP<br>TABLE |
| {NFLUX} | {PRELPS} | (EXP)<br>(SQRT) |
| {NITAWL} | {C4DATA} | {BINRCF}<br>{BINRCX}<br>CLEAR<br>{ISCAN} |
| {NOFRAC} | {IDENTS} | {WTINV} |
| OPENDA | {CINIT}<br>{KNIGHT}<br>{SHORTX} | |
| OPNFIL | CPARAM<br>CSAS<br><RDSTRT><br>{SHORTX} | |
| OPTCNC | OPTYP | AREAD<br>FREAD<br>IREAD |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| OPTDMN | OPTYP | AREAD<br>FREAD<br>IREAD |
| OPTPCH | OPTYP | AREAD<br>FREAD<br>IREAD<br>REED |
| OPTYP | RDOPT | AREAD<br>CNCN<br>CNCSRH<br>DIMSRH<br>DMSN<br>OPTCNC<br>OPTDMN<br>OPTPCH<br>PCHSRH<br>REED<br>STOP |
| {OUTER} | {SPHDAN} | (ACOS)<br>(ASIN)<br>(COS)<br>(SIN)<br>(SQRT)<br>{FINNER}<br>{GAUSS} |
| PARTBL | CSAS | RNDOUT |
| {PATCH} | {TWODXS}<br>{TWODYS} | |
| PCHSRH | FIXTYP<br>OPTYP | PRTSRH<br>RITE<br>STOP<br><XXLIM> |
| <POINT> | SASSY | |
| {POKE} | {SETUPB} | (SQRT) |
| <POSIT> | <LOCATE> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| PRATBL | SASSY | <RTADJ> |
| {PRELPS} | {C4DATA} | {GFLUX}<br>{NFLUX} |
| <PRINT> | <PRTPLT> | <MESH> |
| <PRTARA> | <LOADIT><br>SASSY | |
| <PRTJOM> | <JOMITY><br>SASSY | |
| <PRTLBA> | <ARALBA> | |
| <PRTPLT> | SASSY | CLEAR<br>INQUIR<br><PRINT><br>REED<br><RELATE><br><UNTCRS> |
| PRTSRH | CNCSRH<br>DIMSRH<br>PCHSRH | |
| <PRTWTS> | <LODWTS> | |
| <RCHRS> | <ARAYIN><br><KENOG><br><RDPLOT> | AREAD<br>GETPTR<br>LEN<br>RCRDLN<br>RSTPTR<br>STOP |
| RCRDLN | <RCHRS> | |
| RD | <LODWTS> | |
| <RDALB> | <RDRST> | CLEAR<br>INQUIR<br>IO<br>RITE |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <RDARA> | <RDRST> | INQUIR<br>IO<br>RITE |
| <RDBIAS> | <DATAIN> | AREAD<br>FREAD<br>IO<br>IREAD<br><WAITIN> |
| <RDBOX> | <ARAYIN> | IREAD |
| <RDICE> | <RDRST> | CLEAR<br>INQUIR<br>IO<br>RITE |
| <RDMIXT> | <MIXIT> | |
| RDOPT | CSAS | AREAD<br>CLEAR<br>FIXTYP<br>INQUIR<br>OPTYP<br>RITE |
| <RDORGN> | <KENOG> | AREAD<br>FREAD |
| <RDPLOT> | <DATAIN> | (SQRT)<br>AREAD<br>CLEAR<br>FREAD<br>IO<br>IREAD<br><RCHRS> |
| <RDREF> | <DATAIN> | AREAD<br>CREAD |
| <RDRST> | <DATAIN> | INQUIR<br>IO<br><RDALB><br><RDARA><br><RDICE> |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <RDWTS> |
| | | RITE |
| <RDSTRT> | <DATAIN> | AREAD |
| | | CLEAR |
| | | FREAD |
| | | IO |
| | | IREAD |
| | | OPNFIL |
| <RDWTS> | <RDRST> | INQUIR |
| | | IO |
| | | RITE |
| <READGM> | <GEOMIN> | |
| REED | FIXPCH | |
| | FIXTYP | |
| | <FLDATA> | |
| | {KNIGHT} | |
| | <LOADIT> | |
| | <LODALB> | |
| | <LODARA> | |
| | <LODWTS> | |
| | OPTPCH | |
| | OPTYP | |
| | <PRTPLT> | |
| | {SHORTX} | |
| | <SORTA> | |
| | {STDCMP} | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTARA> | |
| | <WRTICE> | |
| | <WRTRST> | |
| | <WRTWTS> | |
| <RELATE> | <PRTPLT> | CLEAR |
| {REORDR} | {MIXTAB} | {ISCAN} |
| {RESDA} | {SETUPB} | {BONDA} |
| | | CLEAR |
| | | {DANCOF} |
| | | LCOMPR |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| RESETC | {MOREDT} | |
| RESETM | CSAS | |
| <RGUSED> | <FLDATA> | CLEAR<br><LODRGC><br><LSCAN> |
| RITE | <ALBUSE><br>{CINIT}<br>CNCN<br>CNCSRH<br>DIMSRH<br>DMSN<br><FLDATA><br><MIXIT><br>MXTBLE<br>PCHSRH<br><RDALB><br><RDARA><br><RDICE><br>RDOPT<br><RDRST><br><RDWTS><br><RT><br><RTARA><br><SAVST6><br><WAITIN><br><WATES><br><WRTPLT> | |
| RNDIN | CPARAM | |
| RNDOUT | CPARAM<br>PARTBL<br><WRTRST> | |
| RSTPTR | <RCHRS> | |
| <RT> | <DATAIN> | INQUIR<br>RITE |
| <RTADJ> | PRATBL | INDEX |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <RTARA> | <DATAIN> | CLEAR |
| | | INQUIR |
| | | RITE |
| SASSY | CSAS | <ARALBA> |
| | | CMXCHK |
| | | <DATAIN> |
| | | <IOSDUN> |
| | | <JOMITY> |
| | | <LODWTS> |
| | | MOVE |
| | | MXTBLE |
| | | <POINT> |
| | | PRATBL |
| | | <PRTARA> |
| | | <PRTJOM> |
| | | <PRTPLT> |
| | | STOP |
| | | <WRTRST> |
| <SAVST6> | <DATAIN> | CLEAR |
| | | INQUIR |
| | | IO |
| | | MOVE |
| | | RITE |
| | | STOP |
| SCANOF | {LATCEL} | |
| | {MOREDT} | |
| | {MULTRG} | |
| | {STDCMP} | |
| SCANON | CSAS | |
| | {LATCEL} | |
| | {MOREDT} | |
| | {MULTRG} | |
| | {SETUPA} | |
| {SDAN} | {DANDAD} | E3 |
| {SETUPA} | {KNIGHT} | AREAD |
| | | {CSPARM} |
| | | FREAD |
| | | IREAD |
| | | LCOMPR |
| | | LREAD |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | SCANON<br>{SETUPB}<br>STOP |
| {SETUPB} | {SETUPA} | AREAD<br>CLEAR<br>IREAD<br>{LATCEL}<br>LCOMPR<br>{MIXTAB}<br>{MOREDT}<br>{MULTRG}<br>{POKE}<br>{RESDA}<br>{STDCMP}<br>STOP |
| SHELC | {DANCOF} | |
| {SHORTX} | {C4DATA} | CLOSDA<br>COPY<br>DTASET<br>IOWRT<br>{LOGID}<br>OPENDA<br>OPNFIL<br>REED |
| {SLNAME} | {STDCMP} | |
| {SLNS} | {STDCMP} | {ISCAN} |
| {SMMTBA} | {CSPARM}<br>{C4DATA} | LCOMPR |
| <SORTA> | <FLDATA> | CLEAR<br><HOLE><br><LOCBOX><br>REED<br>STOP |
| <SORTR> | <FLDATA> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {SPHDAN} | {BALL} | (ACOS) |
| | | {AGAUSS} |
| | | {OUTER} |
| <SRMAX> | <HOLEXT> | (SQRT) |
| | <JOMCHK> | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |
| | | <VECADD> |
| | | <VECNRM> |
| <SRMIN> | <HOLHOL> | (SQRT) |
| | | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |
| | | <VECADD> |
| | | <VECNRM> |
| {STDCMP} | {SETUPB} | AREAD |
| | | {FAX} |
| | | {FAXPRT} |
| | | {FIX} |
| | | {FOX} |
| | | FREAD |
| | | IREAD |
| | | {ISCAN} |
| | | LREAD |
| | | REED |
| | | SCANOF |
| | | {SLNAME} |
| | | {SLNS} |
| STOP | <ARAYIN> | |
| | {AVGY} | |
| | {CLAPSE} | |
| | CNCSRH | |
| | <CRMAX> | |
| | <CRMIN> | |
| | CSAS | |
| | {C4DATA} | |
| | {DANCOF} | |
| | <DATAIN> | |
| | DIMSRH | |
| | FIXTYP | |
| | <FLDATA> | |
| | <GEOMIN> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | {HOGGIE} | |
| | <HOLE> | |
| | {KNIGHT} | |
| | <MIXIT> | |
| | {MIXTAB} | |
| | {MTABLE} | |
| | MXTBLE | |
| | OPTYP | |
| | PCHSRH | |
| | <RCHRS> | |
| | SASSY | |
| | <SAVST6> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | <SORTA> | |
| | <SRMAX> | |
| | <SRMIN> | |
| | {TWODYS} | |
| | <WATES> | |
| | {XSDRN} | |
| | {XSMESH} | |
| TABLE | MXTBLE | |
| TIMFAC | CPARAM | |
| {TWODXS} | {CLAPSE} | {AVG2} |
| | | CLEAR |
| | | COPY |
| | | IO |
| | | IO4 |
| | | {PATCH} |
| {TWODYS} | {CLAPSE} | {AVGY} |
| | | CLEAR |
| | | COPY |
| | | {HOGGIE} |
| | | IO |
| | | IO4 |
| | | MOVE |
| | | {PATCH} |
| | | STOP |
| {UNFOLD} | {BOND} | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <UNTCRS> | <PRTPLT> | CLEAR |
| <VECADD> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN><br><XRMIN> | |
| <VECNRM> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN> | |
| {VERIFY} | {C4DATA} | {VOLJB} |
| {VOLFRC} | {ICE} | |
| {VOLJB} | {VERIFY} | |
| <VOLUME> | <JOMITY> | (ASIN)<br>(SQRT)<br>CLEAR<br><GOCURS><br><GTVOLS><br><HUNTER> |
| MOVE | | |
| <WAITIN> | <RDBIAS> | INQUIR<br>IO<br>RITE |
| <WATES> | <FLDATA> | INQUIR<br>IO<br>MOVE<br>REED<br>RITE<br>STOP |
| WOT8 | {MIPCHK} | |
| <WRTALB> | <WRTRST> | INQUIR<br>IO<br>REED |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <WRTARA> | <WRTRST> | IO<br>REED |
| <WRTICE> | <WRTRST> | INQUIR<br>IO<br>REED |
| <WRTPLT> | <DATAIN> | INQUIR<br>IO<br>RITE |
| <WRTRST> | SASSY | INQUIR<br>IO<br>REED<br>RNDOUT<br><WRTALB><br><WRTARA><br><WRTICE><br><WRTWTS> |
| <WRTWTS> | <WRTRST> | IO<br>REED |
| {WTINV} | {NOFRAC} | |
| <XRMIN> | <HOLHOL> | (SQRT)<br><DOTPRD><br><VECADD> |
| {XSDRN} | {C4DATA} | {BINFIL}<br>{BINRCF}<br>{BINRCX}<br>CLEAR<br>{LCHKMX}<br>STOP |
| {XSMESH} | {C4DATA} | IREAD<br>STOP |
| <XXIN> | <KENOG> | FREAD<br>IREAD |
| XXINA | <KENOG> | |

Table C4.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <XXLIM> | <HOLEXT><br><JOMCHK><br>PCHSRH | (SQRT) |
| <XXMIN> | <HOLHOL> | (SQRT) |
| YREAD | <ARAYIN> | |
| ZREAD | CPARAM | |

Table C4.A.2 Alphabetical index of MODIFY routines

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| (ACOS) | FIXEDK<br>RCUBIC | |
| ALOCAT | MODIFY | |
| ALTER | LODATA | CLEAR<br>XXLIM |
| AVGKEF | LODATA | (SQRT) |
| CLEAR | ALTER | |
| CLOSDA | MODIFY | |
| (COS) | FIXEDK<br>RCUBIC | |
| CUBFIT | FIXEDK<br>MAXMIN<br>OPTMIZ | DECOMP<br>SOLVE |
| DECOMP | CUBFIT | |
| DENSTY | LODATA | |
| (DSQRT) | OPTMIZ | |
| FIXEDK | OPTMIZ | (ACOS)<br>(COS)<br>(SQRT)<br>CUBFIT |
| GETMS | MODIFY | |
| INQUIR | WRTALB | |
| IO | RERITE<br>WRTALB<br>WRTARA<br>WRTWTS | |
| IOLEFT | MODIFY | |
| ISCAN | MAXMIN | |

Table C4.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| LISTQA | MODIFY | |
| LODATA | LODCOM | ALTER<br>AVGKEF<br>DENSTY<br>MOVE<br>OPTMIZ<br>PRTJOM<br>PRTMIX<br>PRTSRH<br>REED<br>RERITE<br>RESTRT<br>RITE<br>STOP |
| LODCOM | MODIFY | LODATA<br>MOVE |
| MAXMIN | OPTMIZ | (SQRT)<br>CUBFIT<br>ISCAN |
| MODBLK | | |
| MODIFY | | ALOCAT<br>CLOSDA<br>GETMS<br>IOLEFT<br>LISTQA<br>LODCOM<br>OPENDA<br>OPNFIL<br>REED<br>RESETM<br>RITE<br>STOP |
| MOVE | LODATA<br>LODCOM | |
| OPENDA | MODIFY | |
| OPNFIL | MODIFY | |

Table C4.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| OPTMIZ | LODATA | (SQRT)<br>CUBFIT<br>DSQRT<br>FIXEDK<br>MAXMIN<br>RCUBIC |
| PRTJOM | LODATA | |
| PRTMIX | LODATA | |
| PRTSRH | LODATA | |
| RCUBIC | OPTMIZ | (ACOS)<br>(COS)<br>(SQRT) |
| REED | LODATA<br>MODIFY<br>RERITE<br>WRTALB<br>WRTARA<br>WRTWTS | |
| RERITE | LODATA | IO<br>REED<br>WRTALB<br>WRTARA<br>WRTWTS |
| RESETM | MODIFY | |
| RESTRT | LODATA | |
| RITE | LODATA<br>MODIFY | |
| SOLVE | CUBFIT | |
| (SQRT) | AVGKEF<br>FIXEDK<br>MAXMIN<br>OPTMIZ<br>RCUBIC<br>XXLIM | |

Table C4.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| STOP | LODATA<br>MODIFY | |
| WRTALB | RERITE | INQUIR<br>IO<br>REED |
| WRTARA | RERITE | IO<br>REED |
| WRTWTS | RERITE | IO<br>REED |
| XXLIM | ALTER | (SQRT) |

Table C4.A.3 Commons used in CSAS

| COMMON NAME | SUBROUTINE |
|---|---|
| ALBDAT | ALBRD |
| | ALBUSE |
| | CPARAM |
| | DATAIN |
| | DIFALB |
| | FLDATA |
| | LOADIT |
| | PARTBL |
| | POINT |
| | PRATBL |
| | RDALB |
| | RDREF |
| | RDRST |
| | SASSY |
| | WRTALB |
| | WRTRST |
| ALBNAM | ALBRD |
| | ALBUSE |
| | CPARAM |
| | DATAIN |
| | DIFALB |
| | FLDATA |
| | LOADIT |
| | PARTBL |
| | POINT |
| | PRATBL |
| | RDALB |
| | RDREF |
| | RDRST |
| | SASSY |
| | WRTALB |
| | WRTRST |
| BLKLNC | CINIT |
| | CPARAM |
| | MOREDT |
| | PARTBL |
| | SETUPB |
| | STDCMP |
| CCDATA | CSAS |
| | C4DATA |
| | SASSY |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| CDATA | CSAS |
| | C4DATA |
| | SASSY |
| CELMIX | CELLMX |
| | CMXCHK |
| CLAD | DANDAD |
| | GDAN |
| | SDAN |
| COMPOZ | KNIGHT |
| | MTABLE |
| CONMOD | KNIGHT |
| CUNIT | ARCHK |
| | CSAS |
| | CSPARM |
| | FAXPRT |
| | IDENTS |
| | KNIGHT |
| | LATCEL |
| | MIXTAB |
| | MOREDT |
| | MTABLE |
| | MULTRG |
| | NOFRAC |
| | POKE |
| | REORDR |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | SLNAME |
| | SLNS |
| | STDCMP |
| DEAL | C4DATA |
| | DANCOF |
| | LATCEL |
| | MULTRG |
| | POKE |
| | RESDA |
| | SETUPB |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| DIMEN | ARASIZ |
| | ARAYIN |
| | BOX |
| | BOXC |
| | CNCN |
| | CORSIZ |
| | CPARAM |
| | C4DATA |
| | DATAIN |
| | DMSN |
| | FINDBX |
| | FIXTYP |
| | FLDATA |
| | GEOMIN |
| | HOLCHK |
| | JOMCHK |
| | JOMITY |
| | KENOG |
| | LOCATE |
| | LOCBOX |
| | LODARA |
| | MESH |
| | MIXIT |
| | MXTBLE |
| | OPTYP |
| | PARTBL |
| | POINT |
| | POSIT |
| | PRATBL |
| | PRINT |
| | PRTJOM |
| | PRTPLT |
| | RDBIAS |
| | RDBOX |
| | RDICE |
| | RDREF |
| | RDRST |
| | RDSTRT |
| | READGM |
| | RGUSED |
| | RTARA |
| | SASSY |
| | SORTA |
| | SORTR |
| | VOLUME |
| | WRTICE |
| | WRTRST |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| DRTACS | ALBUSE |
| | CINIT |
| | CNCN |
| | CNCSRH |
| | CSAS |
| | DATAIN |
| | DIMSRH |
| | DMSN |
| | FIXPCH |
| | FIXTYP |
| | FLDATA |
| | LOADIT |
| | LODARA |
| | LODWTS |
| | MIXIT |
| | MXTBLE |
| | OPTPCH |
| | OPTYP |
| | PCHSRH |
| | PRTPLT |
| | RDALB |
| | RDARA |
| | RDICE |
| | RDOPT |
| | RDRST |
| | RDWTS |
| | RT |
| | RTARA |
| | SASSY |
| | SAVST6 |
| | SHORTX |
| | SORTA |
| | WAITIN |
| | WATES |
| | WRTALB |
| | WRTICE |
| | WRTPLT |
| | WRTRST |
| | WRTWTS |
| ERRFLG | ARAYIN |
| | PRTLBA |
| | SASSY |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| FLG | ARCHK |
| | CNCSRH |
| | CSAS |
| | C4DATA |
| | DANCOF |
| | DIMSRH |
| | FAX |
| | FIXCNC |
| | FIXDIM |
| | FIXPCH |
| | FIXTYP |
| | LATCEL |
| | MOREDT |
| | MULTRG |
| | OPTCNC |
| | OPTDMN |
| | OPTPCH |
| | OPTYP |
| | PCHSRH |
| | RDOPT |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | SLNS |
| | STDCMP |
| | VERIFY |
| | XSMESH |
| GIN | ADAN |
| | BALL |
| | DAN |
| | DANDAD |
| | FINNER |
| | GDAN |
| | OUTER |
| | SPHDAN |
| ICECOM | ICE |
| | MOREDT |
| | SETUPB |
| IOS | CSAS |
| | SASSY |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| LIBA | C4DATA |
| | EPSIG |
| | FIX |
| | FOX |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MULTRG |
| | RESDA |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | STDCMP |
| LIBC | C4DATA |
| | EPSIG |
| | FIX |
| | FOX |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MULTRG |
| | RESDA |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | STDCMP |
| LOGIC | ALBRD |
| | ARASIZ |
| | ARAYIN |
| | CORSIZ |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | FLDATA |
| | IDX1D |
| | JOMCHK |
| | JOMITY |
| | KENOG |
| | LOCATE |
| | MESH |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | PARTBL |
| | POINT |
| | POSIT |
| | PRTJOM |
| | PRTPLT |
| | RDBIAS |
| | RDBOX |
| | RDREF |
| | RDRST |
| | READGM |
| | RTARA |
| | SASSY |
| | SORTA |
| | SORTR |
| | VOLUME |
| | WATES |
| | WRTRST |
| LOWBND | CORSIZ |
| | GOCURS |
| | HUNTER |
| | JOMCHK |
| | LOCATE |
| | LODARA |
| | MESH |
| | POINT |
| | PRINT |
| | PRTJOM |
| | PRTPLT |
| | VOLUME |
| LPNT | CINIT |
| | RDOPT |
| | SASSY |
| MATRX | FLDATA |
| | PRATBL |
| | SORTA |
| NUTRON | FINDBX |
| | FLDATA |
| | LOCATE |
| | MESH |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| PARM | C4DATA |
| | MOREDT |
| | SETUPA |
| | SETUPB |
| PASS | C4DATA |
| | DANCOF |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MOREDT |
| | MULTRG |
| | POKE |
| | RESDA |
| | SASSY |
| | SETUPA |
| | SETUPB |
| | VERIFY |
| | XSDRN |
| PASS2 | C4DATA |
| | KNIGHT |
| | MIXTAB |
| | MOREDT |
| | SETUPA |
| | SETUPB |
| | XSDRN |
| PICT | MESH |
| | PRINT |
| | PRTPLT |
| | RDPLOT |
| | RELATE |
| PICTTL | MESH |
| | PRINT |
| | PRTPLT |
| | RDPLOT |
| | RELATE |
| POINTA | C4DATA |
| | SETUPB |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| POINTB | C4DATA |
| | SETUPA |
| POINTC | C4DATA |
| | SETUPA |
| POINTR | DATAIN |
| | JOMITY |
| | LOADIT |
| | POINT |
| | SASSY |
| REC1 | CNCN |
| | CSAS |
| | DMSN |
| | FIXPCH |
| | FIXTYP |
| | OPTPCH |
| | OPTYP |
| | PRTSRH |
| | RDOPT |
| | SASSY |
| RUNTYP | BIRITE |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | PARTBL |
| STDATA | DATAIN |
| | RDRST |
| | RDSTRT |
| | SASSY |
| | WRTRST |
| TITL | ARASIZ |
| | CORSIZ |
| | CPARAM |
| | DATAIN |
| | GEOMIN |
| | HOLCHK |
| | HOLEXT |
| | HOLHOL |
| | JOMCHK |
| | KENOG |
| | PARTBL |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | POINT |
| | PRATBL |
| | PRTJOM |
| | PRTLBA |
| | PRTPLT |
| | PRTSRH |
| | PRTWTS |
| | RDBOX |
| | RDPLOT |
| | READGM |
| | VOLUME |
| | WRTRST |
| UNIT | ALBRD |
| | ALBUSE |
| | ARASIZ |
| | ARAYIN |
| | BOX |
| | CINIT |
| | CNCN |
| | CNCSRH |
| | CORSIZ |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | DIFALB |
| | DIMSRH |
| | DMSN |
| | FIXCNC |
| | FIXDIM |
| | FIXPCH |
| | FIXTYP |
| | FLDATA |
| | GEOMIN |
| | ICE |
| | IDX1D |
| | JOMCHK |
| | JOMITY |
| | KENOG |
| | LOADIT |
| | LODARA |
| | LODWTS |
| | MIXIT |
| | MXTBLE |
| | OPTCNC |
| | OPTDMN |
| | OPTPCH |

Table C4.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | OPTYP |
| | PARTBL |
| | PCHSRH |
| | POINT |
| | POSIT |
| | PRINT |
| | PRTARA |
| | PRTJOM |
| | PRTLBA |
| | PRTPLT |
| | PRTSRH |
| | RDALB |
| | RDBIAS |
| | RDBOX |
| | RDICE |
| | RDOPT |
| | RDPLOT |
| | RDREF |
| | RDRST |
| | RDSTRT |
| | RDWTS |
| | READGM |
| | RT |
| | RTARA |
| | SASSY |
| | SAVST6 |
| | SHORTX |
| | SORTA |
| | SORTR |
| | VOLUME |
| | WAITIN |
| | WATES |
| | WRTALB |
| | WRTICE |
| | WRTRST |
| | WRTWTS |
| UNITS | CSAS |
| | C4DATA |
| | SASSY |
| XSDRNC | MOREDT |
| | SETUPB |
| | XSDRN |

Table C4.A.4 Commons used in MODIFY

| COMMON NAME | SUBROUTINE |
|---|---|
| ALBDAT | LODCOM |
| | RERITE |
| | RESTRT |
| | WRTALB |
| ALBNAM | LODCOM |
| | RERITE |
| | RESTRT |
| | WRTALB |
| CCDATA | MODIFY |
| CDATA | MODIFY |
| CYCLE | LODATA |
| | LODCOM |
| | MODIFY |
| DIMEN | LODATA |
| | LODCOM |
| | PRTJOM |
| | RERITE |
| | RESTRT |
| DRTACS | LODATA |
| | LODCOM |
| | MODIFY |
| | RERITE |
| | RESTRT |
| | WRTALB |
| | WRTWTS |
| LOGIC | PRTJOM |
| | RERITE |
| | RESTRT |
| LOWBND | PRTJOM |
| LPNT | LODATA |
| | MODIFY |
| REC1 | LODATA |
| | LODCOM |
| | MODIFY |
| | PRTSRH |
| | RERITE |

Table C4.A.4 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| STDATA | LODCOM |
| | RERITE |
| | RESTRT |
| TITL | LODATA |
| | PRTJOM |
| | PRTMIX |
| | PRTSRH |
| | RESTRT |
| UNIT | LODATA |
| | LODCOM |
| | MODIFY |
| | PRTJOM |
| | PRTSRH |
| | RERITE |
| | WRTALB |
| | WRTWTS |
| UNITS | MODIFY |

## C4.B SAMPLE PROBLEMS

This section contains sample problems to demonstrate some of the options available in CSAS and its associated sequences. A brief problem description and the associated card input data are included for each problem. See Sect. C4.5 for additional examples.

### SAMPLE PROBLEM 1 K-EFFECTIVE CALCULATION

The purpose of this problem is to calculate the k-effective of a system. This problem is the same as the KENO V.a sample problem 12 in Sect. F11.D except the cross-section library and KENO V.a mixing table are prepared by CSAS. The problem represents a critical experiment consisting of a composite array[1,2] of four highly enriched (93.2%) uranium metal cylinders having a density of 18.76 g/cc and four 5.0677-L Plexiglas containers filled with uranyl nitrate solution. The uranium metal cylinders have a radius of 5.748 cm and a height of 10.765 cm. The uranyl nitrate solution has a specific gravity of 1.555 and contains 415 g of uranium per liter. The ID of the Plexiglas bottle is 19.05 cm and the inside height is 17.78 cm. The Plexiglas is 0.635 cm thick. The center-to-center spacing between the metal units is 13.18 cm in the "y" direction and 13.45 cm in the "z" direction. The center-to-center spacing between the solution units is 21.75 cm in the "y" direction and 20.48 cm in the "z" direction. The spacing between the y-z plane that passes through the centers of the metal units and the y-z plane that passes through the centers of the solution units is 17.465 cm in the "x" direction.

The metal units in this experiment are designated in Table II of Ref. 1 as cylinder index 11 and reflector index 1. A photograph of the experiment is given in Fig. C4.B.1.

```
=CSAS25
SAMPLE PROBLEM 1  SET UP 4AQUEOUS 4 METAL IN CSAS25
HANSEN-ROACH    INFHOMMEDIUM
URANIUM         1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
SOLNUO2(NO3)2   2 415 9.783-3 SPG=1.555 1.0 293 92235 92.6 92238 5.9
                                                92234 1.0  92236 0.5 END
PLEXIGLASS      3 END
END COMP
KENO V.A SAMPLE PROBLEM 19 4 AQUEOUS 4 METAL ARRAY OF ARRAYS (SAMP PROB 12)
READ PARAM  FLX=YES FDN=YES NUB=YES NPG=300 GEN=103
END PARAM
READ GEOM
UNIT 1
COM='URANYL NITRATE SOLUTION IN A PLEXIGLAS CONTAINER'
CYLINDER  2 1 9.525 2P8.89
CYLINDER  3 1 10.16 2P9.525
CUBOID  0 1 4P10.875 2P10.24
UNIT 2
COM='URANIUM METAL CYLINDER'
CYLINDER  1 1 5.748 2P5.3825
CUBOID  0 1 4P6.59 2P6.225
UNIT 3
COM='1X2X2 ARRAY OF SOLUTION UNITS'
ARRAY 1 3*0.0
UNIT 4
COM='1X2X2 ARRAY OF METAL UNITS PADDED TO MATCH SOLUTION ARRAY'
ARRAY 2 3*0.0
REPLICATE 0 1 2*0.0 2*8.57 2*8.03 1
END GEOM
READ ARRAY ARA=1 NUX=1 NUY=2 NUZ=2 FILL F1 END FILL
ARA=2 NUX=1 NUY=2 NUZ=2 FILL F2 END FILL GBL=3 ARA=3 NUX=2 NUY=1 NUZ=1
COM='COMPOSITE ARRAY OF SOLUTION AND METAL UNITS'
FILL 4 3 END FILL
END ARRAY
END DATA
END
```

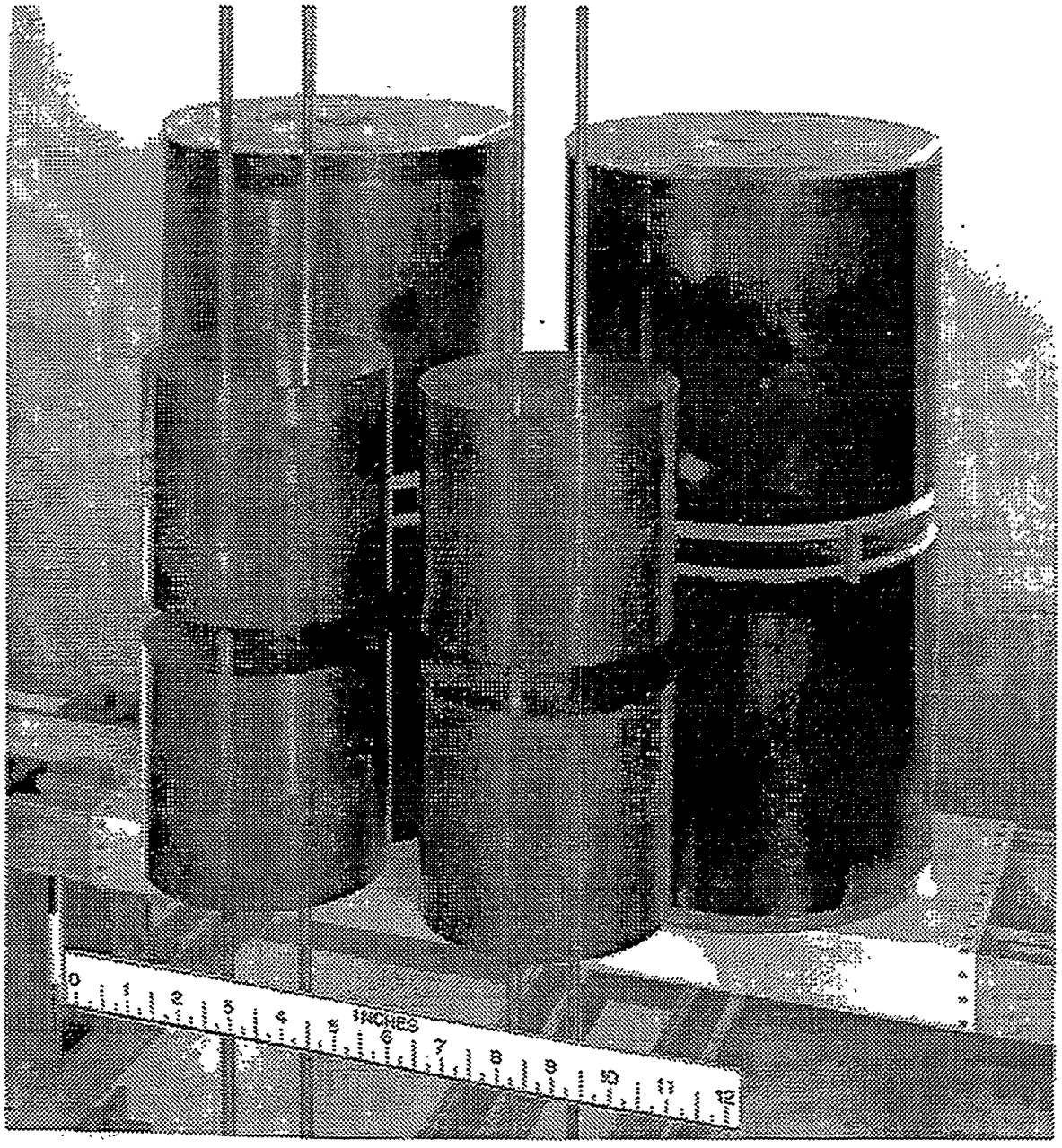



Figure C4.B.1 Critical assembly of four solution units and four metal units

## SAMPLE PROBLEM 2 OPTIMUM PITCH SEARCH USING DETAILED GEOMETRY

This problem represents an attempt to optimize the reactivity of PWR-like fuel bundles in a storage pool. The storage array is an infinite planar array of flooded fuel bundles. Each bundle consists of a 17 × 17 × 1 array of 2.35%-enriched $UO_2$ pins, density 9.21 g/cc, clad with zirconium. The fuel is 0.823 cm in diameter, the clad diameter is 0.9627 cm, and the length of each pin is 366 cm. Each fuel bundle is encased in a boral sheath. There is a 1/4-in. gap flooded with water between the bundle and the sheath. The boral sheath is 3/8 in. thick. One inch of water separates the fuel bundle sheaths in the horizontal plane, and 15 cm of water is present on the top and bottom of the array.

The KENO V.a geometry represents each fuel pin in the bundle discreetly. The search should determine the optimum pitch within the fuel bundle. The gap between the bundle and the boral remains fixed, as does the thickness of the sheath and the spacing between sheaths.

```
=CSAS4
SAMPLE PROBLEM 2  PWR-LIKE FUEL BUNDLE
HANSEN-ROACH LATTICECELL
UO2  1 .84 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
H2O  5 1 END
END COMP
SQUAREPITCH  1.2751 .823 1 3 .9627 2 END
STORAGE ARRAY OF PWR-LIKE FUEL BUNDLES IN POISON SHEATHS
READ PARAM  NUB=YES FAR=YES GEN=53 NPG=300
END PARAM
READ ARRAY NUX=17 NUY=17 NUZ=1 END ARRAY
READ BOUNDS  XYF=MIRROR   END BOUNDS
READ GEOM
CYLINDER  1 1 .4115 183.0 -183.0
CYLINDER  2 1 .48135 183.1 -183.1
CUBOID  3 1 .63755 -.63755 .63755 -.63755 183.1 -183.1
CORE  0 1 0 0 0
REFLECTOR 5 1 4*.635 2Z 1
REFLECTOR 4 1 4*.9525 2Z 1
REFLECTOR 5 1 4*1.27 2Z 1
REFLECTOR 5 2 4Z 2*3.0 5
END GEOM
READ BIAS ID=500 2 6 END BIAS
END DATA
READ SEARCH  OPTIMUM PITCH  END SEARCH
END
```

## SAMPLE PROBLEM 3 OPTIMUM PITCH SEARCH USING HOMOGENIZED GEOMETRY

This problem illustrates the use of CSAS4X to represent a fuel cask. It utilizes a cell-weighted mixture to represent a PWR-like fuel bundle. The cask contains a 2 × 2 × 1 array of fuel bundles. Each fuel bundle consists of a 17 × 17 × 1 array of zirconium clad, 2.35%-enriched $UO_2$ fuel pins with a density of 9.21 g/cc arranged in a square pitch. The pin diameter is 0.823 cm, and its length is 366 cm. The clad is 0.06985 cm thick, and the pitch is 1.275 cm. Each fuel bundle is contained in a 0.6625-cm-thick boral sheath. The bundles are separated by 1 cm of water, representing a flooded cask. The square aluminum cask is 10 cm thick on all faces and is reflected by 15 cm of water.

By using CSAS4X, a cell-weighted cross section is created to represent the fuel bundle. The KENO V.a geometry utilizes the cell-weighted mixture (500) and the overall dimensions of the fuel bundle to represent the

entire fuel bundle as a single homogeneous region. The first reflector card represents the fuel cask, and the second reflector card represents the 15-cm reflector.

```
=CSAS4X
SAMPLE PROBLEM 3  SAMPLE FUEL CASK
HANSEN-ROACH LATTICECELL
UO2  1 .84 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
H2O  5 1 END
AL  6 1 END
END COMP
SQUAREPITCH  1.275 .823 1 3 .9627 2 END
SAMPLE SQUARE FUEL CASK
READ PARAM  NUB=YES FAR=YES GEN=53 NPG=300
END PARAM
READ ARRAY NUX=2  NUY=2  NUZ=1 END ARRAY
READ GEOM
CUBOID 500 1 4P10.8375 2P183.0
CUBOID   4 1 4P11.5    2P183.0
CUBOID   5 1 4P12.0    2P183.0
CORE     0 1 3*0
REFLECTOR 6 1 6*10.0 1
REFLECTOR 5 2 6*3 5
END GEOM
READ BIAS ID=500 2 6 END BIAS
END DATA
READ SEARCH  OPTIMUM PITCH  END SEARCH
END
```

## SAMPLE PROBLEM 4 SEARCH FOR A SPECIFIED VALUE OF K-EFFECTIVE

Find the pitch at which a 2 × 2 × 2 array of cylinders of highly enriched (93.2%) uranium metal with a density of 18.76 g/cc are critical. Each cylinder has a radius of 5.748 cm and a height of 10.765 cm. The surface-to-surface spacing between the units is the same in all directions. The initial guess for the critical surface-to-surface spacing was 3.0 cm. The experimentally critical surface-to-surface separation for this system, shown in Fig. C4.B.2, is 2.248 cm. The input data for this problem are given below.

```
=CSAS4
SAMPLE PROBLEM 4  SET UP 2C8 IN CSAS25
HANSEN-ROACH   INFHOMMEDIUM
URANIUM  1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
CRITICAL PITCH SEARCH FOR CASE 2C8 BARE
READ PARAMETERS  FLX=YES FDN=YES FAR=YES GEN=103 NPG=300
END PARAMETERS
READ GEOMETRY
CYLINDER 1 1 5.748 5.3825 -5.3825
CUBOID  0 1 4P7.248 2P6.8825
END GEOMETRY
READ ARRAY  NUX=2 NUY=2 NUZ=2  END ARRAY
END DATA
READ SEARCH   CRITICAL PITCH   MAXPITCH=15.5  MORE
ALTER  UNIT=1 REG=2 +Z=1.0 -Z=1.0
END SEARCH
END
```

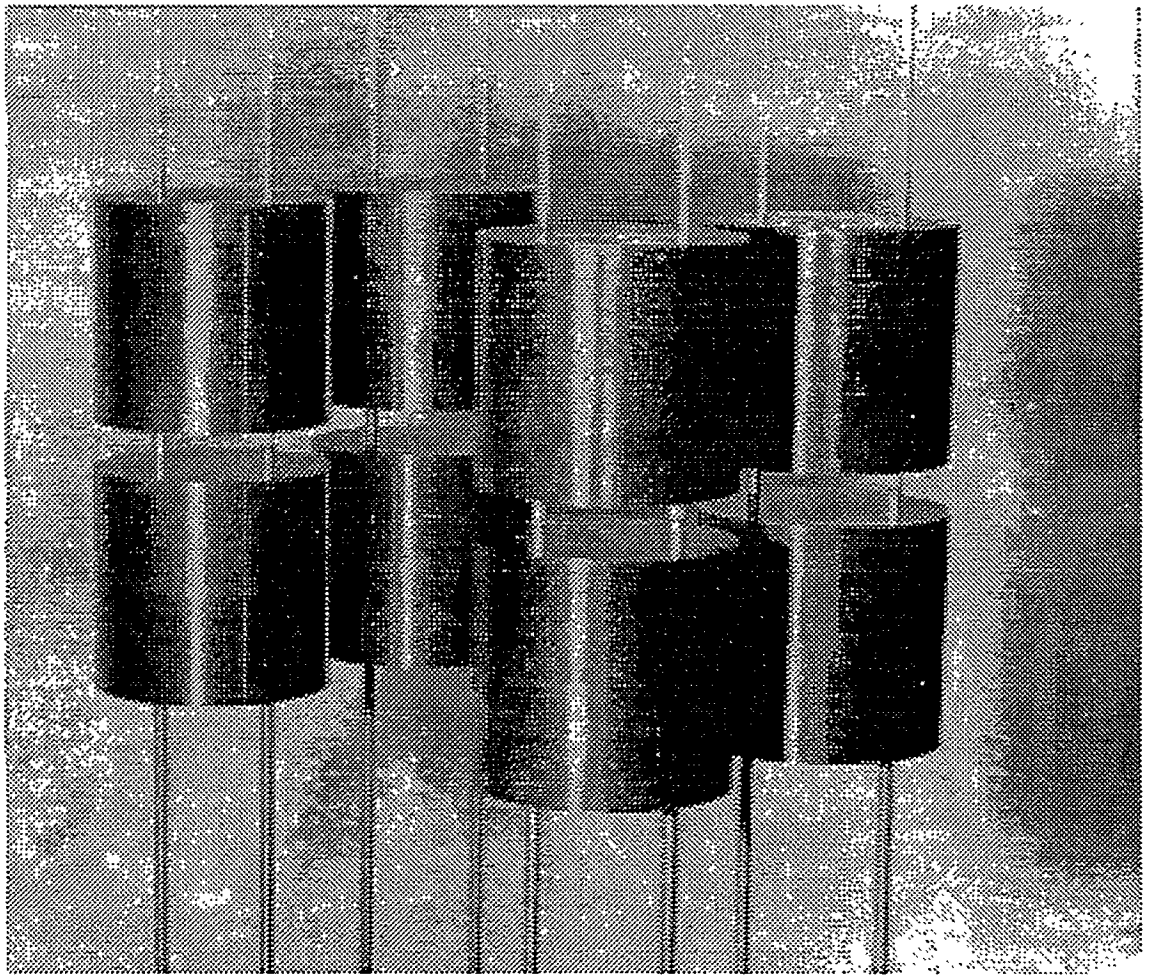


Figure C4.B.2 Critical assembly of eight highly enriched uranium metal cylinders

## C4.B.1 REFERENCES


1. J. T. Thomas, "Critical Three-Dimensional Arrays of U(93.2)-Metal Cylinders," *Nucl. Sci. Eng.* **52**, 350 (November 1973).

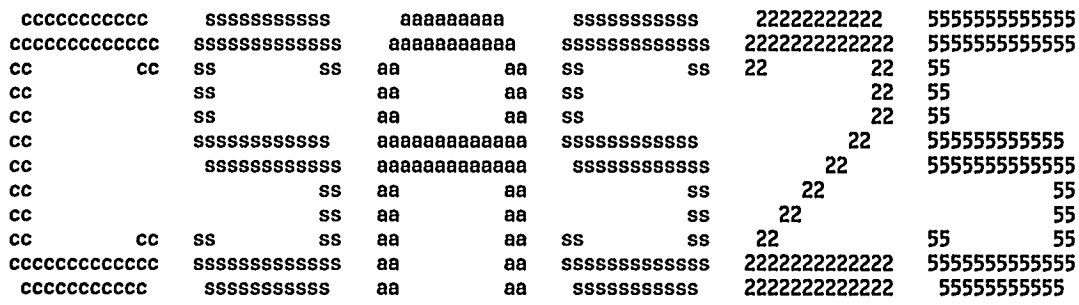
2. J. T. Thomas, *Critical Three-Dimensional Arrays of Neutron-Interacting Units, Part II*, ORNL/TM-868, Union Carbide Corp., Nucl. Div., Oak Ridge Natl. Lab. (July 1964).

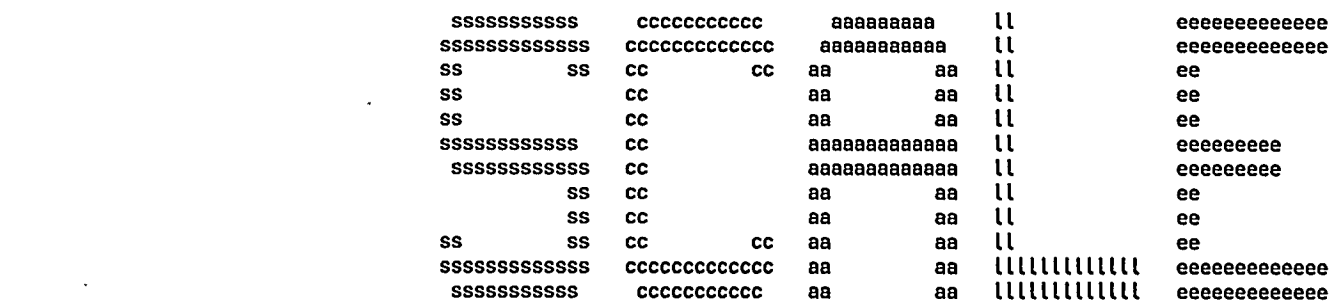
## C4.C SAMPLE OUTPUT

This section contains the computer printout generated by the control module for each sample problem. Output from functional modules is not included in this section. See Sect. C4.6 for a generic explanation of the control module output.

Sample problem 1 does not perform a search, so the k-effective resulting from this calculation does not appear in the control module output. This problem is a description of a critical experiment, so the KENO V.a output should yield a k-effective of 1.0 within statistical accuracy.

Sample problems 2, 3, and 4 perform searches. The search data and search results are printed after the program verification table for MODIFY. The k-effective is printed as each search pass is completed. These k-effectives may vary because of differences in the random number sequences on different computers.


cccccccccc sssssssssss aaaaaaaaa sssssssssss 22222222222 5555555555555
ccccccccccccc ssssssssssssss aaaaaaaaaaa ssssssssssssss 2222222222222 5555555555555
cc cc ss ss aa aa ss ss 22 22 55
cc ss aa aa ss 22 55
cc ss aa aa ss 22 55
cc ssssssssssss aaaaaaaaaaaaa sssssssssssss 22 555555555555
cc sssssssssssss aaaaaaaaaaaaa sssssssssssss 22 5555555555555
cc ss aa aa ss 22 55
cc ss aa aa ss 22 55
cc cc ss ss aa aa ss ss 22 55 55
ccccccccccccc ssssssssssssss aa aa sssssssssssss 2222222222222 5555555555555
cccccccccc sssssssssss aa aa sssssssssss 2222222222222 55555555555
sssssssssss ttttttttttttt 5555555555555
ssssssssssssss ttttttttttttt 5555555555555
ss ss tt 55
ss tt 55
ss tt 55
ssssssssssss tt 555555555555
sssssssssssss tt 5555555555555
ss tt 55
ss tt 55
ss ss tt 55 55
ssssssssssssss tt 5555555555555
sssssssssss tt 55555555555
0000000 33333333333 // 11 44 // 99999999999 44
000000000 3333333333333 // 111 444 // 9999999999999 444
00 00 33 33 // 1111 4444 // 99 99 4444
00 00 33 // 11 44 44 // 99 99 44 44
00 00 33 // 11 44 44 // 99 99 44 44
00 00 333 // 11 44 44 // 9999999999999 44 44
00 00 333 // 11 44 44 // 999999999999 44 44
00 00 33 // 11 444444444444 // 99 444444444444
00 00 33 // 11 4444444444444 // 99 4444444444444
00 00 33 33 // 11 44 // 99 44
000000000 3333333333333 // 11111111 44 // 9999999999999 44
0000000 33333333333 // 11111111 44 // 999999999999 44
11 666666666666 33333333333 22222222222 33333333333 0000000
111 6666666666666 3333333333333 2222222222222 3333333333333 000000000
1111 66 ::: 33 33 22 22 ::: 33 33 00 00
11 66 ::: 33 22 ::: 33 00 00
11 66 ::: 33 22 ::: 33 00 00
11 666666666666 333 22 333 00 00
11 6666666666666 333 22 333 00 00
11 66 66 ::: 33 22 ::: 33 00 00
11 66 66 ::: 33 22 ::: 33 00 00
11 66 66 ::: 33 33 22 ::: 33 33 00 00
11111111 6666666666666 3333333333333 2222222222222 3333333333333 000000000
11111111 66666666666 33333333333 2222222222222 33333333333 0000000




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  csas                                         *****
*****                                                                     *****
*****        creation date:  12/01/93                                     *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  csas                                         *****
*****                                                                     *****
*****              version:  2.2                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:32:30                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
sample problem 1  set up 4aqueous 4 metal in csas25


**** problem parameters ****

lib hansen-roach library
mxx           3 mixtures
msc           3 composition specifications
izm           1 material zones
ge  infhommedium geometry
more          0 0/1 do not read/read optional parameter data
msln          1 fuel solutions


**** problem composition description ****

sc  uranium      standard composition
mx            1 mixture no.
vf         .9850 volume fraction
roth     19.0500 theoretical density
temp       293.0 deg kelvin
           92235  93.20%
           92238   5.60%
           92234   1.00%
           92236    .20%
end

sc  solnuo2(no3) standard composition
mx            2 mixture no.
fd       415.0000 solution fuel density
aml         .0098 acid molarity
vf         1.0000 volume fraction
temp       293.0 deg kelvin
spg        1.5550 specified specific gravity
           92235  92.60%
           92238   5.90%
           92234   1.00%
           92236    .50%
end

sc  plexiglass   standard composition
mx            3 mixture no.
vf         1.0000 volume fraction
roth       1.1800 theoretical density
end


**** problem geometry ****


**** infinite homogeneous medium ****
mfuel         1 mixture no. of the infinite homogeneous medium
```

```
*****************************************************************************************************************
***                                                                                                           ***
***                          sample problem 1  set up 4aqueous 4 metal in csas25                              ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************
***                                                                                                           ***
***                            ********** data library information **********                               ***
***                                                                                                           ***
***          unit                                                  volume                                     ***
***         number              data set name                      name        unit function                  ***
***         ------              -------------                      ----        -------------                  ***
***                                                                                                           ***
***           89      /scale4.2p/datalib/scale.rev05.sclib                     standard composition library     ***
***                                                                                                           ***
***           81      /scale4.2p/datalib/scale.rev02.xn16                      cross section library            ***
***                                                                                                           ***
***           11      ft11f001                                                 short cross section library      ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************
***                                                                                                           ***
***                                                                                                           ***
***                                  standard composition library data                                        ***
***                                  ---------------------------------                                        ***
***                                                                                                           ***
***           unit number  :  89                                                                              ***
***                                                                                                           ***
***           dataset name :  /scale4.2p/datalib/scale.rev05.sclib                                            ***
***                                                                                                           ***
***           library title:  scale-4 standard composition library                                            ***
***                           386 standard compositions, 332 nuclides                                         ***
***                           10 elements with variable isotopic distrbutions.                                ***
***                                                                                                           ***
***           creation date:   1/25/94                                                                        ***
***                                                                                                           ***
***                                                                                                           ***
***                                                                                                           ***
***                                   cross section library data                                              ***
***                                  ---------------------------                                              ***
***                                                                                                           ***
***           unit number  :  81                                                                              ***
***                                                                                                           ***
***           dataset name :  /scale4.2p/datalib/scale.rev02.xn16                                             ***
***                                                                                                           ***
***           library title:  scale 4 - 16 neutron group criticality safety library                           ***
***                             hansen-roach data with knight modifications and some endf/b 4 data            ***
***                               compiled for nrc       1/27/89                                              ***
***                                 last updated        10/12/89                                              ***
***                                   l.m.petrie       ornl                                                   ***
***                                                                                                           ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************

                .....    0 io's were used before reading keno v data    .....
```

```
                    ........    0 io's were used reading the keno v parameter data    ........

                   *************** data reading completed ***************

                    ........    0 io's were used preparing the keno v input data    ........

                    ........    0 io's were used loading the keno v data    ........

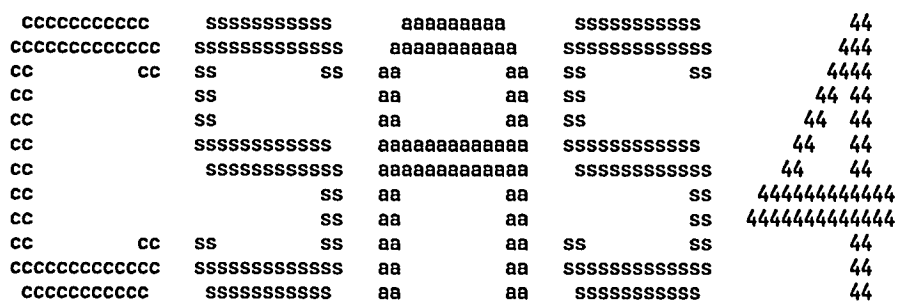
                    ........    0 io's were used loading the data    ........

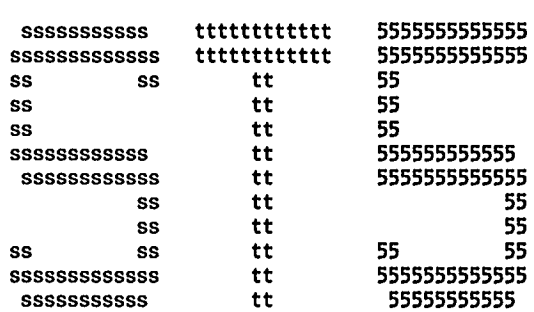
                    ........    0 io's were used checking the keno v geometry data    ........
                   *****  restart data has been written on unit 95  *****

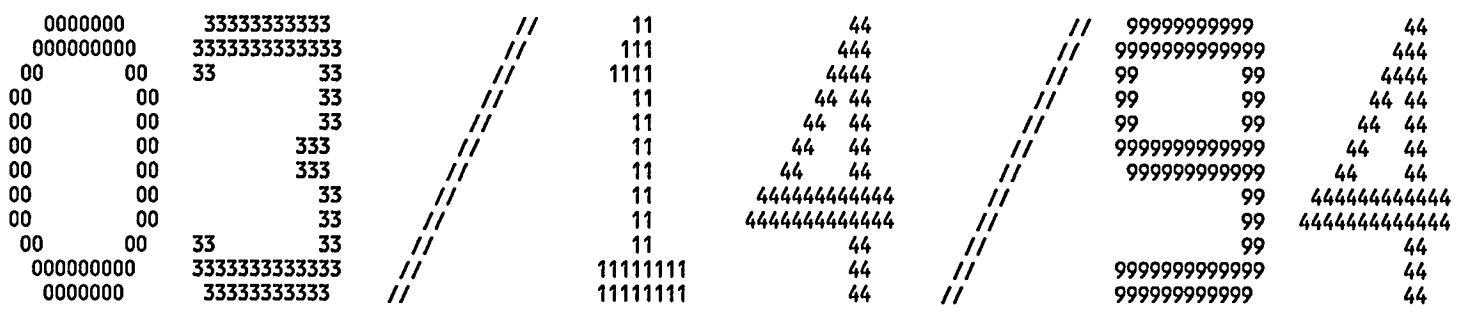
                    ........    0 io's were used writing the keno v - csas data    ........

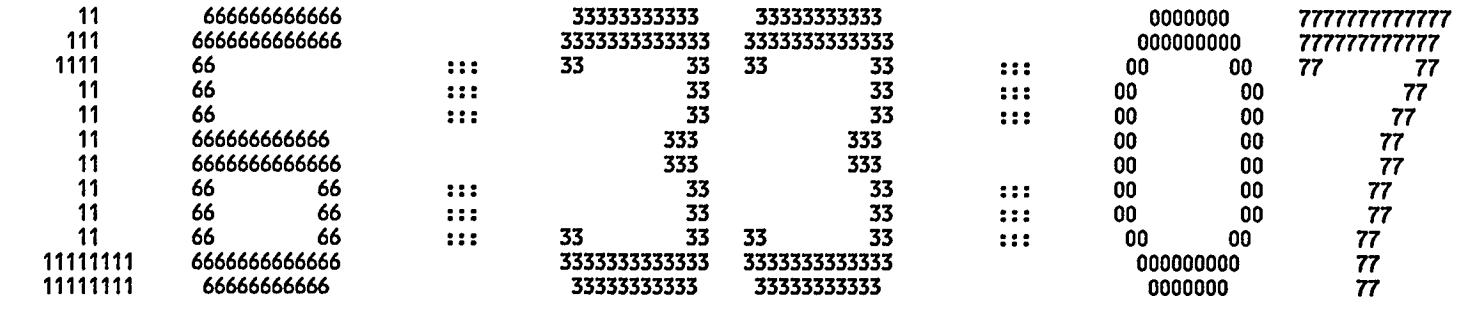
                    ........    0 io's were used processing csas input data    ........

control module csas25   is complete.
```


cccccccccc ssssssssss aaaaaaaa ssssssssss 44
ccccccccccc sssssssssss aaaaaaaaaa sssssssssss 444
cc cc ss ss aa aa ss ss 4444
cc ss aa aa ss 44 44
cc ss aa aa ss 44 44
cc sssssssssss aaaaaaaaaaaa sssssssssss 44 44
cc ssssssssss aaaaaaaaaaaa ssssssssss 44 44
cc ss aa aa ss 444444444444
cc ss aa aa ss 4444444444444
cc cc ss ss aa aa ss ss 44
ccccccccccc sssssssssss aa aa sssssssssss 44
cccccccccc ssssssssss aa aa ssssssssss 44


ssssssssss ttttttttttt 5555555555555
sssssssssss ttttttttttt 5555555555555
ss ss tt 55
ss tt 55
ss tt 55
sssssssssss tt 555555555555
ssssssssss tt 5555555555555
ss tt 55
ss tt 55
ss ss tt 55 55
sssssssssss tt 5555555555555
ssssssssss tt 55555555555


0000000 33333333333 // 11 44 // 99999999999 44
000000000 3333333333333 // 111 444 // 9999999999999 444
00 00 33 33 // 1111 4444 // 99 99 4444
00 00 33 // 11 44 44 // 99 99 44 44
00 00 33 // 11 44 44 // 99 99 44 44
00 00 333 // 11 44 44 // 9999999999999 44 44
00 00 333 // 11 44 44 // 999999999999 44 44
00 00 33 // 11 444444444444 // 99 444444444444
00 00 33 // 11 444444444444 // 99 444444444444
00 00 33 33 // 11 44 // 99 44
000000000 3333333333333 // 11111111 44 // 9999999999999 44
0000000 33333333333 // 11111111 44 // 999999999999 44


11 666666666666 33333333333 33333333333 0000000 7777777777777
111 6666666666666 3333333333333 3333333333333 000000000 777777777777
1111 66 ::: 33 33 33 33 ::: 00 00 77 77
11 66 ::: 33 33 ::: 00 00 77
11 66 ::: 33 33 ::: 00 00 77
11 666666666666 333 333 00 00 77
11 6666666666666 333 333 00 00 77
11 66 66 ::: 33 33 ::: 00 00 77
11 66 66 ::: 33 33 ::: 00 00 77
11 66 66 ::: 33 33 33 33 ::: 00 00 77
11111111 6666666666666 3333333333333 3333333333333 000000000 77
11111111 66666666666 33333333333 33333333333 0000000 77

```
 ssssssssss     ccccccccccc      aaaaaaaaa     ll              eeeeeeeeeeee
ssssssssssss   ccccccccccccc    aaaaaaaaaaa    ll              eeeeeeeeeeee
ss          ss cc          cc  aa          aa  ll              ee
ss             cc              aa          aa  ll              ee
ss             cc              aa          aa  ll              ee
ssssssssssss   cc              aaaaaaaaaaaaa  ll              eeeeeeeee
 ssssssssssss  cc              aaaaaaaaaaaaa  ll              eeeeeeeee
          ss   cc              aa          aa  ll              ee
          ss   cc              aa          aa  ll              ee
ss        ss   cc          cc  aa          aa  ll              ee
ssssssssssss   ccccccccccccc   aa          aa  lllllllllllll   eeeeeeeeeeee
 ssssssssss     ccccccccccc    aa          aa  lllllllllllll   eeeeeeeeeeee
```

```
********************************************************************************
********************************************************************************
********************************************************************************
*****                                                                      *****
*****                  program verification information                 *****
*****                                                                      *****
*****                code system:    scale  version:    4.2              *****
*****                                                                      *****
********************************************************************************
********************************************************************************
*****                                                                      *****
*****                                                                      *****
*****              program:  csas                                          *****
*****                                                                      *****
*****         creation date:  12/01/93                                     *****
*****                                                                      *****
*****              library:  /scale4.2p/exe                                *****
*****                                                                      *****
*****                                                                      *****
*****       production code:  csas                                         *****
*****                                                                      *****
*****              version:  2.2                                           *****
*****                                                                      *****
*****              jobname:  st5                                           *****
*****                                                                      *****
*****     date of execution:  03/14/94                                     *****
*****                                                                      *****
*****     time of execution:  16:33:08                                     *****
*****                                                                      *****
*****                                                                      *****
********************************************************************************
********************************************************************************
********************************************************************************
```

```
sample problem 2  pwr-like fuel bundle


**** problem parameters ****

lib hansen-roach library
mxx           5 mixtures
msc           6 composition specifications
izm           3 material zones
ge  latticecell  geometry
more          0 0/1 do not read/read optional parameter data
msln          0 fuel solutions


**** problem composition description ****

sc  uo2          standard composition
mx             1 mixture no.
vf         .8400 volume fraction
roth     10.9600 theoretical density
temp       293.0 deg kelvin
           92235   2.35%
           92238  97.65%
end

sc  zr           standard composition
mx             2 mixture no.
vf        1.0000 volume fraction
roth      6.4400 theoretical density
end

sc  h2o          standard composition
mx             3 mixture no.
vf        1.0000 volume fraction
roth       .9982 theoretical density
end

sc  b4c          standard composition
mx             4 mixture no.
vf         .3670 volume fraction
roth      2.5200 theoretical density
end

sc  al           standard composition
mx             4 mixture no.
vf         .6360 volume fraction
roth      2.6989 theoretical density
end

sc  h2o          standard composition
mx             5 mixture no.
vf        1.0000 volume fraction
roth       .9982 theoretical density
end
```

```
**** problem geometry ****

ctp squarepitch  cell type
pitch     1.2751 cm center to center spacing
fuelod     .8230 cm fuel diameter or slab thickness
mfuel         1 mixture no. of fuel
mmod          3 mixture no. of moderator
cladod     .9627 cm clad outer diameter
mclad         2 mixture no. of clad


zone specifications for latticecell geometry

              zone  1 is fuel
              zone  2 is clad
              zone  3 is mod
```

```
*****************************************************************************************************************
***                                                                                                           ***
***                              sample problem 2  pwr-like fuel bundle                                       ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************
***                                                                                                           ***
***                               ********** data library information **********                            ***
***                                                                                                           ***
***          unit                                                  volume                                     ***
***         number              data set name                      name        unit function                  ***
***         ------              -------------                      ----        -------------                  ***
***                                                                                                           ***
***           89      /scale4.2p/datalib/scale.rev05.sclib                      standard composition library   ***
***                                                                                                           ***
***           81      /scale4.2p/datalib/scale.rev02.xn16                       cross section library          ***
***                                                                                                           ***
***           11      ft11f001                                                  short cross section library    ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************
***                                                                                                           ***
***                                                                                                           ***
***                                    standard composition library data                                      ***
***                                    ---------------------------------                                      ***
***                                                                                                           ***
***           unit number  :  89                                                                              ***
***                                                                                                           ***
***           dataset name :  /scale4.2p/datalib/scale.rev05.sclib                                            ***
***                                                                                                           ***
***           library title:  scale-4 standard composition library                                            ***
***                           386 standard compositions, 332 nuclides                                         ***
***                           10 elements with variable isotopic distrbutions.                                ***
***                                                                                                           ***
***           creation date:   1/25/94                                                                        ***
***                                                                                                           ***
***                                                                                                           ***
***                                                                                                           ***
***                                     cross section library data                                            ***
***                                    ---------------------------                                            ***
***                                                                                                           ***
***           unit number  :  81                                                                              ***
***                                                                                                           ***
***           dataset name :  /scale4.2p/datalib/scale.rev02.xn16                                             ***
***                                                                                                           ***
***           library title:  scale 4 - 16 neutron group criticality safety library                           ***
***                             hansen-roach data with knight modifications and some endf/b 4 data            ***
***                               compiled for nrc       1/27/89                                              ***
***                                 last updated        10/12/89                                              ***
***                                   l.m.petrie       ornl                                                   ***
***                                                                                                           ***
***                                                                                                           ***
*****************************************************************************************************************
*****************************************************************************************************************

                .....    0 io's were used before reading keno v data    .....
```

```
........    0 io's were used reading the keno v parameter data    ........

*************** data reading completed ***************

........    0 io's were used preparing the keno v input data    ........

........    0 io's were used loading the keno v data    ........

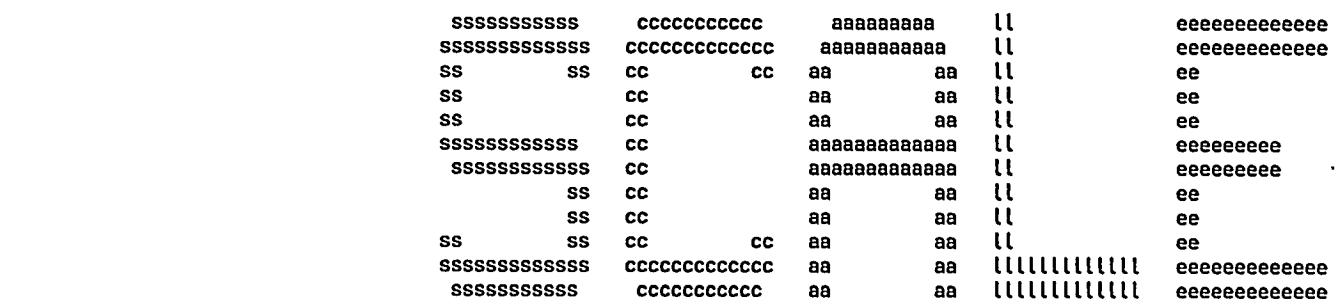
........    0 io's were used loading the data    ........

........    0 io's were used checking the keno v geometry data    ........
*****  restart data has been written on unit 95  *****

........    0 io's were used writing the keno v - csas data    ........

........    0 io's were used before reading search data    ........

........    0 io's were used processing csas input data    ........
```




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  modify                                       *****
*****                                                                     *****
*****        creation date:  10/25/93                                     *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  modify                                       *****
*****                                                                     *****
*****              version:  2.1                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:34:22                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
*********************************************************************************************************************
***                                                                                                               ***
***                          storage array of pwr-like fuel bundles in poison sheaths                             ***
***                                                                                                               ***
*********************************************************************************************************************
*********************************************************************************************************************
***                              ******          search data          ******                                     ***
***                                                                                                               ***
***                                                                                                               ***
***                              search type                          optimum pitch                              ***
***                                                                                                               ***
***                     pas      number of search passes                 10                                     ***
***                                                                                                               ***
***                     npm      number of search parameters              1                                     ***
***                                                                                                               ***
***                     eps      search convergence tolerance          .0050                                    ***
***                     -con     minimum constraint                   -.2450                                    ***
***                                                                                                               ***
***                     +con     maximum constraint                    1.2250                                   ***
***                                                                                                               ***
***                                                                                                               ***
***                       unit    first   last                                                                  ***
***      command          number  region  region  ----------------    search constants    ----------------      ***
***                                                                                                               ***
***                       unit    reg=__  to __   +x or r  -x or +h  +y or -h     -y       +z        -z        ***
***                                                                                                               ***
***  alter/change/modify    1       3       3       1.00      1.00      1.00      1.00      .00       .00       ***
***                                                                                                               ***
*********************************************************************************************************************
*********************************************************************************************************************
```

```
*******************   search pass  1     keff=  7.31988E-01 + or -  6.08683E-03   *******************
                            the parameter was  0.00000E+00
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  2     keff=  3.58234E-01 + or -  4.64021E-03   *******************
                            the parameter was -2.45000E-01
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  3     keff=  7.44512E-01 + or -  3.79809E-03   *******************
                            the parameter was  1.22500E+00
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  4     keff=  8.97264E-01 + or -  4.79952E-03   *******************
                            the parameter was  6.12501E-01
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  5     keff=  9.13226E-01 + or -  4.64053E-03   *******************
                            the parameter was  4.12008E-01
```

storage array of pwr-like fuel bundles in poison sheaths

convergence was achieved on pass 5 the parameter was 4.12008E-01

the equation used in the search was:

k-eff = +7.18481E-01 +1.00598E+00*p -1.54341E+00*p**2 +6.05700E-01*p**3

k-effective= 9.13226E-01 + or - 4.64053E-03 the corresponding geometry follows;

| region | media num | bias id | geometry description for those units utilized in this problem |
|---|---|---|---|
| | | | ----- unit 1 ----- |
| 1 cylinder | 1 | 1 | radius = .41150 +z = 183.00 -z = -183.00 centerline is at x = 0.00000E+00 y = 0.00000E+00 |
| 2 cylinder | 2 | 1 | radius = .48135 +z = 183.10 -z = -183.10 centerline is at x = 0.00000E+00 y = 0.00000E+00 |
| 3 cuboid | 3 | 1 | +x = .90023 -x = -.90023 +y = .90023 -y = -.90023 +z = 183.10 -z = -183.10 |

storage array of pwr-like fuel bundles in poison sheaths

based on the preceding data, the best estimate of the parameter is 4.39709E-01

the geometry corresponding to this parameter follows:

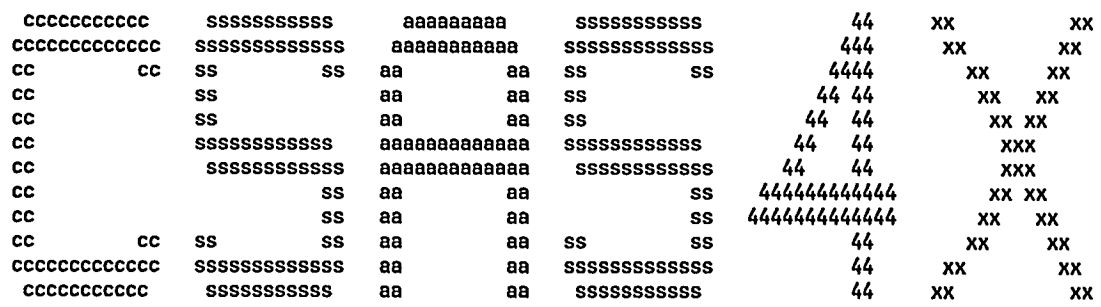
| region | media num | bias id | geometry description for those units utilized in this problem |
|---|---|---|---|
| | | | ----- unit 1 ----- |
| 1 cylinder | 1 | 1 | radius = .41150 +z = 183.00 -z = -183.00 centerline is at x = 0.00000E+00 y = 0.00000E+00 |
| 2 cylinder | 2 | 1 | radius = .48135 +z = 183.10 -z = -183.10 centerline is at x = 0.00000E+00 y = 0.00000E+00 |
| 3 cuboid | 3 | 1 | +x = .91789 -x = -.91789 +y = .91789 -y = -.91789 +z = 183.10 -z = -183.10 |

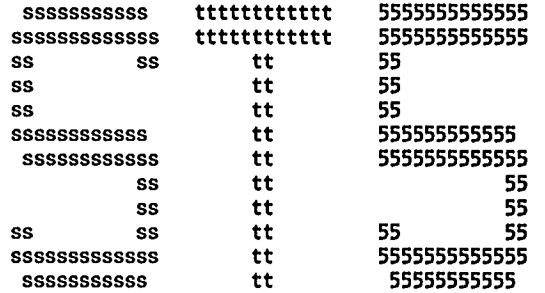
**********
**********

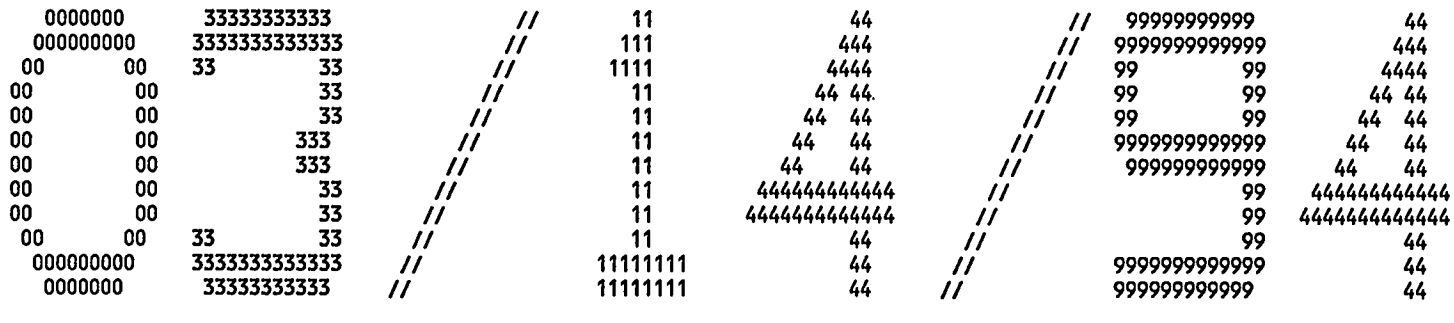
***** modified keno v data has been rewritten on unit 95 *****

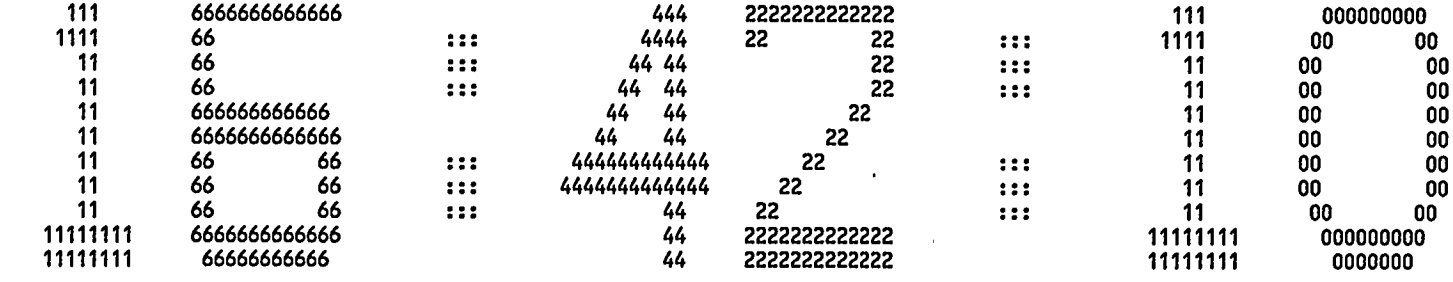
***** convergence has been achieved in the search package. *****

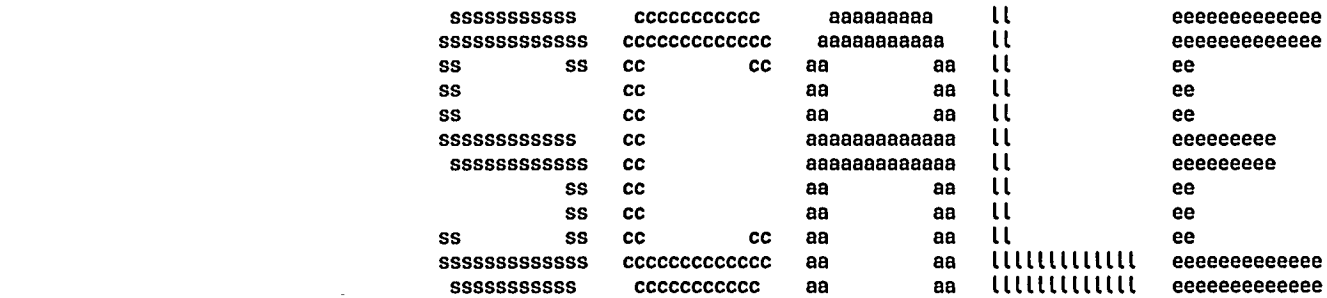
control module modify is complete.


cccccccccc ssssssssss aaaaaaaaa ssssssssss 44 xx xx
cccccccccccc ssssssssssss aaaaaaaaaaa ssssssssssss 444 xx xx
cc cc ss ss aa aa ss ss 4444 xx xx
cc ss aa aa ss 44 44 xx xx
cc ss aa aa ss 44 44 xx xx
cc sssssssssss aaaaaaaaaaaaa sssssssssss 44 44 xxx
cc ssssssssssss aaaaaaaaaaaaa ssssssssssss 44 44 xxx
cc ss aa aa ss 444444444444 xx xx
cc ss aa aa ss 4444444444444 xx xx
cc cc ss ss aa aa ss ss 44 xx xx
cccccccccccc ssssssssssss aa aa ssssssssssss 44 xx xx
cccccccccc ssssssssss aa aa ssssssssss 44 xx xx


ssssssssss ttttttttttt 5555555555555
ssssssssssss ttttttttttt 5555555555555
ss ss tt 55
ss tt 55
ss tt 55
sssssssssss tt 555555555555
ssssssssssss tt 5555555555555
ss tt 55
ss tt 55
ss ss tt 55 55
sssssssssssss tt 5555555555555
ssssssssss tt 55555555555


0000000 33333333333 // 11 44 // 99999999999 44
000000000 3333333333333 // 111 444 // 9999999999999 444
00 00 33 33 // 1111 4444 // 99 99 4444
00 00 33 // 11 44 44 // 99 99 44 44
00 00 33 // 11 44 44 // 99 99 44 44
00 00 333 // 11 44 44 // 9999999999999 44 44
00 00 333 // 11 44 44 // 999999999999 44 44
00 00 33 // 11 444444444444 // 99 444444444444
00 00 33 // 11 4444444444444 // 99 4444444444444
00 00 33 33 // 11 44 // 99 44
000000000 3333333333333 // 11111111 44 // 9999999999999 44
0000000 33333333333 // 11111111 44 // 999999999999 44


11 666666666666 44 22222222222 11 0000000
111 6666666666666 444 2222222222222 111 000000000
1111 66 ::: 4444 22 22 ::: 1111 00 00
11 66 ::: 44 44 22 ::: 11 00 00
11 66 ::: 44 44 22 ::: 11 00 00
11 666666666666 44 44 22 11 00 00
11 6666666666666 44 44 22 11 00 00
11 66 66 ::: 444444444444 22 ::: 11 00 00
11 66 66 ::: 4444444444444 22 ::: 11 00 00
11 66 66 ::: 44 22 ::: 11 00 00
11111111 6666666666666 44 2222222222222 11111111 000000000
11111111 66666666666 44 2222222222222 11111111 0000000




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  csas                                         *****
*****                                                                     *****
*****        creation date:  12/01/93                                     *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  csas                                         *****
*****                                                                     *****
*****              version:  2.2                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:42:11                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
sample problem 3  sample fuel cask

**** problem parameters ****

lib hansen-roach library
mxx           6 mixtures
msc           7 composition specifications
izm           3 material zones
ge  latticecell  geometry
more          0 0/1 do not read/read optional parameter data
msln          0 fuel solutions

**** problem composition description ****

sc  uo2         standard composition
mx            1 mixture no.
vf        .8400 volume fraction
roth    10.9600 theoretical density
temp     293.0 deg kelvin
          92235   2.35%
          92238  97.65%
end

sc  zr          standard composition
mx            2 mixture no.
vf       1.0000 volume fraction
roth     6.4400 theoretical density
end

sc  h2o         standard composition
mx            3 mixture no.
vf       1.0000 volume fraction
roth      .9982 theoretical density
end

sc  b4c         standard composition
mx            4 mixture no.
vf        .3670 volume fraction
roth     2.5200 theoretical density
end

sc  al          standard composition
mx            4 mixture no.
vf        .6360 volume fraction
roth     2.6989 theoretical density
end

sc  h2o         standard composition
mx            5 mixture no.
vf       1.0000 volume fraction
roth      .9982 theoretical density
end


sc  al          standard composition
mx            6 mixture no.
```

```
vf        1.0000 volume fraction
roth      2.6989 theoretical density
end


**** problem geometry ****

ctp squarepitch  cell type
pitch     1.2750 cm center to center spacing
fuelod     .8230 cm fuel diameter or slab thickness
mfuel           1 mixture no. of fuel
mmod            3 mixture no. of moderator
cladod     .9627 cm clad outer diameter
mclad           2 mixture no. of clad


zone specifications for latticecell geometry

                zone  1 is fuel
                zone  2 is clad
                zone  3 is mod
```

```
*** sample problem 3  sample fuel cask ***

********** data library information **********

unit number | data set name | volume name | unit function
------ | ------------ | ---- | ------------
89 | /scale4.2p/datalib/scale.rev05.sclib | | standard composition library
81 | /scale4.2p/datalib/scale.rev02.xn16 | | cross section library
11 | ft11f001 | | short cross section library

standard composition library data
---------------------------------

unit number  : 89

dataset name : /scale4.2p/datalib/scale.rev05.sclib

library title: scale-4 standard composition library
               386 standard compositions, 332 nuclides
               10 elements with variable isotopic distrbutions.

creation date:  1/25/94

cross section library data
--------------------------

unit number  : 81

dataset name : /scale4.2p/datalib/scale.rev02.xn16

library title: scale 4 - 16 neutron group criticality safety library
               hansen-roach data with knight modifications and some endf/b 4 data
               compiled for nrc       1/27/89
               last updated          10/12/89
               l.m.petrie            ornl

**** xsdrn mesh intervals ****
```

```
 4 mesh intervals in zone  1
 4 mesh intervals in zone  2
14 mesh intervals in zone  3

                    .........   0 io's were used before reading keno v data     .........

                    .........   0 io's were used reading the keno v parameter data     .........

                   *************** data reading completed ***************

                    .........   0 io's were used preparing the keno v input data     .........

                    .........   0 io's were used loading the keno v data     .........

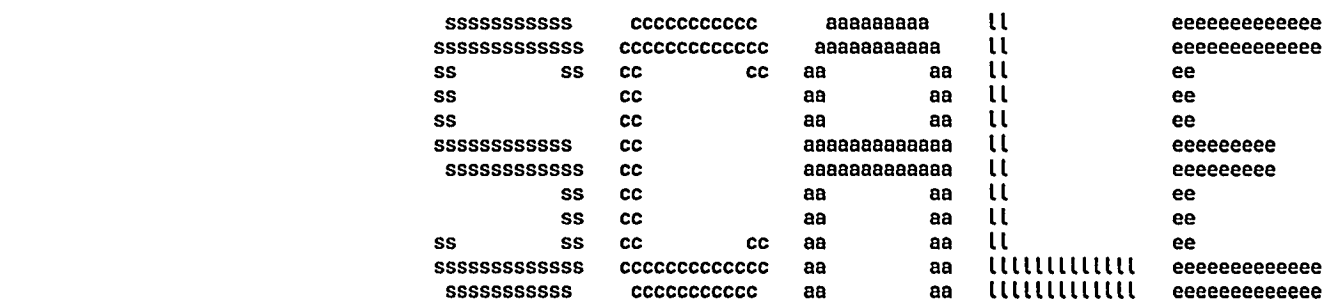
                    .........   0 io's were used loading the data     .........

                    .........   0 io's were used checking the keno v geometry data     .........
                   *****  restart data has been written on unit 95  *****

                    .........   0 io's were used writing the keno v - csas data     .........

                    .........   0 io's were used before reading search data     .........

                    .........   0 io's were used processing csas input data     .........
```




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  modify                                       *****
*****                                                                     *****
*****        creation date:  10/25/93                                     *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  modify                                       *****
*****                                                                     *****
*****              version:  2.1                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:43:11                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
*********************************************************************************************************************
***                                                                                                               ***
***                                  sample square fuel cask                                                      ***
***                                                                                                               ***
*********************************************************************************************************************
*********************************************************************************************************************
***                                  ******          search data          ******                                  ***
***                                                                                                               ***
***                                                                                                               ***
***                                  search type                          optimum pitch                           ***
***                                                                                                               ***
***                        pas       number of search passes                  10                                 ***
***                                                                                                               ***
***                        npm       number of search parameters               1                                 ***
***                                                                                                               ***
***                        eps       search convergence tolerance           .0050                                ***
***                        -con      minimum constraint                    -.0417                                ***
***                                                                                                               ***
***                        +con      maximum constraint                     .2083                                ***
***                                                                                                               ***
***                                                                                                               ***
***                          unit    first   last                                                               ***
***      command             number  region  region  ----------------    search constants    ----------------   ***
***                                                                                                               ***
***                          unit    reg=__  to __   +x or r  -x or +h  +y or -h     -y        +z        -z     ***
***                                                                                                               ***
***  alter/change/modify      1       3       3       1.00      1.00      1.00      1.00       .00       .00    ***
***                                                                                                               ***
*********************************************************************************************************************
*********************************************************************************************************************
```

```
*******************   search pass  1     keff=  7.11079E-01 + or -  5.77865E-03   *******************
                          the parameter was  0.00000E+00
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  2     keff=  7.62379E-01 + or -  4.31515E-03   *******************
                          the parameter was -4.16667E-02
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  3     keff=  6.20806E-01 + or -  5.39177E-03   *******************
                          the parameter was  2.08333E-01
                  *****  modified keno v data has been rewritten on unit 95  *****

*******************   search pass  4     keff=  6.43165E-01 + or -  4.71612E-03   *******************
                          the parameter was  1.04167E-01
```

sample square fuel cask

convergence was achieved on pass 2 the parameter was -4.16667E-02

the equation used in the search was:

k-eff = +7.11079E-01 -1.03319E+00*p +4.44000E+00*p**2 -7.49067E+00*p**3

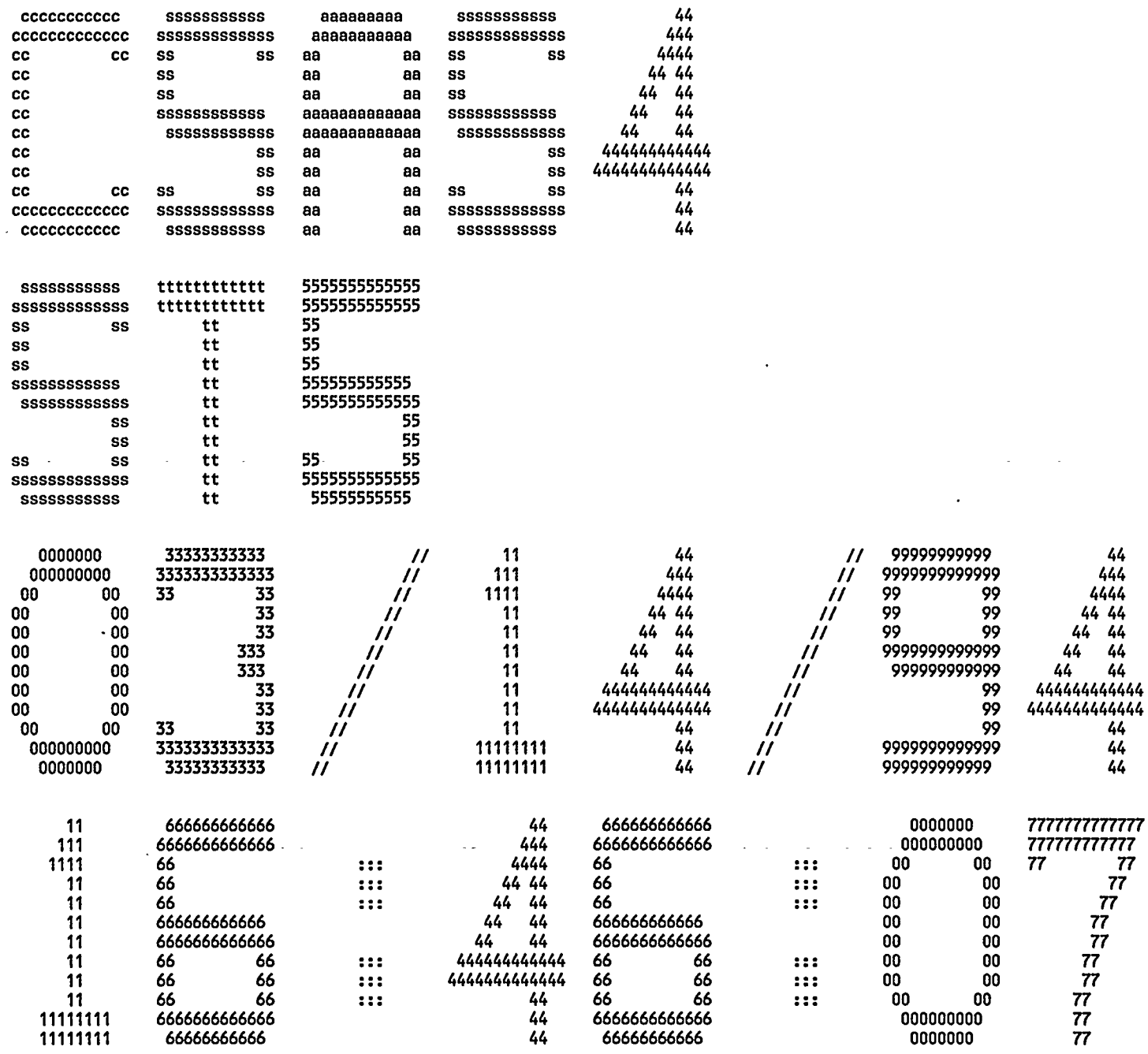
k-effective= 7.62379E-01 + or - 4.31515E-03 the corresponding geometry follows;

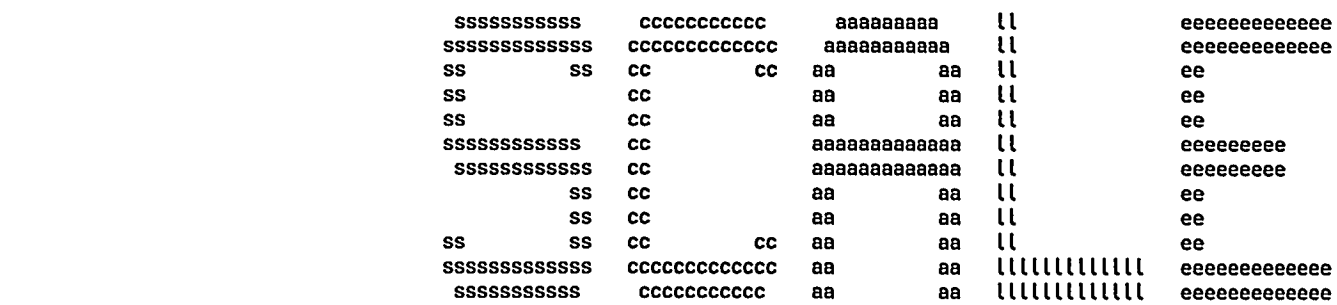
| region | media num | bias id | geometry description for those units utilized in this problem |
|---|---|---|---|
| | | | ----- unit 1 ----- |
| 1 cuboid | 500 | 1 | +x = 10.837 -x = -10.837 +y = 10.837 -y = -10.837 +z = 183.00 -z = -183.00 |
| 2 cuboid | 4 | 1 | +x = 11.500 -x = -11.500 +y = 11.500 -y = -11.500 +z = 183.00 -z = -183.00 |
| 3 cuboid | 5 | 1 | +x = 11.500 -x = -11.500 +y = 11.500 -y = -11.500 +z = 183.00 -z = -183.00 |

***** modified keno v data has been rewritten on unit 95 *****

***** convergence has been achieved in the search package. *****

control module modify is complete.


CCCCCCCCCCC SSSSSSSSSS AAAAAAAAA SSSSSSSSSS 44
CCCCCCCCCCCCC SSSSSSSSSSSS AAAAAAAAAAA SSSSSSSSSSSS 444
CC CC SS SS AA AA SS SS 4444
CC SS AA AA SS 44 44
CC SS AA AA SS 44 44
CC SSSSSSSSSSS AAAAAAAAAAAAA SSSSSSSSSSS 44 44
CC SSSSSSSSSSS AAAAAAAAAAAAA SSSSSSSSSSS 44 44
CC SS AA AA SS 444444444444
CC SS AA AA SS 4444444444444
CC CC SS SS AA AA SS SS 44
CCCCCCCCCCCCC SSSSSSSSSSSS AA AA SSSSSSSSSSSS 44
CCCCCCCCCCC SSSSSSSSSS AA AA SSSSSSSSSS 44
SSSSSSSSSS TTTTTTTTTTTT 5555555555555
SSSSSSSSSSSS TTTTTTTTTTTT 5555555555555
SS SS TT 55
SS TT 55
SS TT 55
SSSSSSSSSSS TT 555555555555
SSSSSSSSSSS TT 5555555555555
SS TT 55
SS TT 55
SS SS TT 55 55
SSSSSSSSSSSS TT 5555555555555
SSSSSSSSSS TT 55555555555
0000000 33333333333 // 11 44 // 99999999999 44
000000000 3333333333333 // 111 444 // 9999999999999 444
00 00 33 33 // 1111 4444 // 99 99 4444
00 00 33 // 11 44 44 // 99 99 44 44
00 00 33 // 11 44 44 // 99 99 44 44
00 00 333 // 11 44 44 // 9999999999999 44 44
00 00 333 // 11 44 44 // 999999999999 44 44
00 00 33 // 11 444444444444 // 99 444444444444
00 00 33 // 11 4444444444444 // 99 4444444444444
00 00 33 33 // 11 44 // 99 44
000000000 3333333333333 // 11111111 44 // 9999999999999 44
0000000 33333333333 // 11111111 44 // 999999999999 44
11 666666666666 44 666666666666 0000000 7777777777777
111 6666666666666 444 6666666666666 000000000 777777777777
1111 66 ::: 4444 66 ::: 00 00 77 77
11 66 ::: 44 44 66 ::: 00 00 77
11 66 ::: 44 44 66 ::: 00 00 77
11 666666666666 44 44 666666666666 00 00 77
11 6666666666666 44 44 6666666666666 00 00 77
11 66 66 ::: 444444444444 66 66 ::: 00 00 77
11 66 66 ::: 4444444444444 66 66 ::: 00 00 77
11 66 66 ::: 44 66 66 ::: 00 00 77
11111111 6666666666666 44 6666666666666 000000000 77
11111111 66666666666 44 66666666666 0000000 77




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  csas                                         *****
*****                                                                     *****
*****         creation date:  12/01/93                                    *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  csas                                         *****
*****                                                                     *****
*****              version:  2.2                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:46:07                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
sample problem 4  set up 2c8 in csas25


**** problem parameters ****

lib hansen-roach library
mxx            1 mixtures
msc            1 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           0 fuel solutions


**** problem composition description ****

sc  uranium      standard composition
mx             1 mixture no.
vf         .9850 volume fraction
roth     19.0500 theoretical density
temp       293.0 deg kelvin
           92235  93.20%
           92238   5.60%
           92234   1.00%
           92236    .20%
end


**** problem geometry ****


**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

```
                                sample problem 4  set up 2c8 in csas25

                                ********** data library information **********

        unit                                                          volume
        number              data set name                             name        unit function
        ------              -------------                             ----        -------------

          89      /scale4.2p/datalib/scale.rev05.sclib                            standard composition library

          81      /scale4.2p/datalib/scale.rev02.xn16                             cross section library

          11      ft11f001                                                        short cross section library

                                    standard composition library data
                                    ---------------------------------

          unit number  :  89

          dataset name :  /scale4.2p/datalib/scale.rev05.sclib

          library title:  scale-4 standard composition library
                          386 standard compositions, 332 nuclides
                          10 elements with variable isotopic distrbutions.

          creation date:   1/25/94

                                     cross section library data
                                    --------------------------

          unit number  :  81

          dataset name :  /scale4.2p/datalib/scale.rev02.xn16

          library title:  scale 4 - 16 neutron group criticality safety library
                            hansen-roach data with knight modifications and some endf/b 4 data
                              compiled for nrc       1/27/89
                                last updated        10/12/89
                                  l.m.petrie       ornl

                .....    0 io's were used before reading keno v data    .....
```

```
 ........    0 io's were used reading the keno v parameter data    ........

*************** data reading completed ***************

 ........    0 io's were used preparing the keno v input data    ........

 ........    0 io's were used loading the keno v data    ........

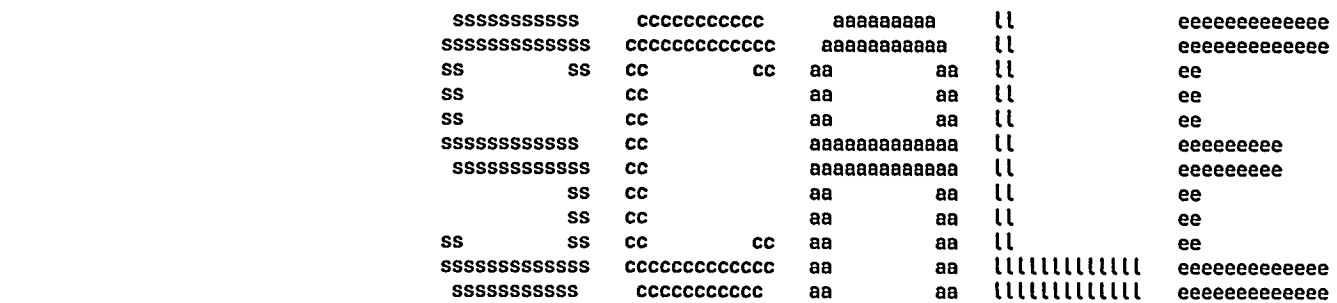
 ........    0 io's were used loading the data    ........

 ........    0 io's were used checking the keno v geometry data    ........
 *****  restart data has been written on unit 95  *****

 ........    0 io's were used writing the keno v - csas data    ........

 ........    0 io's were used before reading search data    ........

 ........    0 io's were used processing csas input data    ........
```




```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.2           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  modify                                       *****
*****                                                                     *****
*****        creation date:  10/25/93                                     *****
*****                                                                     *****
*****              library:  /scale4.2p/exe                               *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  modify                                       *****
*****                                                                     *****
*****              version:  2.1                                          *****
*****                                                                     *****
*****              jobname:  st5                                          *****
*****                                                                     *****
*****    date of execution:  03/14/94                                     *****
*****                                                                     *****
*****    time of execution:  16:46:26                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
*****************************************************************************************************************
***                              critical pitch search for case 2c8 bare                                      ***
*****************************************************************************************************************
*****************************************************************************************************************
***                                 ******          search data          ******                               ***
***                                                                                                           ***
***                                 search type                          critical pitch                       ***
***                                                                                                           ***
***                        pas      number of search passes                 10                               ***
***                                                                                                           ***
***                        npm      number of search parameters              1                               ***
***                                                                                                           ***
***                        eps      search convergence tolerance           .0050                             ***
***                                                                                                           ***
***                        kef      desired k-effective                   1.0000                             ***
***                                                                                                           ***
***                        -con     minimum constraint                    -.2070                             ***
***                                                                                                           ***
***                        +con     maximum constraint                     .0693                             ***
***                                                                                                           ***
***                          unit    first   last                                                           ***
***       command            number  region  region   ----------------    search constants    ----------------  ***
***                                                                                                           ***
***                          unit    reg=__  to__    +x or r   -x or +h  +y or -h     -y       +z       -z     ***
***                                                                                                           ***
***  alter/change/modify       1       2       2       1.00      1.00      1.00      1.00       .00      .00    ***
***                                                                                                           ***
***                                                                                                           ***
***  alter/change/modify       1       2       2        .00       .00       .00       .00      1.00     1.00    ***
*****************************************************************************************************************
*****************************************************************************************************************

     ********************   search pass  1     keff=  9.62581E-01 + or -  4.66186E-03   ********************
                                the parameter was  0.00000E+00
                      *****  modified keno v data has been rewritten on unit 95  *****

     ********************   search pass  2     keff=  9.26968E-01 + or -  4.76267E-03   ********************
                                the parameter was  6.92605E-02
                      *****  modified keno v data has been rewritten on unit 95  *****

     ********************   search pass  3     keff=  1.01030E+00 + or -  4.65530E-03   ********************
                                the parameter was -7.27719E-02
                      *****  modified keno v data has been rewritten on unit 95  *****

     ********************   search pass  4     keff=  1.00968E+00 + or -  4.18861E-03   ********************
                                the parameter was -5.70604E-02
                      *****  modified keno v data has been rewritten on unit 95  *****

     ********************   search pass  5     keff=  9.80129E-01 + or -  5.04216E-03   ********************
                                the parameter was -3.87869E-02
                      *****  modified keno v data has been rewritten on unit 95  *****

     ********************   search pass  6     keff=  9.98210E-01 + or -  4.21917E-03   ********************
                                the parameter was -5.09593E-02
                           critical pitch search for case 2c8 bare
```

```
**********************************************************************************************************************************
**********************************************************************************************************************************

                        convergence was achieved on pass  6  the parameter was  -5.09593E-02

                        the equation used in the search was:

                        k-eff = +9.39989E-01 -9.76969E-01*p +7.86001E+00*p**2 +8.33712E+01*p**3

                        k-effective=  9.98210E-01 + or -  4.21917E-03   the corresponding geometry follows;

region          media bias     geometry description for those units utilized in this problem
                 num   id

                                                  -----   unit     1   -----

  1 cylinder         1   1  radius =  5.7480     +z =  5.3825     -z = -5.3825     centerline is at  x = 0.00000E+00  y = 0.00000E+00

  2 cuboid           0   1      +x =  6.8786     -x = -6.8786     +y =  6.8786     -y = -6.8786     +z =  6.5318     -z = -6.5318

**********************************************************************************************************************************
**********************************************************************************************************************************
```

critical pitch search for case 2c8 bare

based on the preceding data, the best estimate of the parameter is -5.17118E-02

the geometry corresponding to this parameter follows:

| region | media num | bias id | geometry description for those units utilized in this problem |
|---|---|---|---|
| | | | ----- unit 1 ----- |
| 1 cylinder | 1 | 1 | radius = 5.7480 +z = 5.3825 -z = -5.3825 centerline is at x = 0.00000E+00 y = 0.00000E+00 |
| 2 cuboid | 0 | 1 | +x = 6.8732 -x = -6.8732 +y = 6.8732 -y = -6.8732 +z = 6.5266 -z = -6.5266 |

**********
**********

***** modified keno v data has been rewritten on unit 95 *****

***** convergence has been achieved in the search package. *****

control module modify is complete.

NUREG/CR-0200
Revision 5
Volume 1, Section C6
ORNL/NUREG/CSD-2/V1/R5

Computational Physics and Engineering Division

# CSAS6: CONTROL MODULE FOR ENHANCED CRITICALITY SAFETY ANALYSIS WITH KENO VI

D. F. Hollenbach
L. M. Petrie

Date Published: March 1997

Prepared for the
Office of Nuclear Material Safety and Safeguards
U. S. Nuclear Regulatory Commission
Washington, DC 20555
Under Interagency Agreement DOE 1886-C000-9B
NRC JCN No. B0009

Prepared by the
OAK RIDGE NATIONAL LABORATORY
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORP.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-96OR22464

## ABSTRACT

The Criticality Safety Analysis Sequences (CSAS) were developed within the SCALE code system to provide automated, problem-dependent, cross-section processing followed by calculation of the neutron multiplication factor for the system being modeled. The current version of CSAS decides which sequence to run based on the first record in the problem description. If the sequence of modules to be run includes KENO V.a, the problem geometry is read and checked in MIPLIB which is part of CSAS. Since KENO VI has a different geometry package than KENO-V.a, the current version of CSAS is not compatible with KENO VI. This document describes a version of CSAS, named CSAS6, specifically written for KENO VI.

CSAS6 contains two control sequences, CSAS26 and CSAS26X. CSAS26 activates the cross-section processing codes BONAMI and NITAWL-II to provide resonance-corrected cross sections for use in KENO VI. CSAS26X, in addition to activating BONAMI and NITAWL-II, calls XSDRNPM to provide cell- weighted cross sections. The enhanced geometric modeling capabilities available in KENO VI, coupled with the automated cross-section processing within the control sequences, allow complex, three-dimensional systems to be easily analyzed.

## CONTENTS


Page

ABSTRACT ..... C6.iii
LIST OF FIGURES ..... C6.vii
LIST OF TABLES ..... C6.ix
ACKNOWLEDGMENTS ..... C6.xi

C6.1 INTRODUCTION ..... C6.1.1

C6.2 TECHNIQUES ..... C6.2.1
C6.2.1 CSAS6 LIMITATIONS ..... C6.2.1
C6.2.2 REFERENCES ..... C6.2.2

C6.3 LOGICAL PROGRAM FLOW ..... C6.3.1
C6.3.1 INITIATION OF CSAS6 ..... C6.3.1
C6.3.2 OVERALL FLOW DIAGRAM ..... C6.3.5
C6.3.3 THE MATERIAL INFORMATION PROCESSOR ..... C6.3.8
C6.3.4 KENO VI DATA PROCESSOR ..... C6.3.8
C6.3.4.1 Control of the Data Input for KENO VI ..... C6.3.11
C6.3.4.2 Read KENO VI Array Data ..... C6.3.14
C6.3.4.3 Read KENO VI Geometry Data ..... C6.3.15
C6.3.4.4 Read KENO VI Mixing Table Data ..... C6.3.18
C6.3.4.5 Read KENO VI Plot Data ..... C6.3.19
C6.3.4.6 Read KENO VI Biasing Data ..... C6.3.20
C6.3.4.7 Read KENO VI Restart Data ..... C6.3.21
C6.3.4.8 Generate Remaining KENO VI Data ..... C6.3.22
C6.3.4.9 Process the KENO VI Geometry Data ..... C6.3.25
C6.3.4.10 Load KENO VI Data from the Direct-Access File ..... C6.3.26
C6.3.4.11 Calculate KENO VI Geometry Volumes ..... C6.3.27
C6.3.4.12 Generate Printer Plot ..... C6.3.28
C6.3.4.13 Write KENO VI Input Data on Restart File ..... C6.3.30

C6.4 INPUT DATA GUIDE ..... C6.4.1
C6.4.1 INPUT DATA SUMMARY FOR EXPERIENCED USERS ..... C6.4.3
C6.4.2 INPUT DATA REQUIREMENTS AND DATA CHECKING ..... C6.4.21
C6.4.3 MATERIAL INFORMATION PROCESSOR DATA ..... C6.4.22
C6.4.4 STANDARD COMPOSITIONS SPECIFICATION DATA ..... C6.4.24
C6.4.5 OPTIONAL UNIT CELL SPECIFICATION FOR INFINITE HOMOGENEOUS PROBLEMS ..... C6.4.31
C6.4.6 UNIT CELL SPECIFICATION FOR LATTICECELL PROBLEMS ..... C6.4.32
C6.4.7 UNIT CELL SPECIFICATION FOR MULTIREGION PROBLEMS ..... C6.4.39
C6.4.8 OPTIONAL PARAMETER DATA ..... C6.4.42
C6.4.9 SUMMARY OF KENO VI DATA ..... C6.4.47

C6.5 NOTES FOR CSAS6 USERS ..... C6.5.1
C6.5.1 CSAS6 MODULES ..... C6.5.1
C6.5.2 AUTOMATIC CROSS-SECTION PROCESSING FOR CRITICALITY CALCULATIONS ..... C6.5.4
C6.5.2.1 Run KENO-VI using CSAS26 ..... C6.5.4
C6.5.2.2 Run KENO-VI using CSAS26X ..... C6.5.6
C6.5.2.3 Run XSDRNPM using CSAS1X ..... C6.5.7
C6.5.3 AUTOMATED CROSS-SECTION PREPARATION FOR USE BY SUBSEQUENT CODES ..... C6.5.8
C6.5.3.1 Create a Microscopic, AMPX Working Format, Cross-Section Library for Use by KENO-VI ..... C6.5.8
C6.5.3.2 Create a Microscopic, Cell-Weighted AMPX Working Format, Cross-Section Library ..... C6.5.10
C6.5.3.3 Create a Macroscopic, AMPX Working Format, Cross-Section Library and a Macroscopic, MORSE/KENO Cross-Section Library .... C6.5.15
C6.5.3.4 Create a Macroscopic, Cell-Weighted, AMPX Working Format, Library and a Macroscopic, Cell-Weighted, MORSE/KENO Library ... C6.5.19
C6.5.4 CREATE AND USE TWO CELL-WEIGHTED MIXTURES IN A PROBLEM ..... C6.5.22
C6.5.5 CREATE A CELL-WEIGHTED MIXTURE REPRESENTING A NONHOMOGENEOUS FUEL ASSEMBLY ..... C6.5.26

C6.6 DESCRIPTION OF OUTPUT ..... C6.6.1
C6.6.1 CONTROL MODULE HEADER PAGE ..... C6.6.1
C6.6.2 CONTROL MODULE PROGRAM VERIFICATION TABLE ..... C6.6.2
C6.6.3 MATERIAL INFORMATION PROCESSOR OUTPUT ..... C6.6.3
C6.6.4 KENO VI OUTPUT ..... C6.6.4

C6.7 WARNING AND ERROR MESSAGES ..... C6.7.1
C6.7.1 MATERIAL INFORMATION PROCESSOR MESSAGES ..... C6.7.1
C6.7.2 MESSAGES FROM CRITICALITY SAFETY SEQUENCE NO. 6 ..... C6.7.19
C6.7.3 STOP CODES ..... C6.7.25
C6.7.4 MESSAGES ASSOCIATED WITH STOP 20 IN CSAS ..... C6.7.27
C6.7.5 UNNUMBERED MESSAGES ..... C6.7.28

C6.A ALPHABETICAL INDEX OF SUBROUTINES ..... C6.A.1
C6.B SAMPLE PROBLEMS ..... C6.B.1
C6.B.1 REFERENCES ..... C6.B.10
C6.C SAMPLE OUTPUT ..... C6.C.1

## LIST OF FIGURES


Page

C6.3.1 Control modules available in CSAS6 ..... C6.3.4

C6.3.2 Overall flow of CSAS6 ..... C6.3.5

C6.3.3 Flowchart of KENO VI data processor ..... C6.3.9

C6.3.4 Flowchart of KENO VI data reading routines ..... C6.3.12

C6.3.5 Flowchart for reading KENO VI array data ..... C6.3.14

C6.3.6 Flowchart of routines to read the KENO VI geometry data ..... C6.3.15

C6.3.7 Flowchart for reading KENO VI geometric shape data ..... C6.3.16

C6.3.8 Flowchart for reading KENO VI mixing table data ..... C6.3.19

C6.3.9 Flowchart for reading KENO VI plot data ..... C6.3.20

C6.3.10 Flowchart for reading KENO VI biasing data ..... C6.3.21

C6.3.11 Flowchart for reading KENO VI restart data ..... C6.3.22

C6.3.12 Flowchart for generating remaining KENO VI data ..... C6.3.23

C6.3.13 Flowchart for processing KENO VI geometry data ..... C6.3.25

C6.3.14 Flowchart for loading data from direct access ..... C6.3.26

C6.3.15 Flowchart for calculating KENO VI geometry volumes ..... C6.3.27

C6.3.16 Flowchart for printer plots ..... C6.3.29

C6.3.17 Flowchart for writing data on the restart data file ..... C6.3.30

C6.4.1 Arrangement of materials in a SQUAREPITCH and SPHSQUAREP unit cell ..... C6.4.35

C6.4.2 Arrangement of materials in a TRIANGPITCH and SPHTRIANGP unit cell ..... C6.4.35

C6.4.3 Arrangement of materials in a SYMMSLABCELL unit cell having reflected left and right boundary conditions ..... C6.4.36

C6.4.4 Arrangement of materials in an ASQUAREPITCH or ASPHSQUAREP unit cell ..... C6.4.37

C6.4.5 Arrangement of materials in an ATRIANGPITCH or ASPHTRIANGP unit cell ..... C6.4.37

C6.4.6 Arrangement of materials in an ASYMSLABCELL unit cell ..... C6.4.38

C6.5.1 Mixing table from CSASN with PARM=CHECK ..... C6.5.9

C6.5.2 Mixing table from CSASN edited to KENO-VI format ..... C6.5.9

C6.5.3 Mixing table from CSASN edited to KENO-VI format ..... C6.5.10

C6.5.4 Second mixing table from CSASIX ..... C6.5.11

C6.5.5 Mixing table from CSASIX edited to KENO VI format ..... C6.5.12

C6.5.6 Mixing tables from CSASIX ..... C6.5.13

C6.5.7 KENO VI mixing table using all CSASIX entries ..... C6.5.14

C6.5.8 KENO VI mixing table using only mixtures 4, 5 and 6 from CSASIX ..... C6.5.14

C6.5.9 Condensed KENO VI mixing table using only mixtures 4, 5, and 6 from CSASIX ..... C6.5.15

C6.5.10 1.98% enriched $UO_2$ fuel assembly ..... C6.5.27

C6.6.1 Sample control module header page ..... C6.6.2

C6.6.2 Sample control module program verification table ..... C6.6.3

C6.6.3 Part 1 of Material Information Processor printout ..... C6.6.3

C6.6.4 Part 2 of Material Information Processor printout ..... C6.6.4

C6.6.5 Summary of KENO VI I/O usage by CSAS control module ..... C6.6.5

C6.B.1 Critical assembly of $UO_2F_2$ solution in a 30°-Y aluminum pipe ..... C6.B.3

C6.B.2 Critical assembly of $UO_2F_2$ solution in a plexiglass cross ..... C6.B.5

C6.B.3 Grotesque experimental setup ..... C6.B.8

## LIST OF TABLES


| | | Page |
|---|---|---|
| C6.3.1 | Description of units utilized by the CSAS6 control module | C6.3.2 |
| C6.3.2 | Description of units used by the functional modules of CSAS6 | C6.3.3 |
| C6.4.1 | Functional modules executed by CSAS6 control modules | C6.4.2 |
| C6.4.2 | I/O units on which cross-section libraries are written | C6.4.2 |
| C6.4.3 | I/O units generated by CSAS6 and associated modules | C6.4.2 |
| C6.4.4 | Input data requirements | C6.4.4 |
| C6.4.5 | Summary of KENO VI input data | C6.4.13 |
| C6.4.7 | Summary of data input requirements | C6.4.21 |
| C6.4.8 | Outline of Material Information Processor data | C6.4.23 |
| C6.4.9 | Outline of standard composition specification data | C6.4.25 |
| C6.4.10 | Optional unit cell specification for INFHOMMEDIUM problems | C6.4.32 |
| C6.4.11 | Unit cell specification for LATTICECELL problems | C6.4.34 |
| C6.4.12 | Geometry specification for MULTIREGION problems | C6.4.40 |
| C6.4.13 | Optional parameter data | C6.4.43 |
| C6.4.14 | Outline of KENO VI data | C6.4.48 |
| C6.4.15 | Summary of KENO VI data | C6.4.49 |
| C6.5.1 | Description of units utilized by the Criticality Safety Analysis Sequence No. 6 control module | C6.5.2 |
| C6.5.2 | Description of units used by the functional modules of CSAS6 | C6.5.3 |
| C6.6.1 | Location of functional module output | C6.6.1 |
| C6.A.1 | Alphabetical index of subroutines | C6.A.2 |
| C6.A.2 | Alphabetical index of MODIFY routines | C6.A.32 |

C6.A.3 Commons used in CSAS6 ..... C6.A.36

C6.A.4 Commons used in MODIFY ..... C6.A.48

## ACKNOWLEDGMENTS

The CSAS6 Criticality Safety Analysis Sequence is based on the CSAS4 control module, described in Sect. C4, and the KENO VI functional module, described in Sect. F17. Therefore, special acknowledgment is made to J. A. Bucholz, R. M. Westfall, and J. R. Knight who developed CSAS2, which is the precursor to CSAS4. G. E. Whitesides is acknowledged for his contributions through early versions of KENO. Appreciation is expressed to C. V. Parks for his guidance in developing CSAS.

## C6.1 INTRODUCTION

Within the life cycle of the SCALE system, the CSAS program has superseded earlier programs that contained criticality safety analysis sequences. Embedded within the CSAS program are numerous sequences that enable automated cross-section processing and criticality analyses. Currently, all the criticality analysis sequences (sometimes referred to as modules) within SCALE are contained within the CSAS program. Unfortunately, due to the conflicting geometry packages in KENO V.a and KENO VI, the inclusion of a criticality analysis sequence within CSAS that calls KENO VI is not practical. Instead, a new criticality safety analysis sequence has been developed solely for KENO VI.

Criticality Safety Analysis Sequence No. 6 (CSAS6) was developed to provide automated cross-section processing for KENO VI in the SCALE system. Because it includes all the options available in earlier SCALE criticality sequences, except searches, and is designed for use with KENO VI only, the program is called by the name CSAS6.

The standardized automated procedures process SCALE cross sections using the Bondarenko method (via BONAMI) and the Nordheim integral method (via NITAWL-II) to provide a resonance-corrected cross-section library based on the physical characteristics of the problem being analyzed. This cross-section library is then utilized by KENO VI, a three-dimensional (3-D) multigroup Monte Carlo criticality program.

The user may select the CSAS26 or the CSAS26X sequence in CSAS6. An X in the module name specifies that XSDRNPM will be used to provide cell-weighted cross sections. CSAS26X should be used when cell-weighted cross sections are used to provide a homogeneous representation of the fuel pins in the KENO VI geometry. Cell-weighting is accomplished by describing a unit cell, as in a fuel assembly, for which a one-dimensional (1-D) eigenvalue calculation of the unit cell is made to determine the spatially dependent flux spectrum. This flux spectrum is used to cell-weight the microscopic cross-section data. The nuclide number densities in the unit cell are homogenized, and the resultant homogenized cell-weighted cross sections are identified as mixture 500 and utilized in the KENO VI geometry. Note that only one mixture 500 can be created per problem.

The codes utilized in CSAS6 start with an AMPX master format cross-section library and generate a self-shielded, group-averaged library applicable to the specific problem configuration. These cross sections are then used in the KENO VI Monte Carlo code to determine the effective neutron multiplication factor. The codes and their functions are given below.

1. BONAMI — performs resonance self-shielding calculations for nuclides that have Bondarenko data associated with their cross sections.

2. NITAWL-II — applies a Nordheim resonance self-shielding correction to nuclides having resonance parameters and produces a working problem-dependent cross-section library.

3. KENO VI — calculates k-effective of a 3-D system using the Monte Carlo method.

## C6.2 TECHNIQUES

CSAS6 is designed to prepare a resonance-corrected cross-section library for subsequent use in KENO VI. In order to minimize human error, the SCALE data handling is automated as much as possible. CSAS6 applies a standardized procedure to provide appropriate cross sections for the calculation. This procedure is carried out by the Material Information Processor that generates number densities and related information, prepares geometry data for resonance self-shielding and flux-weighting cell calculations, and creates data input files for the cross-section processing codes. After reading the problem input and checking as much as possible, the control sequence executes BONAMI and NITAWL-II to prepare a resonance-corrected microscopic cross-section library in the AMPX working library format. If cell-weighted cross sections are requested, XSDRNPM performs the necessary calculations and produces a cell-weighted microscopic cross-section library in the AMPX working library format. KENO VI is then executed to calculate the k-effective or neutron multiplication factor using the cross-section library that was prepared by the control sequence.

The Material Information Processor is described in Sect. M7. It is responsible for reading the standard composition data and other engineering-type specifications, including volume fraction or percent theoretical density, temperature, and isotopic distribution. The techniques used in the Material Information Processor and their applications and limitations are discussed in Sect. M7.2. The input data for the Material Information Processor is the same as that used for CSAS4. For convenience, the Material Information Processor input data guide is included in Sect. C6.4.

The Material Information Processor checks the input data pertaining to cross-section preparation and prepares binary input files for the applicable functional modules. Depending on the specified sequence, data are prepared for BONAMI, NITAWL-II, XSDRNPM, and ICE as appropriate.

The Bondarenko method is discussed in the Material Information Processor document, Sect. M7.2.5.1, and BONAMI is described in Sect. F1. The Nordheim method and NITAWL-II are described in Sects. M7.2.5.2 and F2, respectively. The Dancoff treatment is discussed in Sect. M7.2.5.3. The automatic mesh generator, the automatic quadrature generator, and the convergence criteria for control sequences that execute XSDRNPM are described in Sects. M7.2.5.6, M7.2.5.7, and M7.2.5.8, respectively. XSDRNPM is described in Sect. F3, ICE is described in Sect. F8, and KENO VI is described in Sect. F17. See Sect. M7A for an explanation of many of the techniques used in processing the cross sections.

### C6.2.1 CSAS6 LIMITATIONS

The CSAS6 control module was developed to use simple input data and prepare problem-dependent cross sections for use in calculating the effective neutron multiplication factor of a 3-D system using KENO VI and possibly XSDRNPM. An attempt was made to make the system as general as possible within the constraints of the standardized methods chosen to be used in SCALE. Standardized methods of data input were adopted to allow easy data entry and for quality assurance purposes. Some of the limitations of the CSAS6 sequences are due to assumptions in the Nordheim integral treatment[1] as implemented in CSAS6. Inherent limitations in the NITAWL-II Nordheim integral treatment of resonance nuclides are as follows:

1. The treatment assumes no resonance overlap from other resonances or other material regions.

2. The treatment of an external moderator assumes an asymptotic flux present at the absorber-moderator interface.

3. The treatment of spatial transport uses the first-flight escape probability for the absorber, the two-region reciprocity theorem, and Dancoff factors.[2]

In many cases, the Nordheim integral treatment is an excellent method for processing resonance cross sections for use in the analysis of light-water reactor (LWR) fuel pin cells and for many other applications commonly used in neutronics analysis. However, caution should be used in applications involving several isotopes with large resonances or complicated absorber-moderator geometries. Another area of probable inadequacy involves the analysis of machinery for dissolving spent fuel elements. Such a system usually requires the simultaneous consideration of resonance absorbers located in both the solid fuel pellets and the acid solution, thereby resulting in an inadequate treatment by NITAWL-II.

The major limitations of the analytical model used by the Nordheim integral treatment in CSAS6 are as follows:

1. A lattice system whose fuel or moderator contains an absorber that has rapidly varying cross sections across the resonance region may be inadequately treated. Examples include gadolinium, silver, indium, and hafnium.

2. The presence of more than one resonance isotope in a material can lead to an effect called resonance overlap, which can result in enhanced or reduced group-averaged cross sections, depending on the predominance of resonance scattering or absorption and the relative locations of the resonances within the energy groups. The resonance overlap effect is discussed in more detail in Sect. M7.A.7.

3. Resonance interference between two different media is not correctly treated.

4. Fissile lumps in fissile solutions are not correctly treated.

5. The determination of parameters used in processing the cross sections are derived from the geometry of the cell description. CSAS6 does not allow multiple cell descriptions.

6. It is strongly recommended that CSAS calculations be made of benchmark experiments similar to the problem being calculated in order to demonstrate the validity of the code/cross-section combination for that type of problem.

## C6.2.2 REFERENCES


1. L. W. Nordheim, "The Theory of Resonance Absorption," *Proceedings of Symposia in Applied Mathematics*, Vol XI, p. 58, G. Birkhoff and E. P. Wigner, Eds., Am. Math. Soc. (1961).

2. S. M. Dancoff and M. Ginsburg, *Surface Resonance Absorption in a Close-Packed Lattice*, CP-2157, Clinton Plant, U.S. Atomic Energy Commission, September 1944.

## C6.3 LOGICAL PROGRAM FLOW

The general flow of CSAS6 is given in this section. The program flow is divided into several sections. An abbreviated representation, rather than a formal flowchart, is used to outline the program flow of each section. A flowchart may show a subroutine or library routine only once, even though it may be called multiple times in that portion of the program. The corresponding text usually states when multiple calls are made to a routine. Accompanying each abbreviated flowchart is an explanation of the purpose of that segment of the program and text describing each of the referenced subroutines.

CSAS6 has been written in a general form to allow maximum flexibility. Data are transferred from one module to another via sequential data files and direct-access data files. Table C6.3.1 lists the data files utilized by this program. CSAS6 is used to represent the control module defining the sequence. Data files utilized by the individual functional modules are included in Table C6.3.2.

The flowchart representation for CSAS6 is defined as follows:

> The SCALE DRIVER is contained in a diamond, subroutines initiating control modules are contained in circles, functional modules are contained in hexagons, major program segments are contained in rectangularized ovals, subroutine names are contained in boxes, and library routines are represented as bare names. An arrow in the flowchart indicates that the subroutine or module associated with the arrow will be discussed later in the section.

### C6.3.1 INITIATION OF CSAS6

This portion of the program is the initiation of the criticality safety analysis sequence. The SCALE DRIVER, described in Sect. M1, obtains the name of the module to be accessed from the first record (module specification record) in the user input data file. The control modules available in CSAS6 are shown in Fig. C6.3.1. These control modules use the Material Information Processor data as described in Sect. M7.3.

SCALE DRIVER allows implementation of a modular code system which includes Criticality Safety Analysis Sequence No. 6. See Sect. M1 for additional information about the driver.

CSAS26 calculates the system k-effective for 3-D problems. This module sequentially activates the functional modules BONAMI, NITAWL-II, and KENO-VI to process the required cross sections and calculate the k-effective. Note that a cell-weighted mixture cross section cannot be used.

CSAS26X calculates the system k-effective for 3-D problems that require a cell-weighted mixture. It is the same as CSAS26 except a cell-weighted mixture cross section is always created. This control module sequentially activates the functional modules BONAMI, NITAWL-II, XSDRNPM, and KENO VI to process the required cross sections and calculate the k-effective.

CSAS1X creates a microscopic cell-weighted library in the AMPX working library format. This control module sequentially activates the functional modules BONAMI and NITAWL-II to process the necessary cross sections and calculates the system k-effective of the 1-D problem using the discrete-ordinates code XSDRNPM. This sequence has been disabled in CSAS6.

Table C6.3.1 Description of units utilized by the CSAS6 control module

| Unit No. | Type of data | Creating module | User module | Type of file |
|---|---|---|---|---|
| 5 | Input | DRIVER | CSAS6 | Sequential |
| 8 | Scratch file | CSAS6 | CSAS6 | Direct or random-access data |
| 11 | Short master cross-section library | CSAS6 | BONAMI | Sequential |
| 16 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 17 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 18 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 19 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 70 | User-supplied cross sections | | CSAS6 | AMPX master format library |
| 81 | 16-group cross sections | | CSAS6 | AMPX master format library |
| 82 | 27-group cross sections | | CSAS6 | AMPX master format library |
| 83 | 44-group cross sections | | CSAS6 | AMPX master format library |
| 84 | 238-group cross sections | | CSAS6 | AMPX master format library |
| 85 | 22 neutron-18 gamma cross sections | | CSAS6 | AMPX master format library |
| 86 | 18-gamma cross sections | | CSAS6 | AMPX master format library |
| 87 | 27-group cross sections for burnup | | CSAS6 | AMPX master format library |
| 88 | 27 neutron-18 gamma cross sections | | CSAS6 | AMPX master format library |
| 89 | Standard Composition Library | | CSAS6 | Direct or random-access data file |
| 92 | ICE input data | CSAS6 | ICE | Sequential |
| 95 | KENO-VI restart data | CSAS6 KENO-VI | KENO-VI | Sequential |
| 96 | BONAMI input data | CSAS6 | BONAMI | Sequential |
| 97 | NITAWL-II input data | CSAS6 | NITAWL-II | Sequential |
| 98 | XSDRNPM input data | CSAS6 | XSDRNPM | Sequential |
| 99 | Printed output | CSAS6 | CSAS6 | Sequential |

Table C6.3.2 Description of units used by the functional modules of CSAS6

| Unit # | Module | Action | Type of file |
|---|---|---|---|
| 1 | BONAMI | WRITE | Microscopic master working format cross sections |
| | NITAWL-II | READ | Microscopic master working format cross sections |
| 2 | ICE | WRITE | Macroscopic working format cross sections |
| 3 | XSDRNPM | WRITE | Microscopic cell-weighted working format cross sections |
| | ICE | READ | Microscopic cell-weighted working format cross sections |
| | KENO-VI | READ | Microscopic cell-weighted working format cross sections |
| 4 | NITAWL-II | WRITE | Microscopic working format cross sections |
| | ICE | READ | Microscopic working format cross sections |
| | KENO-VI | READ | Microscopic working format cross sections |
| | XSDRNPM | READ | Microscopic working format cross sections |
| 6 | BONAMI | WRITE | Printed output |
| | ICE | WRITE | Printed output |
| | KENO-VI | WRITE | Printed output |
| | NITAWL-II | WRITE | Printed output |
| | XSDRNPM | WRITE | Printed output |
| 8 | BONAMI | WRITE/READ | Direct or random-access scratch files |
| | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 9 | BONAMI | WRITE/READ | Direct or random-access scratch files |
| | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct or random-access scratch files |
| | NITAWL-II | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 10 | ICE | WRITE/READ | Direct or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct or random-access scratch files |
| 11 | BONAMI | READ | Short AMPX master format cross sections |
| 14 | ICE | WRITE | Monte Carlo-formatted macroscopic cross sections |
| | KENO-VI | READ/WRITE | Monte Carlo-formatted macroscopic cross sections |
| 16 | KENO-VI | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Angular flux file |
| 17 | XSDRNPM | WRITE/READ | Sequential scratch file |
| 18 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 19 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 79 | KENO-VI | READ | KENO albedo file |
| 80 | KENO-VI | READ | KENO weighting file |
| 92 | ICE | READ | Binary input file |
| 95 | KENO-VI | READ | Binary input file |
| 96 | BONAMI | READ | Binary input file |
| 97 | NITAWL-II | READ | Binary input file |
| 98 | XSDRNPM | READ | Binary input file |

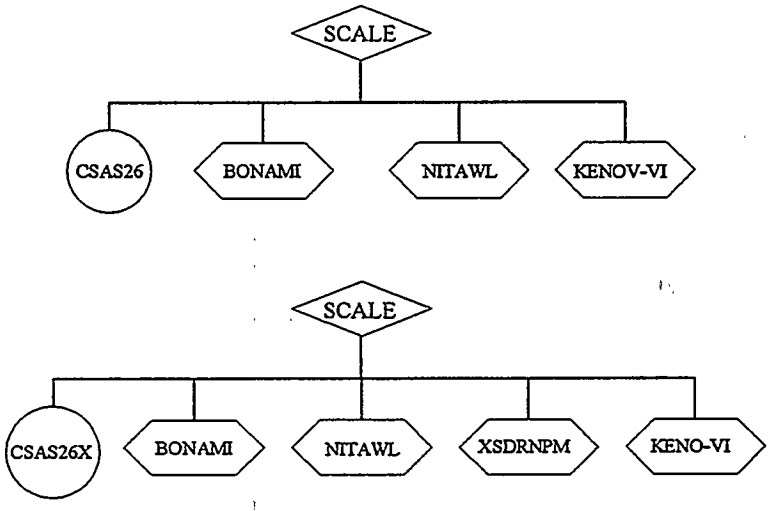



Figure C6.3.1 Control modules available in CSAS6

CSASN creates a resonance-corrected microscopic cross-section library in the AMPX working library format by sequentially activating the functional modules BONAMI and NITAWL-II. This section has been disabled in CSAS6.

CSASI provides a resonance-corrected macroscopic mixture cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, and ICE. This sequence has been disabled in CSAS6.

CSASIX provides a resonance-corrected, cell-weighted, macroscopic mixture cross-section library in the MORSE/KENO (ICE Monte Carlo-processed) format by sequentially activating the functional modules BONAMI, NITAWL-II, XSDRNPM, and ICE. The discrete-ordinates code XSDRNPM also computes the k-effective for the 1-D problem. This sequence has been disabled in CSAS6.

BONAMI is a functional module in the SCALE system that performs resonance self-shielding calculations for nuclides that contain Bondarenko data in their cross sections. See Sect. F1 for detailed information concerning BONAMI.

NITAWL-II is a functional module in the SCALE system. It applies a Nordheim resonance self-shielding correction to nuclides having resonance parameters associated with the cross-section data. See Sect. F2 for detailed information.

XSDRNPM is a functional module in the SCALE system. It provides cell-weighted cross sections based on the specified unit cell. It can also calculate the k-effective for 1-D problems. See Sect. F3 for additional information.

ICE is a functional module in the SCALE system. It creates a Monte Carlo-formatted mixed cross-section library for use by KENO V.a, KENO VI, or MORSE. See Sect. F8 for additional details.

KENO VI is a functional module in the SCALE system. It calculates the system k-effective of a 3-D problem. See Sect. F17 for a detailed description of KENO VI.

## C6.3.2 OVERALL FLOW DIAGRAM

Figure C6.3.2 illustrates the overall flow of CSAS6. The control modules CSAS26 and CSAS26X communicate with the driver and dictate the order in which other modules are to be executed.

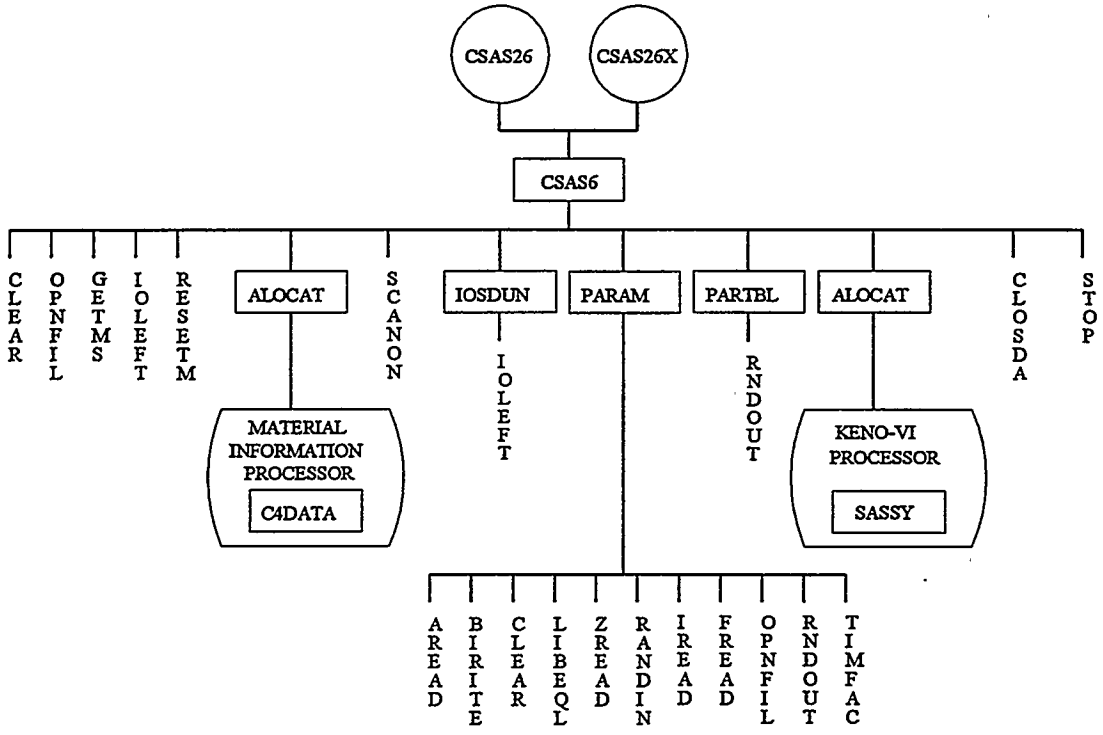



Figure C6.3.2 Overall flow of CSAS6

CSAS26 - This control module calculates k-effective for a 3-D system, using KENO VI, when a cell-weighted mixture cross section is not needed.

CSAS62X - This control module calculates k-effective for a 3-D system, using KENO VI, when a cell-weighted mixture cross section is required.

CSAS6 - This is the main program for CSAS6 and its associated control modules. GETMS is called to read the analytical sequence specification and the parameters associated with it. The sequence specification is used to set flags to cause the desired functional modules to be executed in the

proper order. IOLEFT is called to initialize the counter that keeps track of the number of I/Os that are done. INDEX is a FORTRAN-supplied intrinsic function used to determine if a variable matches a specified character string. IONUMS is called to set the input and output units for the free-form reading routines. OPNFIL is called to open the input and output units that are used by the control module. IOSDUN is called several times to determine the number of I/Os that have been used in various operations. C4DATA is called from ALOCAT with the requested storage allocation. This accesses the MATERIAL INFORMATION PROCESSOR. SCANON is called to activate the feature that allows scanning the input data for the word END. CPARAM is called to read the KENO VI title and to read and initialize the KENO VI parameter data. SASSY is called from ALOCAT to read the KENO VI data and write the information in a form that can be utilized by the CSAS6 module. If errors are encountered during reading, preparing, and checking the data, execution is terminated without activating any of the functional modules. The direct-access files are closed out when the problem is completed or terminated. This program makes use of error flags in an effort to continue reading and checking input data even after errors are found. However, some errors are severe enough to cause immediate termination.

CLEAR - This library routine is called to initialize the logical unit numbers used by CSAS.

OPNFIL - This library routine is called to open the I/O units or devices used by the specified CSAS control module and all of the functional modules it invokes.

GETMS - This assembly language library routine is called to read the analytical sequence specification and any associated parameter data. It establishes communication with the SCALE driver and enters data obtained from the SCALE driver into common. An example of an analytical sequence specification is given on the following page. This particular example specifies that the control module CSAS is to be executed. The associated parameters are entered following the word "PARM=," which must start beyond column 10. In this example the parameters specify that the data are to be checked without executing the problem, and a maximum of 200,000 words is requested to complete this task.

```
=CSAS26     PARM= (CHECK,SIZE=200000).
```

IOLEFT - This library routine is called to store the initial I/O count in common for determining the number of I/Os used in certain parts of the program.

RESETM - This library routine is an entry point in the assembly language library routine GETMS. It transmits control data back to the SCALE driver.

ALOCAT - This assembly language library routine is called with three arguments. The first argument is a subroutine name, the second argument is the maximum number of words of core storage to be allocated, and the third argument is the statement number to return to if the available core storage is less than that requested by the second argument. ALOCAT is used to call C4DATA and SASSY. In each call an array name, the length of the array, and a statement number in a nonstandard return are passed to the subroutine.

C4DATA - MATERIAL INFORMATION PROCESSOR - This segment of the program is accessed by calling subroutine C4DATA via ALOCAT. It is responsible for reading the standard compositions

specification data and associated engineering-type specifications used to generate the mixing table and in performing the cell-weighting calculations. It checks the data for obvious errors and generates information necessary for processing the cross sections. It continues checking and preparing data for use in the functional modules activated by the sequence. The MATERIAL INFORMATION PROCESSOR is described in more detail in Sect. M7.3.

SCANON - This library routine is called to activate the feature that allows scanning for the word END during data reading.

IOSDUN - This subroutine is called several times to indicate the number of I/Os used in various operations.

PARAM - This subroutine is responsible for initializing certain default values for the KENO VI parameter data, reading the KENO VI parameter data, and writing the parameter data on the restart data file to communicate it to the KENO VI functional module. AREAD, IREAD, FREAD, and ZREAD are used to read the parameter data. BIRITE is used to write the parameter data on the restart data file, which is used by the functional module KENO VI to define the problem. CLEAR is called to zero a common. LIBEQL determines the unit from which cross sections are to be read. RNDIN is called to load the random number if it is entered as parameter data. OPNFIL opens the I/O units that will be used, RNDOUT stores the current random number, and TIMFAC adjusts the time estimate.

AREAD - This library routine may be called many times from subroutine PARAM. It is used to read parameter names and alphanumeric parameter data.

BIRITE - This library routine is used to write the binary data used by the functional modules.

CLEAR - This library routine is called to zero common DIMEN.

LIBEQL - This subroutine is called to determine which unit or device will be used to read the resonance-corrected microscopic library.

ZREAD - This library routine can be called from subroutine PARAM to read a hexadecimal random number that will be used as a kernel for the random number package.

RNDIN - This library routine is called from subroutine PARAM to transfer the random number read by ZREAD to the random number package. It is called only if a random number is entered as parameter data.

IREAD - This library routine can be called many times from subroutine PARAM. It is used to read integer parameter data.

FREAD - This library routine can be called many times from subroutine PARAM. It is used to read floating-point parameter data.

OPNFIL - This library routine opens the I/O units or devices that will be utilized in the problem.

RNDOUT - This library routine is used to preserve the current random number so it can be written on the restart data file.

TIMFAC - This library routine provides the proper adjustment factor by which the allotted time is multiplied to adjust the execution time for execution for different computers.

PARTBL - This subroutine is called only if KENO V.a is to be executed. It is responsible for printing the KENO V.a numeric and logical parameter tables. RNDOUT is called to preserve the current random number so that it can be printed in the numeric parameter table.

SASSY - KENO VI DATA PROCESSOR - This segment of the program is accessed by calling subroutine SASSY via ALOCAT. It calls SASSY6 which reads and checks the remainder of the KENO VI data used by CSAS6. It writes the data on the restart data file to be used by the functional module KENO VI. This segment of the program is described in detail in Sect. C6.3.4.

CLOSDA - This library routine is called to close out each direct-access file when the problem is completed in a normal manner.

STOP - This library routine is called several times to write error messages. It can be instructed to provide a traceback and can terminate execution or allow execution to continue, depending on the arguments provided to it.

### C6.3.3 THE MATERIAL INFORMATION PROCESSOR

The Material Information Processor is responsible for reading data that specify the cross-section library, define the materials to be used in the problem, and provide data used to apply geometric and resonance corrections to the cross sections. After the data are read and checked, calculations are performed to create a mixing table and provide data to the functional modules that are invoked by the control module to perform cross-section processing to prepare a problem-dependent cross-section library. Section M7.3 contains the program flow for the Material Information Processor.

### C6.3.4 KENO VI DATA PROCESSOR

This portion of the program, as shown in Fig. C6.3.3, is responsible for processing the KENO VI input data. All the KENO VI data, except the title and parameters, are read in this portion of the program. The data are carefully checked for errors and are written on the restart data file to be used by the functional module KENO VI.

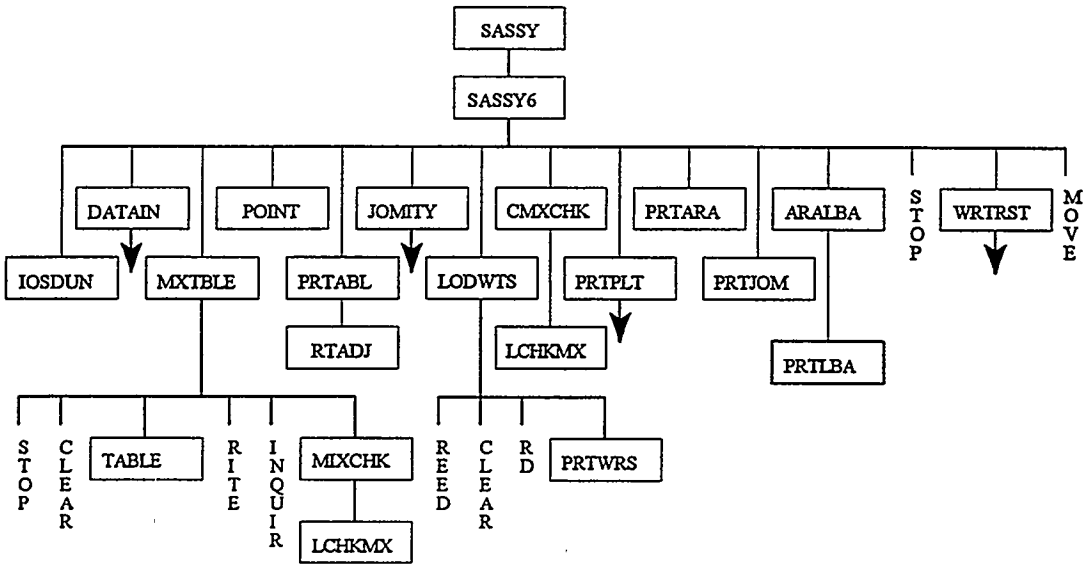



Figure C6.3.3 Flowchart of KENO VI data processor

SASSY - This subroutine is an interface between CSAS6 and the KENO-VI DATA PROCESSOR controlled by SASS6.

SASSY6 - This subroutine controls the processing of the KENO-VI data. It calls CLEAR to zero the direct-access pointer and length arrays. IOSDUN is called to indicate the number of I/Os used in the different operations involved in processing the KENO-VI data. Then RITE is used to load the direct-access pointer and length arrays on the direct-access data file. Various subroutines are called to read and check the KENO-VI data. If an error is recognized in the data, STOP is called to write a message, and control is returned to the MAIN program. If the data are found to be error free, a restart data file is written by subroutine WRTRST before returning to MAIN.

IOSDUN - This subroutine is called several times to indicate the number of I/Os used for various operations.

IOLEFT - This library routine stores the I/O count in common.

DATAIN - This subroutine controls the reading of all the KENO-VI input data except the title card and parameters. It is explained in more detail in Sect. C6.3.4.1.

MXTBLE - This subroutine initializes the threshold for printing cross-section processing error messages for invalid moments if it was not entered as data, sets up pointers for mixing table data for KENO VI, calls STOP if more storage is required, and calls TABLE to read the mixing table data. RITE is used to write the mixing table data to a random-access file, and INQUIR is called to set the next block for a direct or random-access device. MIXCHK is called to ensure that the mixtures used in the geometry data were also specified in the Material Information Processor data.

TABLE - This subroutine is responsible for reading the mixing table data created by the Material Information Processor and storing it in the proper arrays for KENO VI.

MIXCHK - This subroutine reads the KENO VI mixing table data and writes an error message if a mixture utilized in the geometry data was not created by the Material Information Processor.

LCHKMX - This logical function is called by MIXCHK to return a value of true if a mixture specified on a KENO VI geometry card is not specified in the mixing table and a value of false if it is.

POINT - This subroutine calculates pointers for storing and accessing KENO VI data in memory.

PRATBL - This subroutine prints the KENO VI table of additional data.

RTADJ - This subroutine right-adjusts the albedo boundary condition names that are read in as left-adjusted data. RTADJ locates the first blank character.

JOMITY - This subroutine is primarily responsible for generating additional geometry data, checking the geometry data, writing appropriate geometry error messages, and printing the geometry that is used in the problem. See Sect. C6.3.4.9 for additional information.

LODWTS - This subroutine is called only if PWT=YES is specified in the KENO VI parameter data. LODWTS prints the bias IDs versus the material I/Os used in the problem. It also loads and prints the biasing or weighting data. The library routine REED is used to load the bias IDs from the direct-access device. CLEAR is used to zero the weighting arrays. RD is used to load the biasing data, and subroutine PRTWTS is called to print it.

PRTWTS - This subroutine prints the group-dependent weight average array for each biasing region in a compact fashion.

CMXCHK - This subroutine is used to ensure that a component of the cell-weighted mixture is not used in the KENO VI geometry data. If it is, an error message is written, and an error flag is set.

LCHKMX - This logical function is called by CMXCHK to return a value of true if a mixture specified on a KENO VI geometry card is not a component of the cell-weighted mixture and a value of false if it is.

PRTPLT - This subroutine is called to generate and print two-dimensional (2-D) printer plots representing slices through the geometrical representation of the physical problem. See Sect. C6.3.4.14 for details.

PRTARA - This subroutine prints a table listing the array number, the number of units in the x, y, and z directions, and the nesting level for each array utilized in the problem.

PRTJOM - If a geometry error is recognized, this subroutine is called to print the KENO VI geometry data that is used in the problem.

ARALBA - This subroutine calls PRTLBA for each unit orientation array used in the problem. Subroutine PRTLBA checks each array for geometrical consistency and prints the array data.

STOP - This library routine is called to print an error message and terminate if errors were found in the CSAS6 or KENO VI data.

PRTLBA - If a geometry error is recognized, and more than one unit is entered in the geometry data, this subroutine is called to print the KENO VI unit orientation array data. STOP is called to write an error message if too many units are used in the KENO VI geometry description.

WRTRST - This subroutine controls the writing of all the KENO VI data on the restart data file for use by the KENO VI functional module. MOVE is called to load the required common information into COMMON/REC1/, a common that is used by the analytical sequence. The array that contains the IDs of the 1-D cross sections is written on the restart data file. REED and IO are then used to write the geometry data, unit orientation data, biasing data, and start data on the restart data file.

MOVE - This library routine is called to load data into common.

### C6.3.4.1 Control of the Data Input for KENO VI

This section of the program, shown in Fig. C6.3.4, is responsible for reading the input data that will be used by KENO VI, with the exception of the title and parameter data.

DATAIN - This subroutine controls the reading of all the KENO -VI data except the title card and parameter data. It initializes COMMON/STDATA/, the common that contains the start data, and initializes the array that contains the IDs of the 1-D cross sections that are to be utilized in the problem. The data reading is accomplished by reading blocks of data. Each block of data is preceded by a keyword that indicates the type of data being entered. After reading the keyword, the appropriate subroutine is called to read the accompanying data. After the data have been read, it is written on the direct-access data file. IOWRT is called several times to generate a table that lists the unit numbers used by KENO VI, their names, data set names, and the volume containing the data set. This table can be valuable for quality assurance applications. The library routine STOP is called to write error messages. Subroutine FLDATA is then called to supply information that was not entered as data.

CLEAR - This library routine is called to initialize COMMON/STDATA/. If biasing data are entered, it is also called to clear the space that will contain the biasing data. CLEAR is called with two arguments, a beginning location and a length. It initializes all included locations to zero.

AREAD - This library routine is used to read the READ flag and the keyword defining the type of associated data as well as the END flag and the associated keyword. It can be called many times from DATAIN.

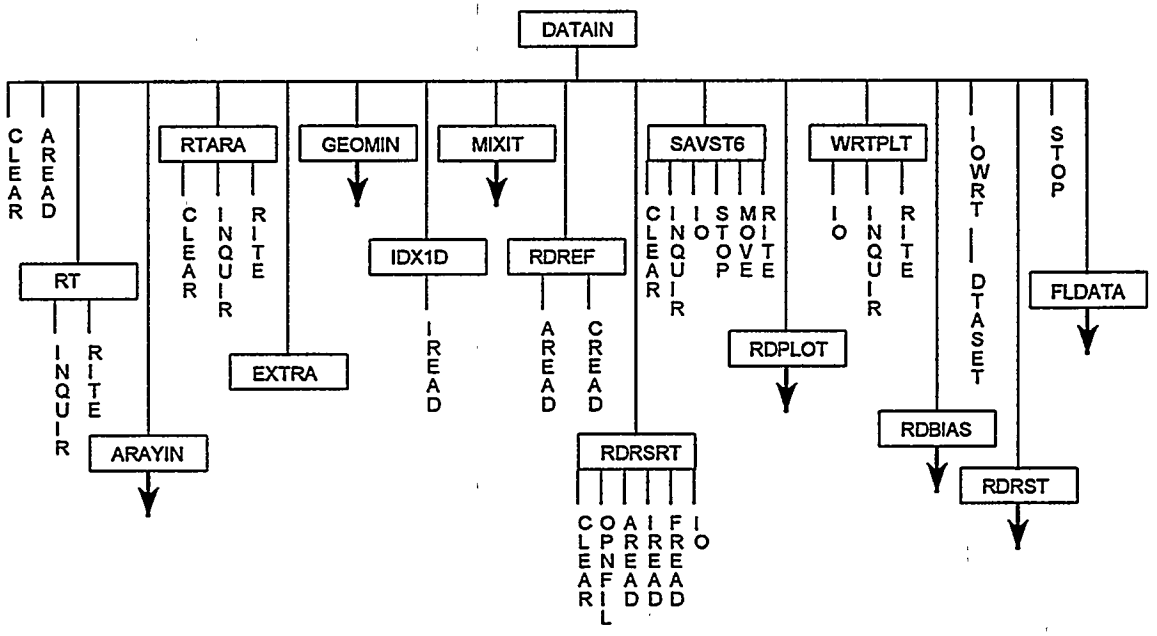



Figure C6.3.4 Flowchart of KENO-VI data reading routines

RT - This subroutine may be called many times to write data on the direct-access data file. It passes information between DATAIN and RITE. INQUIR is called to return the value of the next direct-access record after the geometry data and/or the extra 1-D cross sections are written on the direct-access device.

RITE - This library routine is called from RT to write an array of data on the direct-access data file.

ARAYIN - This subroutine is called to read data defining the array size. It also reads the unit orientation data if it is entered. ARAYIN is always called if array data are specified. See Sect. C6.3.4.2 for a more complete description.

RTARA - This subroutine is called only if ARAYIN is called. CLEAR is called to zero the array before RTARA reads the array data from the scratch unit and writes them on the direct-access device. INQUIR is called to return the value of the next direct-access record after the array data are written on the direct-access device.

EXTRA - This dummy subroutine is provided to allow the user to input extra data that is not normally processed by KENO-VI. The user must provide the programming to read and utilize the data.

GEOMIN - This subroutine is called to read the geometry region data. See Sect. C6.3.4.3 for additional information. The geometry region data are described in detail in Sect. F17.4.4.

IDX1D - This subroutine is called if the number of extra 1-D cross sections is greater than zero and extra 1-D data are entered. It reads the extra 1-D IDs and loads them into the MT array. The data reading is accomplished using the library routine IREAD. Section F17.4.9 describes the data input for defining extra 1-D data.

MIXIT - This subroutine is called to read the mixing table data block that defines the mixtures that are to be created. Section C6.3.4.4 explains the mixing procedure in more detail.

RDREF - This subroutine is called to read the boundary conditions (or albedo options) that are to be applied at the outer boundaries of the system described by the KENO-VI geometry data and unit orientation data. The boundary condition data are read using the library routine AREAD. The library routine CREAD is used to read the albedo names. Some preliminary data checks are made to detect invalid face code names and incompatible boundary conditions. Section F17.4.6 describes the input data for defining the KENO-VI boundary conditions.

RDSTRT - This subroutine is called to read start data for KENO-VI. This data are used to define the spatial distribution of the initial generation. The library routine CLEAR is called to zero the array that will contain the start data. The library routine AREAD is used to read the keywords associated with the start data. IREAD and FREAD are used to read the integer and floating-point start data, respectively. The library routine IO is used to write the start data associated with start type 6 on the scratch data file. The library routine INQUIR returns the value of the next direct-access record after writing the start data on the direct-access device. Data input for defining the initial source distribution is described in Sect. F17.4.8.

SAVST6 - This subroutine is called to save the data associated with start type 6. SAVST6 is called only if start type 6 is specified in the start type data. See Sect. F17.4.8 for information concerning the specification of start data. The library routine CLEAR is called to initialize the arrays that will contain the start data. The library routine IO is used to read the start data array from the scratch data file and load it into memory. The library routine MOVE is used to move the start data array into the neutron bank. The library routine RITE is called to write the neutron bank data on the direct-access data file. The library routine INQUIR returns the value of the next direct-access record after writing the start type 6 data on the direct-access device.

RDPLOT - This subroutine is called to read the plot data block that is used to generate printer plots. Section C6.3.4.5 explains the processing of the plot data in more detail. Section F17.4.11 describes the plot input data in detail.

WRTPLT - This subroutine reads the plot data block from the scratch data file and loads it on direct access. IO reads the data from the scratch data file and loads it in memory. RITE writes the data on the direct-access data file. INQUIR returns the value of the next direct-access record.

RDBIAS - This subroutine is called to read the biasing or weighting data to be used in the problem. See Sect. C6.3.4.6 for more information. The biasing input data block is described in detail in Sect. F17.4.7.

IOWRT - This subroutine is called with five arguments. In sequence they are (1) the output unit, (2) a four-character hollerith name representing a unit name, (3) the unit number represented by the second argument, (4) the number of words of hollerith information contained in the fifth argument,

and (5) hollerith information to be printed. IOWRT is called several times to generate a table of the unit numbers, their names, the data set names, and the volumes on which each resides.

DTASET - This library routine is called from IOWRT to provide the data set name of the requested I/O unit and the volume on which it resides.

RDRST - After the data reading is complete, this subroutine is called if a unit containing data for restarting the problem has been defined. It loads data from the restart data file as described in Sect. C6.3.4.7.

FLDATA - This subroutine is called to supply default data for arrays that were not entered as input. Section C6.3.4.8 contains a more detailed account of the exact procedure.

### C6.3.4.2. Read KENO VI Array Data

This section explains the procedure involved in reading the KENO VI array data utilized by the analytical sequence, as shown in Fig. C6.3.5.

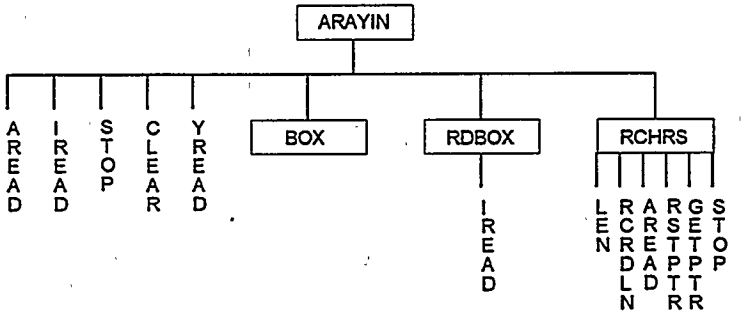



Figure C6.3.5 Flowchart for reading KENO VI array data

ARAYIN - This subroutine is called from DATAIN when the words READ ARRA are encountered. It is responsible for reading the data that define the size of each unit orientation array. It reads the unit orientation data for each array using YREAD for the FILL option and subroutine RDBOX for the loop option. BOX then writes the array data on the scratch data file. CLEAR is called to zero the unit orientation array. The data are read using the library routines AREAD, IREAD, and YREAD. The data read by this subroutine is described in Sect. F11.4.5.

AREAD - This library routine is used to read the keywords associated with the array data.

IREAD - This library routine is used to read the integer data associated with the array data.

STOP - This library routine is called to write an error message and stop if insufficient memory is available to accommodate the unit orientation array.

CLEAR - This library routine is used to zero arrays.

YREAD - This library routine is called to read the unit orientation data for the FILL option.

BOX - This subroutine is called to write the array data on the scratch data file.

RDBOX - This subroutine is called only if the LOOP option is used for entering the unit orientation data. It uses the library routine IREAD to read the unit orientation data. Some data consistency checks are made, and appropriate error messages are written if errors are encountered. If the input geometry is to be printed, RDBOX prints the unit orientation for each array.

RCHRS - This subroutine is used to read the comment associated with an array. The intrinsic function LEN is determines the length of the comment. GETPTR is used to return the current pointer in the input buffer. RSTPTR resets the pointer. AREAD is used to read the input data, and RCRDLN sets the length of the input buffer. STOP is called to write an error message and stop if the array comment is too long (i.e., the ending delimiter is missing).

### C6.3.4.3. Read KENO VI Geometry Data

This section of the program reads the KENO-VI geometry data required for the problem as shown in Fig. C6.3.6 and Fig. C6.3.7.

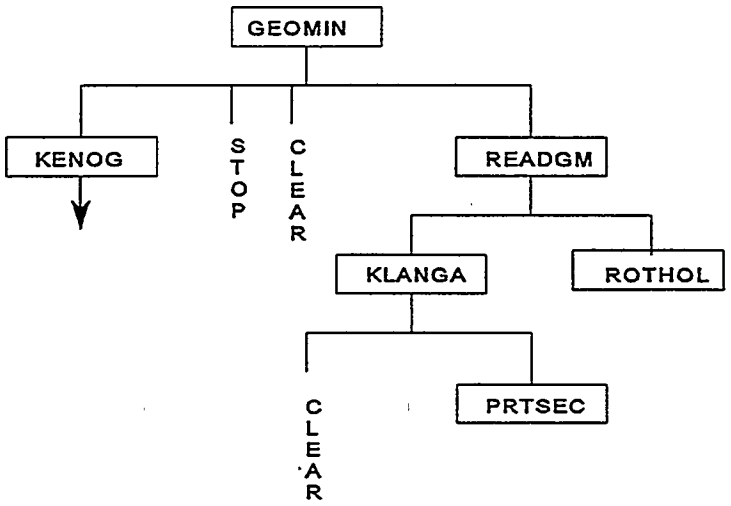



Figure C6.3.6 Flowchart of routines to read the KENO VI geometry data

This section of the program reads the geometry data.

GEOMIN - This subroutine controls the reading of the KENO-VI geometry data. KENOG is called to read the data and write it on the scratch data file. Pointers are then calculated and CLEAR is called to zero the arrays that hold the geometry data. READGM is called to read the data from the scratch data

file and load it into the appropriate arrays. STOP is called to write an error message if more storage is needed. Section F17.4.4 describes the KENO-VI geometry input data in detail.

KENOG - This subroutine uses AREAD to read the geometry work and IREAD to read the unit number. KENOG then calls the appropriate subroutine, based on the geometry work previously read, to read in the associated geometry data. The necessary geometry data block is written on the scratch data unit. If the input geometry data block is to be printed, KENOG does this as the data are read.

READGM - This subroutine reads the geometry data from the scratch data unit and loads them into the proper arrays.

KLANGA - This subroutine reads the geometry data from the scratch data unit created in KENOG and loads them into the proper arrays.

ROTHOL - This subroutine creates a rotation matrix for the holes if required.

PRTSEC - This subroutine writes the sector array to the output file.

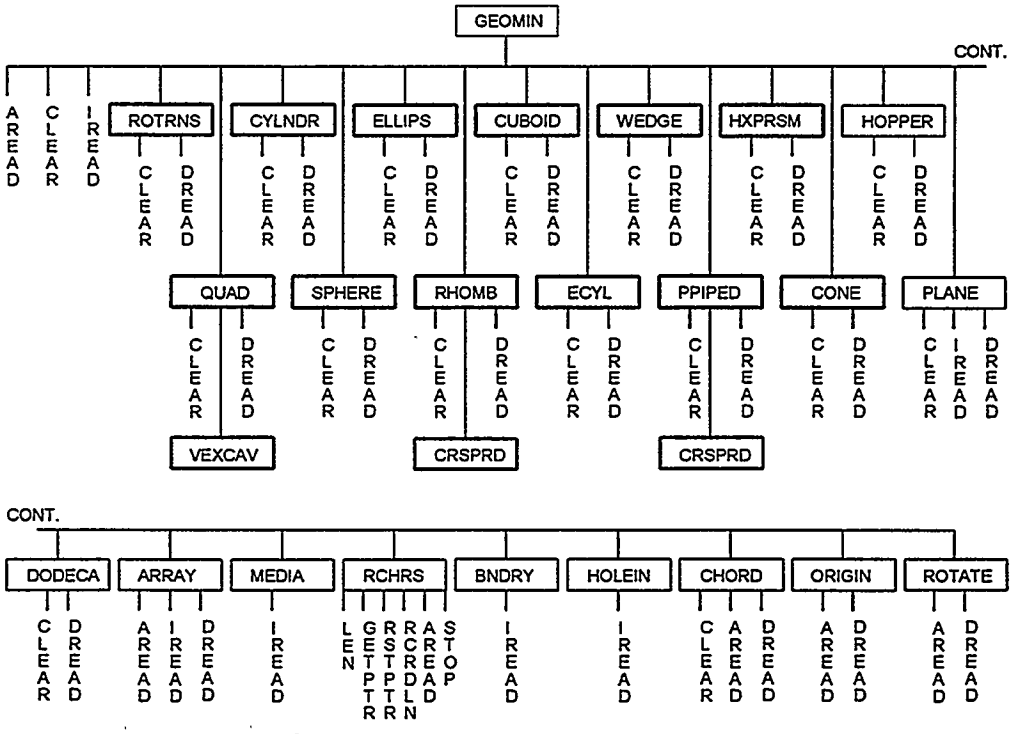



Figure C6.3.7 Flowchart for reading KENO VI geometric shape data

This section of the program contains the subroutines that read the geometry data for each geometric input shape as well as holes and arrays and writes the data to a scratch file.

ROTRNS - This subroutine translates and rotates the location and orientation of a geometric body based on the geometric keyword read in and the data relating to the body read from the subroutines ORIGIN and ROTATE.

SPHERE - This subroutine read the radius of a sphere using DREAD and generates the coefficients of the quadratic equation that represents the surface of the sphere.

CYLNDR - This subroutine reads the radius and top and bottom surfaces of a cylinder using DREAD and generates the set of quadratic equations that represent the surface of the cylinder.

ECYL - This subroutine reads the x-radius, y-radius, and top and bottom surfaces of an elliptical cylinder using DREAD and generates the set of quadratic equations that represent the surface of the cylinder.

ELLIPS - This subroutine reads the x-radius, y-radius, and z-radius of an ellipse using DREAD and generates the quadratic equation that represents the surface of the ellipse.

CONE - This subroutine reads the top radius, bottom radius, top surface, and bottom surface using DREAD and generates the set of quadratic equations that represent the surface of the cone section.

CUBOID - This subroutine reads the +X, -X, +Y, -Y, +Z, and -Z faces of the cuboid using DREAD and generates the set of quadratic equations that represent the surfaces of the cuboid.

HXPRSM - This subroutine reads the inscribed radius, the top surface and the bottom surface of a hexprism using DREAD and generates the set of quadratic equations that represent the surface of the hexprism.

WEDGE - This subroutine reads the x-coordinate of the triangular base, the x and y coordinates of the intersection of the other 2 sides, and the height of the wedge using DREAD and generates the set of quadratic equations that represent the surface of the wedge.

RHOMB - This subroutine reads the length of a side and the angle between the y-face and the y-axis of the rhomboid using DREAD and generates the set of quadratic equations that represent the surface of the rhomboid.

CRSPRD - This subroutine generates and returns a unit vector from the cross product of 2 input vector.

PPIPED - This subroutine reads the x-distance, y-distance, and z-distance, the angle between the y-distance and the y-axis, the angle between the z-distance and the z-axis, and the angle between the projection of the z-distance on the x-y plane and the x-axis using DREAD and generates the set of quadratic equations that represent the surface of the parallelpiped.

HOPPER - This subroutine reads the x-half length, y-half length and z-surface for the top and the x-half length, y-half length, and z-surface for the bottom using DREAD and generates the set of quadratic equations that represent the surface of the hopper.

PLANE - This subroutine reads the keywords and associated dimensions that represent a plane using AREAD and DREAD respectively and generates the quadratic equation that represents the planar surface.

QUAD - This subroutine reads the keywords and associated coefficients of a quadratic using AREAD and DREAD respectively.

VEXCAV - This function determines if the quadratic equation specified using QUAD is convex or concave.

DODECA - This subroutine reads the inscribed radius of the dodecahedron using DREAD and generates the quadratic equations that represents the surface of the dodecahedron.

ARRAY - This subroutine reads the array number and definition vector of the region containing the array using IREAD, the keyword for placing the array in the unit using AREAD, and the origin of the unit at the specified array position relative to the origin of the unit containing the array using IREAD and DREAD.

MEDIA - This subroutine reads the media number, the importance region, and the definition vector for the sector using IREAD.

BNDRY - This subroutine reads the vector definition array of the boundary of each unit using IREAD.

HOLEIN - This subroutine reads the unit number inserted in the hole and the definition vector of the region containing the hole using IREAD.

CHORD - This subroutine reads the chord orientation keyword using AREAD and the plane location using DREAD, then generates the quadratic equation which represents the surface of the plane and adds it to the set of equations that represents the geometric shape being developed.

ORIGIN - This subroutine reads the set of translation keywords using AREAD and the values associated with each keyword using DREAD. These values are then stored in common block /RTRN/.

ROTATE - This subroutine reads the set of rotation keywords using AREAD and the values associated with each keyword using DREAD. These values are then stored in common block /RTRN/.

### C6.3.4.4. Read KENO VI Mixing Table Data

This section deals with reading the mixing table data, as shown in Fig. C6.3.8.

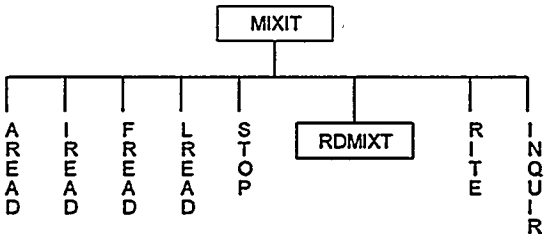



Figure C6.3.8 Flowchart for reading KENO-VI mixing table data

MIXIT - This subroutine uses AREAD to read the mixture keywords and the scattering keyword. IREAD is used to read the mixture numbers and the number of scattering angles as well as the nuclide IDs. LREAD is used to determine if the next item is numeric or character data. FREAD reads the number densities. The necessary data arrays are written on the scratch data file. Pointers are calculated for the necessary storage arrays, and RDMIXT is called to load the data from the scratch file into the storage arrays. The library routine RITE is called to write the mixing table information on the direct-access device, and INQUIR is called to return the value of the next direct-access record.

LREAD - This library routine returns a value of "true" if the next position contains a numeric digit. Otherwise, a value of "false" is returned.

STOP - This library routine is called from MIXIT if the storage space is insufficient to hold the mixing table arrays.

RDMIXT - This subroutine reads the mixing table data arrays from the scratch data file and loads them into the storage arrays.

### C6.3.4.5. Read KENO-VI Plot Data

This section of the program, shown in Fig. C6.3.9, reads the plot or picture data used to generate printer plot maps of the mixtures, units, and/or bias IDs used in the problem. The plot input data block is described in Sect. F17.4.11.

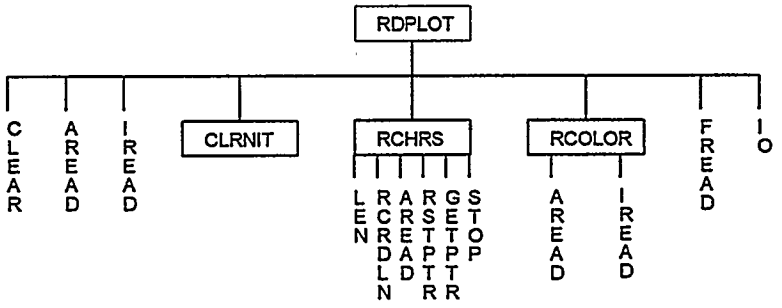



Figure C6.3.9 Flowchart for reading KENO VI plot data

RDPLOT - This subroutine uses CLEAR to initialize the data arrays. RCHRS is used to read the plot title and the character string of symbols to be used in the plot. AREAD, IREAD, and FREAD are used to read the plot or picture input data. SQRT is used to determine the normalization factor for the direction cosines, and IO is called to load the plot data on the scratch data file.

CLRNIT - This subroutine initializes the 255 default colors used in screen plots.

RCHRS - This subroutine is used to read the plot title and the character string that defines the symbols to be used in the plot. LEN is a FORTRAN-supplied routine that sets the length of the data string. GETPTR is used to return the current pointer in the input buffer. RSTPTR resets the pointer. AREAD is used to read the input data, and RCRDLN sets the length of the input buffer. STOP is called to write an error message if the plot title is too long and terminate the problem.

RCOLOR - This subroutine is used to read new colors that are used in screen plots.

### C6.3.4.6. Read KENO-VI Biasing Data

This section of the program reads the biasing data used in the problem. The biasing input data block is described in Sect. F17.4.7 and illustrated in Fig. C6.3.10.

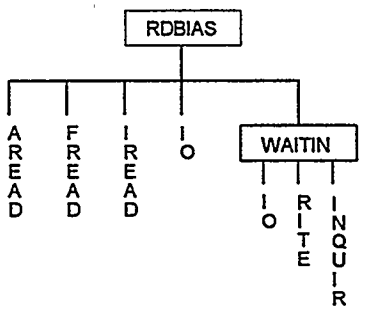



Figure C6.3.10 Flowchart for reading KENO-VI biasing data

RDBIAS - This subroutine is responsible for reading the biasing data block and writing it on the scratch data file. AREAD is used to read the keywords used in the biasing data and a title for the biasing material if the energy- and space-dependent values of the biasing function are entered from cards. IREAD and FREAD are used to read the numerical data. Pointers for the storage arrays needed to process the biasing data are determined, and WAITIN is called to load the data from the scratch data file into the storage arrays and write them on the direct-access data file.

WAITIN - This subroutine reads the biasing data block from the scratch data file and loads it into the storage arrays. IO is used to load the energy- and space-dependent biasing function (*wtavg*) into the storage arrays. RITE is used to write the biasing data on the direct-access data file. The library routine INQUIR is called to set the next block on a direct- or random-access device.

### C6.3.4.7. Read KENO-VI Restart Data

This section of the program reads restart information from the restart data file. The flowchart for this procedure is given in Fig. C6.3.11.

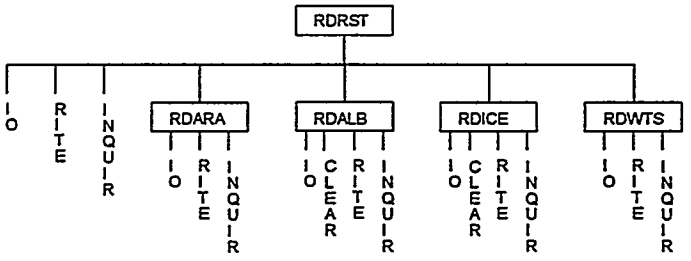



Figure C6.3.11 Flowchart for reading KENO-VI restart data

RDRST - This subroutine is called only if the problem is to use data from the restart data file. The program recognizes that restart data will be read if the restart unit is defined as a number greater than zero. IO is used to load the array that contains the 1-D IDs from the restart data file. Each type of data is loaded from the restart data file using IO and is written on the direct-access data file by RITE. All restart data except the mixed cross-section data, the differential albedo data, the array data, and the biasing data are processed directly in RDRST. RDICE is called to load the cross-section data on the direct-access data file, RDALB is called to load the differential albedo data on the direct-access data file, RDARA is called to load the array data on the direct-access data file, and RDWTS is called to load the biasing data on the direct-access data file. The library routine INQUIR is called to return the value of the next direct-access record.

RDALB - This subroutine is called from RDRST to read the albedo data block from the restart data file and write it on the direct-access data file. IO is used to load the albedo pointer and length arrays into memory from the restart data file. CLEAR is called to zero the albedo pointer and length arrays, and RITE writes them on the direct-access data file. INQUIR stores the next direct-access block number. Each record of albedo data is read from the restart data file, loaded into memory using IO,

and written on the direct-access data file using RITE. When all the records of albedo data have been processed, the updated pointer and length arrays are rewritten over the initial ones using RITE.

RDICE - This subroutine is called from RDRST to read the cross-section data block from the restart data file and write it on the direct-access data file. CLEAR is called to zero the pointer and length arrays. The length array is then read from the restart data file. RITE is used to write the pointer array and the length array on the direct-access data file. INQUIR stores the next direct-access block number. Then IO and RITE are used to load the cross-section data from the restart data file and write them on the direct-access data file. This procedure is repeated for every record of each mixture. The updated pointer and length arrays are then rewritten over the initial ones using RITE.

RDARA - This subroutine is called from RDRST to read the array data block from the restart data file and write it on the direct-access data file. IO is used to load the data from the restart data file, and RITE is used to write them on the direct-access data file. INQUIR stores the next direct-access block number.

RDWTS - This subroutine is called from RDRST to read the biasing data block from the restart data file and write it on the direct-access data file. IO is used to load the data from the restart data file, and RITE is used to write them on the direct-access data file. INQUIR stores the next direct-access block number.

### C6.3.4.8. Generate Remaining KENO-VI Data

This section of the program, shown in Fig. C6.3.12, is responsible for generating data that are required for a KENO-VI problem but are not entered directly as data.

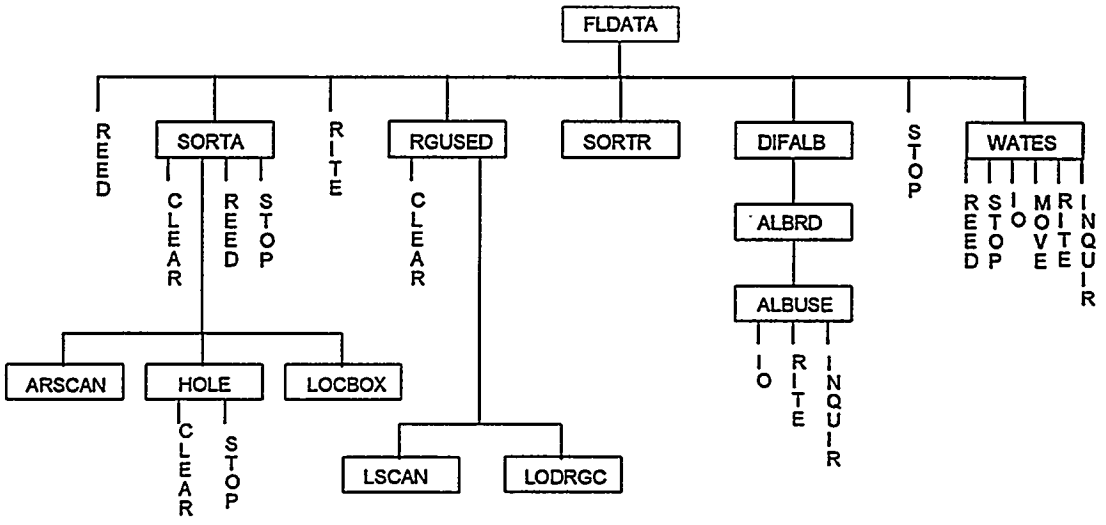



Figure C6.3.12 Flowchart for generating remaining KENO VI data

FLDATA - The library routine REED is used to load data arrays from the beginning of the mixtures used in the geometry through the geometry data. The library routine STOP may be called to print an error message. SORTA is called to determine which arrays and holes are used as well as the array and hole nesting levels. RITE is called to write the geometry data on the direct-access data file. RGUSED is called to determine which geometry regions are used in the problem. SORTR is called to generate the mixture correspondence array. It is called again to generate the bias region correspondence array. These correspondence arrays are used to avoid storing mixture cross sections and biasing data that are entered as data but are not actually used in the problem. If boundary conditions specify differential albedo data, and they are not available from the restart data file, DIFALB is called to read the albedo data block and load the requested data on the direct-access file. If the requested biasing data were unavailable from the restart data file, WATES is called to load the energy and position-dependent biasing function (*wtavg* array) on the direct-access data file.

SORTA - This subroutine checks to see that the global array is properly defined. It determines the array correspondence array and the nesting levels for holes and arrays. SORTA uses CLEAR to initialize arrays. REED is used to load the array data, and STOP is used to write an error message and terminate if more computer storage is needed for the problem.

ARSCAN - This function, which is called from sorta, is used to determine if array data is entered when an array is specified.

HOLE - This subroutine is called from SORTA to determine what holes occur at the next nesting level and to adjust the array nesting level for arrays that occur in holes. It also checks to ensure holes are not recursively nested.

LOCBOX - This function subprogram is called from SORTA to determine the unit or box type at a given position in the unit orientation array.

RGUSED - This subroutine determines which geometry regions are used in the problem. The library routine CLEAR is called to zero the space for the region correspondence array. LSCAN determines if a particular unit or box type has been used in the unit orientation array, and if it has, LODRGC is called to load the region number into the region correspondence array.

LSCAN - This is a logical function that returns a value of true if the specified unit or box type is used in the unit orientation array. A value of false is returned if the unit or box type was not used in the unit orientation array.

LODRGC - This subroutine loads the region number in the region correspondence array.

SORTR - This subroutine is called twice from FLDATA to create a mixture correspondence array and a biasing correspondence array. These correspondence arrays are used to avoid storing mixture cross sections and biasing information that are defined in the input data but were not referenced in the geometry data utilized in the problem. They are also used throughout the code for accessing the proper mixture cross sections and biasing information.

DIFALB - This subroutine is called if differential albedos are specified as a boundary condition but are not available from the restart data file. It rewinds the albedo data file, reads the header record, and calculates pointers. ALBRD is called to load the albedo data.

ALBRD - This subroutine searches through the albedo data file to locate the requested albedo name or boundary condition. If it is not found, an error message is written. If it is found, the number of different differential albedos that were requested are tabulated and ALBUSE is called.

ALBUSE - This subroutine writes the albedo pointer array on the direct-access data file. IO is used to load data from the albedo data file, and RITE is used to write the data on the direct-access data file. Then the pointer and length arrays are rewritten on the direct-access data file. INQUIR stores the next direct-access block number.

WATES - This subroutine reads the biasing input data block from the direct-access data file and reads the KENO-VI weights library. STOP is called if the computer storage space is too small to contain the energy- and position-dependent biasing function (*wtavg*). IO is used to load the biasing function into a temporary storage array. If a specific biasing function is to be used, MOVE is called to load it into the *wtavg* array. If biasing or weighting data are entered from cards, REED is used to load the data into a temporary storage array. If a specific biasing function that was loaded from cards is to be used in the problem, MOVE is called to load it into the *wtavg* array. When all the data have been processed, RITE is called to load the biasing input data on the direct-access data file. RITE is called again to load the *wtavg* array on the direct-access data file. INQUIR is used to store the next direct-access block number.

### C6.3.4.9. Process the KENO-VI Geometry Data

This portion of the program, shown in Fig. C6.3.13, is primarily responsible for loading the geometry data, generating additional geometry data, checking the geometry for consistency, writing error messages related to the geometry, and printing the geometry that is used in the problem.

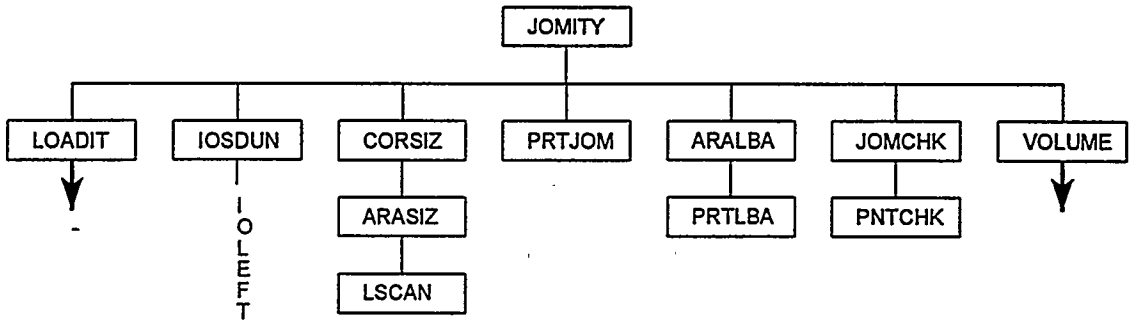



Figure C6.3.13 Flowchart for processing KENO VI geometry data

JOMITY - This subroutine is responsible for generating additional geometry data, checking the geometry data, writing geometry error messages, and printing the geometry.

LOADIT - This subroutine loads the geometry data and the nonsupergrouped portion of the albedo data. Section C6.3.4.10 contains a more detailed description of the procedure.

IOSDUN - This subroutine returns the numbers of I/Os used. The library routine IOLEFT returns the numbers of I/O requests remaining before the system cancels the job.

CORSIZ - This subroutine sends the appropriate lattice or array information to ARASIZ for each lattice that is used in the problem. Using this information, it calculates the overall positive dimensions of the global array. The library routine, SQRT, is utilized to calculate the maximum chord length of an unreflected array, a reflected array, or a single unit problem.

ARASIZ - This subroutine uses the array unit orientation data to calculate the positive dimensions of the core boundary for that array or lattice. The function LSCAN is called to determine if a specified unit has been used in the array. ARASIZ also checks to ensure that the faces of adjacent units are the same size and shape. Several error messages are written if errors are encountered.

LSCAN - This is a logical function that returns a value of true if the specified unit or box type is used in the unit orientation array. A value of false is returned if the unit or box type was not used in the unit orientation array.

PRTJOM - This subroutine prints the geometry data used in the problem.

ARALBA - This subroutine calls PRTLBA for each array that is used in the problem.

PRTLBA - This subroutine is called to print the unit orientation data for each lattice or array that is used in the problem. It may print a warning message associated with one or more lattices.

VOLUME - This subroutine is responsible for calculating the volume of each geometry region and the cumulative volumes for each unit that is used in the problem. See Sect. C6.3.4.12 for additional details. Volumes are currently not calculated in KENO VI.

### C6.3.4.10. Load KENO-VI Data from the Direct-Access File

This portion of the program loads data from the direct-access data file into permanent memory as shown in Fig. C6.3.14.

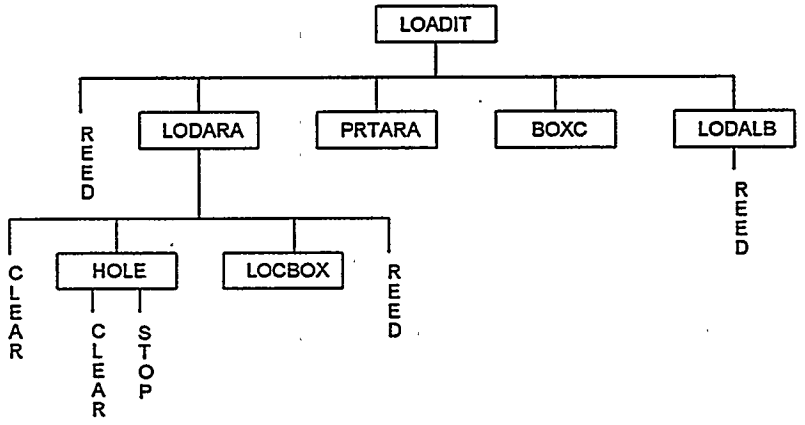



Figure C6.3.14 Flowchart for loading data from direct access

LOADIT - This subroutine calls the library routine REED to load the geometry data. If the problem is an array problem (lattice geometry), LODARA is called to load the lattices that are used in the problem and recompute and readjust the array nesting level array and hole nesting array. If multiple boxes are used in the problem, PRTARA is called to print the unit orientation array for each lattice used in the problem. BOXC is called to load the box correspondence array, and LODALB is called to load the nonsupergrouped portion of the albedo data.

LODARA - This subroutine is responsible for loading the lattices (unit orientation arrays) that are used in the problem, computing the hole nesting level array, and computing and adjusting the array nesting level array. CLEAR is used to initialize the arrays, and REED is used to load the unit orientation arrays. HOLE and LOCBOX are both called by LODARA.

HOLE - This subroutine is called from LODARA to determine which holes occur at the next nesting level and to adjust the array nesting level for arrays that occur in holes. It also checks to ensure that

holes are not recursively nested. CLEAR is used for initialization purposes, and STOP is called if holes are recursively nested.

LOCBOX - This function is called from LODARA to return the unit type at a given position in the unit orientation array.

PRTARA - This subroutine prints a table of the arrays used in the problem. The array number, the number of units in the x, y, and z directions, and the nesting level are printed for each array.

BOXC - This subroutine uses the number of units or box types and the geometry region number corresponding to the first and last geometry region of each unit to generate the box correspondence array that contains the unit or box type number for each geometry region. This is loaded in the appropriate position as it is generated.

LODALB - This subroutine calls the library routine REED to load the pointer and length arrays for the albedo data from the direct-access data file. REED is used to load the nonsupergrouped albedo data, for each albedo that is used, into a temporary array. A loop over the number of angles is then used to load these data into the appropriate arrays. When all of the albedos used in the problem have been processed, REED is called to load the pointer arrays for the cross sections and albedos as well as the arrays defining the albedo to cross-section energy group correlation.

### C6.3.4.11. Calculate KENO VI Geometry Volumes

This portion of the program, illustrated in Fig. C6.3.15, is responsible for calculating the volume of each KENO-VI geometry region used in the problem, the cumulative volumes for each unit used in the problem, the number of times each unit was used in the problem, and the total volume of each region summed over all occurrences. Currently, volumes are not calculated in KENO VI.

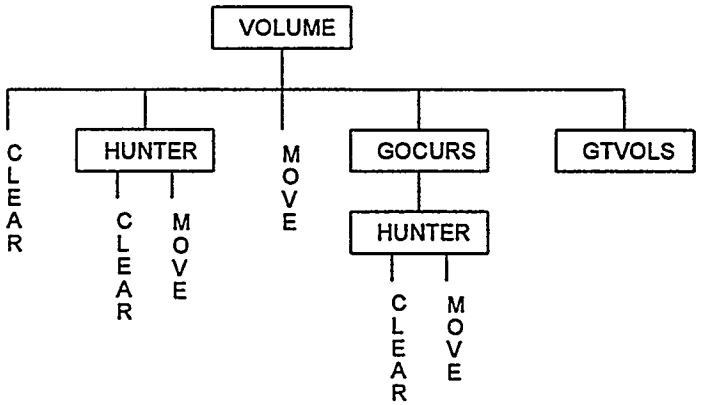



Figure C6.3.15 Flowchart for calculating KENO VI geometry volumes

VOLUME - This subroutine calculates the volume of each region for every unit that is used in the problem. It then calculates the cumulative volumes for each unit. The intrinsic functions ASIN, which returns the arc sign of a variable, and SQRT, which returns the square root of an expression, are used when calculating the volume of a hemicylinder. CLEAR is used to initialize arrays. If an external reflector is present, HUNTER is called to determine the number of times each array and/or hole is used in the reflector. GOCURS is used to determine the number of times each unit, array, and hole is used in the problem. GTVOLS is called to calculate and print the number of occurrences for each unit and the corresponding total volumes for the entire system. CLEAR is called to initialize the array for the total volume of each mixture used in the problem prior to calculating and printing those totals. Currently, volumes are not calculated.

HUNTER - This subroutine determines the number of times each unit, array, and hole is used. CLEAR is used to initialize storage arrays for the present hole level and the next hole level. MOVE is used to move the storage arrays.

GOCURS - This subroutine loops over the array size and calls HUNTER to determine the number of times each unit, lattice or array, and hole is used in the problem.

GTVOLS - This subroutine calculates the total volume of each region for the entire problem by multiplying the volume of the region by the number of times the region is used in the problem.

### C6.3.4.12. Generate Printer Plot

This portion of the program, illustrated in Fig. C6.3.16, generates printer plots of a 2-D slice through the KENO-VI geometry. As many plots as are desired can be printed.

PRTPLT - This subroutine controls the generation of the printer plots. The library routine CLEAR is called to zero the necessary space. Subroutine UNTCRS is called to generate a unit correspondence array. REED is used to load the picture data from direct access. INQUIR is called to return the value of the next direct-access data block. The picture title is printed, and subroutine RELATE is called to print a heading for the symbol map and print the symbol map. The picture coordinates, direction cosines, and number of symbols across and down the page, as well as the step intervals, are printed. Then subroutine PRINT is called to generate the actual picture.

UNTCRS - This subroutine calls CLEAR to zero the unit correspondence array. It then loads the appropriate unit number in the appropriate position of the array.

RELATE - This subroutine calls CLEAR to zero the reverse correspondence array. It then prints the plot header and loads the array that correlates the symbols.

PRINT - This subroutine determines the number of pages that will be needed to print the printer plot picture. Subroutine MESH is called to load the appropriate mixture numbers, unit numbers, or bias ID numbers for each line of print for the picture. Then PRINT prints the line of symbols corresponding to them.

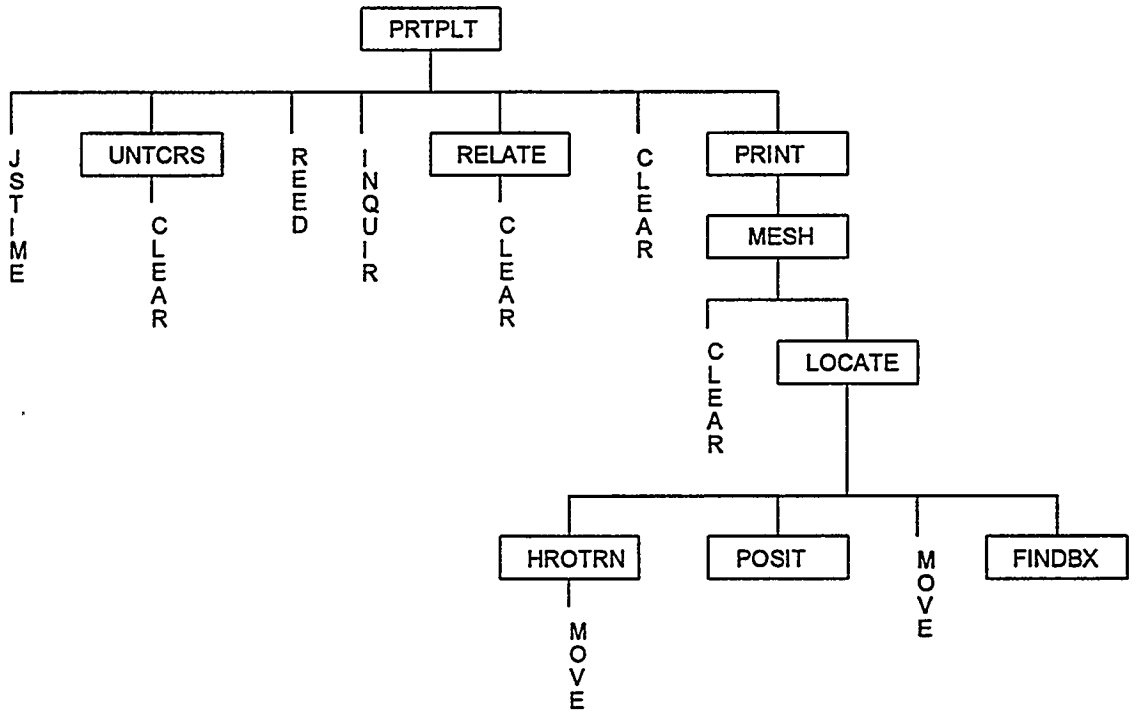



Figure C6.3.16 Flowchart for printer plots

MESH - This subroutine loads an array that contains the appropriate mixture number, unit number, or bias ID number for each character in a line. CLEAR is called to initialize arrays if nested holes or nested arrays are present in the problem. LOCATE is called to determine the geometry region and unit that contains each mesh point. MESH then loads the mixture number unit number, or bias ID number.

LOCATE - This subroutine is responsible for determining the geometry region for each mesh point in the picture. Subroutine POSIT is called to determine the region that contains the specified mesh point. If array data are used, MOVE is called to load the current array data into the array stack. If the mesh point is within an array, FINDBX is called to determine the position within the lattice or array that contains the mesh point. LOCBOX is then called to determine the unit or box type that contains the mesh point. POSIT is used to determine the geometry region that contains the mesh point. If the mesh point is in a region that contains a hole, the coordinate of the mesh point is translated to the hole and POSIT is called again.

FINDBX - This subroutine locates the position in an array that contains a specified point.

LOCBOX - This function returns the unit or box type at a given position in an array.

POSIT - This subroutine determines the region within a unit that contains a specified point.

### C6.3.4.13. Write KENO VI Input Data on Restart File

This portion of the program, illustrated in Fig. C6.3.17, is responsible for writing all data except the calculated results on the restart data file. This section of the program is omitted if a unit number has not been assigned for the restart data file. This information is entered as parameter data, WRS=, as described in Sect. F17.5.3. The calculated results are written on the restart data file later in the program. The restart data file is used for restarting a problem.

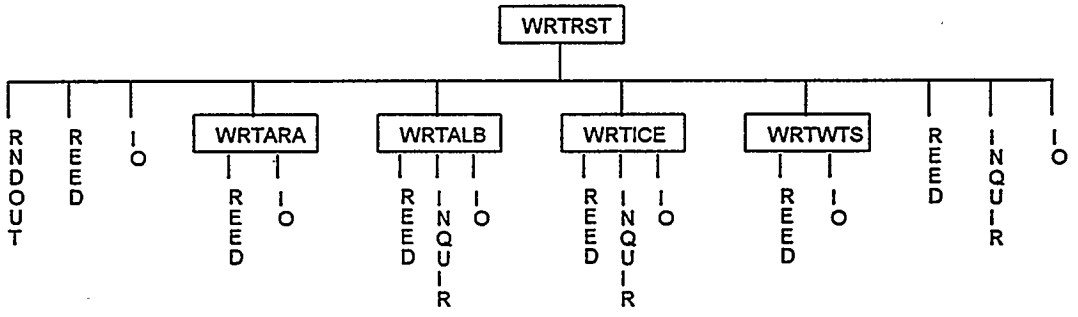



Figure C6.3.17 Flowchart for writing data on the restart data file

WRTRST - This subroutine writes the input data on the restart data file. RNDOUT is called to return the seed of the next random number. The problem title and parameter data are in the first record written on the restart data file. The array containing the IDs of the 1-D cross sections is then written. The geometry region data, mixing table data, extra 1-D data, biasing data, start data, plot data, and energy and inverse velocities are written.

The unit orientation header record is written, and WRTARA is called to write the unit orientation data. The albedo header record is written, and WRTALB is called to write the albedo data. The cross-section header record is written, and WRTICE is called to write the cross sections. The header record for user-supplied weighting data is written, and WRTWTS is called to write out the user-supplied weighting or biasing data.

WRTARA - This subroutine is called from WRTRST to write the array number, array size, and corresponding unit orientation array on the restart data file for each array that is entered in the problem. The library routine REED is called to load the direct-access pointers for the albedo data block. INQUIR is called to return the value of the next direct-access record. REED is used to read the data from the direct-access data file, and IO is used to write it on the restart data file.

WRTALB - This subroutine is called from WRTRST to write the albedo data on the restart data file. The library routine REED is used to read the albedo data from the direct-access data file, and IO is used to write the data on the restart data file.

WRTICE - This subroutine is called from WRTRST to write the cross-section data block on the restart data file. REED is called to load the direct-access pointers for the cross-section data blocks. INQUIR

is called to return the value of the next direct-access record. The library routine REED is used to read the rest of the cross-section information from the direct-access data file, and IO is used to write it on the restart data file.

WRTWTS - This subroutine is called from WRTRST to write the biasing input data block on the restart data file. The library routine REED is used to read the data block from the direct-access data file, and IO is used to write it on the restart data file.

## C6.4 INPUT DATA GUIDE

This section describes the input data required for Criticality Safety Analysis Sequence No. 6. This sequence contains two control modules. These are listed in Table C6.4.1 with the functional modules they invoke. Note that modules with an "X" in the name create cell-weighted libraries by using XSDRNPM.

Each sequence generates various libraries that may be saved for future use. Table C6.4.2 lists the various I/O units that may be saved and their definitions. Table C6.4.3 lists those I/O units containing cross-section libraries that are generated by each sequence. See Table C6.5.1 in Section C6.5 for a complete list of I/O units utilized by Criticality Safety Analysis Sequence No. 6 and Table C6.5.2 for the I/O units utilized by the associated functional modules.

The input data for these CSAS6 modules are composed of two broad categories of data. The first (Material Information Processor, including Standard Composition Specification Data and Unit Cell Geometry Specification) specifies the cross-section library and defines the composition of each mixture and the unit cell geometry that is used to process the cross sections. The second category of data, the KENO-VI input data, are used to specify the geometric and boundary conditions that represent the physical 3-D configuration of the problem. Both data blocks are necessary for CSAS26 and CSAS62X.

All data are entered in free form, allowing alphanumeric data, floating-point data, and integer data to be entered in an unstructured manner. Eighty columns of data entry are allowed, simulating a card of data. Data can usually start or end in any column. Each data entry must be followed by one or more blanks to terminate the data entry. For numeric data, either a comma or a blank can be used to terminate each data entry. Integers may be entered for floating values. For example, 10 will be interpreted as 10.0. Imbedded blanks are not allowed within a data entry unless an E precedes a single blank as in an unsigned exponent in a floating-point number. For example, 1.0E 4 would be correctly interpreted as $1.0 \times 10^4$.

The word "END" is a special data item. An "END" may have a name or label associated with it. The name or label associated with an "END" is separated from the "END" by a single blank and is a maximum of 12 characters long. *At least 2 blanks MUST follow every labeled and unlabeled "END." It is the user's responsibility to ensure compliance with this restriction. Failure to observe this restriction can result in the use of incorrect or incomplete data without the benefit of warning or error messages.*

Multiple entries of the same data value can be achieved by specifying the number of times the data value is to be entered, followed by either R, *, or $, followed by the data value to be repeated. Imbedded blanks are not allowed between the number of repeats and the repeat flag. For example, 5R12, 5*12, 5$12, or 5R 12, etc., will enter five successive 12s in the input data. Multiple zeros can be specified as nZ where n is the number of zeroes to be entered. The SCALE free-form reading routines are described in more detail in Sect. M3.

The purpose of this section is to define the input data in discrete subsections relating to a particular type of data. Tables of the input data are included in each subsection, and the entries are described in more detail after the tables illustrated in Fig. C6.4.1. Section C6.4.1 is provided for experienced users and contains all the tables that are used to describe the input data. Section C6.4.2 describes input data requirements and the data-checking capability. Sections C6.4.3 through C6.4.8 describe the Material Information Processor data, Sect. C6.4.9 summarizes the KENO-VI data that are explained in detail in Sect. F17.4.

Resonance-corrected cross sections utilize Dancoff corrections based on the unit cell specification. Because only one unit cell can be specified in the data, resonance corrections can only be applied to one mixture via conventional methods. However, if multiple resonance-corrected mixtures are required, the necessary data can be entered via the OPTIONAL PARAMETER DATA described in Section C6.4.8 and Tables C6.4.4 and C6.4.13.

Table C6.4.1 Functional modules executed by CSAS6 control modules

| Control module | Search function | Functional modules executed by the control module | | | |
|---|---|---|---|---|---|
| CSAS26 | No search | BONAMI | NITAWL-II | | KENO-VI |
| CSAS26X | No search | BONAMI | NITAWL-II | XSDRNPM | KENO-VI |

Table C6.4.2 I/O units on which cross-section libraries are written

| Unit | Type of data | Creating module |
|---|---|---|
| 2 | Resonance-corrected mixed working library | ICE |
| 3 | Resonance-corrected, cell-weighted working library | XSDRNPM |
| 4 | Resonance-corrected working library | NITAWL-II |
| 14 | Resonance-corrected, mixed cross-section, MORSE/KENO library. If the invoked sequence contains an X in its name, this library contains a cell-weighted mixture identified as mixture 500 | ICE |

Table C6.4.3 I/O units generated by CSAS6 and associated modules

| | | Unit numbers of generated cross-section libraries | | | |
|---|---|---|---|---|---|
| Module name | Primary product | 2 | 3 | 4 | 14 |
| CSAS26 | KENO-VI k-effective | X | | X | X |
| CSAS26X | KENO-VI k-effective using homogenized cell | X | X | X | X |

To check the input data without actually processing the cross sections, the words "PARM=CHECK" or "PARM=CHK" should be entered, starting in or after column 11 of the analytical sequence specification.

```
=CSAS26  PARM=CHK
         or
#CSAS26  PARM=CHK
```

This would cause the input data for CSAS26 to be checked and appropriate error messages to be printed. If plots are specified in the data, they will be printed. This feature allows the user to debug and verify the input data while using a minimum of computer time. Many problems can be checked in 10 seconds or less. The number of words of storage requested by CSAS6 and its associated sequences is defaulted to 100,000 words of storage. This can be respecified through the use of the PARM command on the analytical sequence specification. For example, if 150,000 words of storage are required for the problem, "PARM=SIZE=150,000" should be entered, starting in or after column 11 of the analytical sequence specification as shown below.

```
=CSAS26  PARM=SIZE=150000
```

To combine checking and size specification, specify "PARM=(CHECK,SIZE=nnn)" or "PARM=(CHK,SIZE=nnn)" starting in or after column 11 on the analytical sequence specification. The value of nnn is given in words of storage. The minimum value of nnn is the amount of storage required by the problem, and the maximum value of nnn is the available computer storage.

### C6.4.1 INPUT DATA SUMMARY FOR EXPERIENCED USERS

Section C6.4.1 is provided for the convenience of experienced users. Other users should skip to Section C6.4.2. All of the input data for Criticality Safety Analysis Sequence No. 6 are illustrated in Fig. C6.4.1, and are summarized in Table C6.4.4. This table is provided as an input data guide for users who are familiar with CSAS6 data requirements. Each component table of Table C6.4.4 is repeated and discussed in detail in Sections C6.4.4 through C6.4.12. The KENO-VI data are included in this table in very general terms. Table C6.4.5 contains a summary of the KENO-VI input data. They are described in detail in Section F17.4 with additional information contained in Section2 F17.5.

The Material Information Processor now allows terse data input for all data except the standard composition names and the keyword entries in the optional parameter data (MORE DATA). Terse data entry requires the user to enter only as many characters as is necessary to uniquely define the data entry. However, as many characters as are specified by the user must match the name allowed by the code.

For example, refer to Table C6.4.4, **Outline of Material Information Processor data**, data position 3. The available choices for the type of calculation are: INFHOMMEDIUM, LATTICECELL, and MULTIREGION. Therefore, I would be sufficient to specify an infinite homogeneous medium, L would be sufficient for lattice cell, and M would specify multiregion. If it is desirable to use more characters, the user is free to do so. However, as many characters as are entered must match the name allowed by the code. For example, if INFHOMMEDIUM is to be specified, INFH is correct, but INFI is not.

WARNING: If several choices begin with the same letters, sufficient characters should be entered to uniquely define the name. For example, ASPHSQUAREP and ASPHTRIANGP are identical through the first four characters. If only ASPH is entered, the code will accept the first name whose first four characters are ASPH. This may not be the one the user intended to use.

Table C6.4.4 Input data requirements

**Summary of data input requirements** (see Sect. C6.4.2 for details)

| | Type of data | Analytical sequence | |
|---|---|---|---|
| 1 | Analytical sequence specification | =CSAS26 | =CSAS26X |
| 2 | Material Information Processor data | Yes | Yes |
| 3 | KENO-VI data | Yes | Yes |
| 4 | END for the analytical sequence | Yes | Yes |

**Outline of Material Information Processor data** (see Sect. C6.4.3 for details)

| Data position | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | TITLE | Enter a title | 80 characters |
| 2 | Cross-section library name | HANSEN-ROACH<br>27GROUPNDF4<br>44GROUPNDF5<br>238GROUPNDF5<br>27BURNUPLIB<br>etc. | The currently available libraries are listed in Table M8.8.2. |
| 3 | Type of calculation | INFHOMMEDIUM<br>LATTICECELL<br>MULTIREGION | These are the available options. See the explanation in Sect. C6.4.3. |
| 4 | Standard Composition specification data | Enter the appropriate data | Terminate this data block with END COMP. See Sect. C6.4.4. |
| 5 | Unit cell geometry specification | Enter the appropriate data (optional for INFHOMMEDIUM) | Optional for INFHOMMEDIUM.<br>See Sect. C6.4.5 for INFHOMMEDIUM.<br>See Sect. C6.4.6 for LATTICECELL.<br>See Section C6.4.7 for MULTIREGION. |
| 6 | Optional parameter data | Enter the desired data | Precede this data block by MORE DATA if more parameter data are to be entered. Otherwise, omit these data entirely. See Sect. C6.4.8. |

Table C6.4.4 (continued)

**Outline of standard compositions specification data** (see Sect. C6.4.4 for details)

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 1 | SC | Standard composition component name | Always | Enter once for each standard composition. Enter the alphanumeric description from Table M8.2.1. Additional allowed names include those beginning with ARBM for arbitrary materials and SOLN for solutions |
| A1 | ROTH | Theoretical density of material (g/cc) | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A2 | NEL | Number of elements in the material | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A3 | IVIS | No longer used but still must be entered | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1. |
| A4 | ICP | Compound indicator | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 1 for a compound, 0 for alloys, mixtures, etc. |
| A5 | IRS | No longer used but still must be entered | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1. |
| A6 | NCZA | ID number (from far right column of Table M8.2.1) | ARBM | Repeat the sequences A6 and A7 for each element in the arbitrary material before entering entry number 2. Enter the number from the far right column of Table M8.2.1. (Premixed standard compositions cannot be used in an arbitrary material definition) |
| A7 | ATPM | Number of atoms of this element per molecule of arbitrary material | ARBM & ICP=1 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry number 2. Do not enter a value unless ICP=1 |
| | | or | or | or |
| | | Weight percent of this element in this arbitrary material | ARBM & ICP=0 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry number 2. Do not enter a value unless ICP=0 |

Table C6.4.4 (continued)

**Outline of standard compositions specification data (continued)**

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 2 | MX | Mixture ID number | Always | Enter once for each standard composition component |
| S1 | FD | Fuel density (grams of U or Pu per liter of solution) | SOLN | Enter once for a solution |
| S2 | AML | Acid molarity of the solution | SOLN | Enter once for a solution. AML=0 if there is no acid in the solution |
| O1 | SPGR | Specific gravity of the solution | Optional | If the specific gravity (SPGR) of the solution is known, it should be entered as SPG=SPGR |
| | or | or | | or |
| | ROTH | Density of the basic standard composition | | if the density of a basic standard composition (ROTH) is to be entered, use DEN=ROTH |
| 3 | VF | Density multiplier | See comment column | Enter the density multiplier (density fraction, volume fraction, or a combination). Default value is 1. This item can be omitted if entries 4, 5, 6a, and 6b are also omitted. VF=0 is not allowed for SOLN or ARBM |
| 4 | ADEN | Number density (atoms/b-cm) for the nuclide | VF=0 | Enter only if VF=0.0 |
| 5 | TEMP | Temperature, in K | See comment column | Default value is 293 K. This entry can be omitted if entries 6a and 6b are also omitted |
| 6a | IZA | Isotope's ZA number | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 6b | WTP | Weight percent of the isotope | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0.0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 7[a] | END | Terminate a standard composition | Always | Enter once for each standard composition component. This terminates the data for a standard composition component. Enter END to terminate the component. Repeat entries 1 through 7 until all the mixtures have been defined. At least two blanks must separate entry 7 from the next entry |
| | END COMP | Terminate the data block | Terminus | Enter once for a problem. Enter the words END COMP when all the standard composition components have been described. At least two blanks must follow the END COMP |

[a]NOTE: Entry 7 should not begin in column 1 unless a name is associated with it. At least two blanks should separate the last entry 7 from the END COMP.

Table C6.4.4 (continued)

**Optional unit cell specifications for INFHOMMEDIUM problems** (see Sect. C6.4.5 for details)

| Entry number | Variable name | Type of data | Data entry | Comments |
|---|---|---|---|---|
| 1 | NAME | Keyword | CELLMIX | Initiate reading cell data for INFHOMMEDIUM |
| 2 | MFUEL | Mixture number in the cell | Mixture number | Specifies the mixture number to be used in the cell. Defaults to the smallest mixture number entered in the standard composition data |

NOTE: Unit cell data are necessary for INFHOMMEDIUM only if a mixture number other than the smallest mixture number is to be used in the cell and if a control sequence that executes XSDRNPM has been specified. If mixtures 1, 2, and 3 are specified and if mixture 3 is to be used in the cell, enter CELLMIX 3.

Table C6.4.4 (continued)

**Unit cell specification for LATTICECELL problems** (see Sect. C6.4.6 for details)

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CTP | Type of lattice | Always | | Describes the type of lattice or array configuration |
| | | | | SQUAREPITCH | Use for cylindrical rods in a square pitch |
| | | | | ASQUAREPITCH | Use for annular cylindrical rods in a square pitch |
| | | | | ASQP | Use for annular cylindrical rods in a square pitch |
| | | | | TRIANGPITCH | Use for cylindrical rods in a triangular pitch |
| | | | | ATRIANGPITCH | Use for annular cylindrical rods in a triangular pitch |
| | | | | ATRP | Use for annular cylindrical rods in a triangular pitch |
| | | | | SPHSQUAREP | Use for spherical pellets in a cubic lattice |
| | | | | ASPHSQUAREP | Use for annular spherical pellets in a cubic lattice |
| | | | | ASSP | Use for annular spherical pellets in a cubic lattice |
| | | | | SPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASTP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | SYMMSLABCELL | Use for a symmetric array of slabs |
| | | | | ASYMSLABCELL | Use for a periodic but asymmetric array of slabs |
| 2 | PITCH | Array pitch (cm) | Always | Appropriate dimension | The center-to-center spacing (cm) between fuel lumps. For asymmetric slab cell, enter the distance from the center of one moderator to the center of the other moderator (cm) |
| 3 | FUELOD | Outside dimension of fuel (cm) | Always | Appropriate dimension | Outside diameter of fuel (cm), or the thickness of the fuel in a slab |
| 4 | MFUEL | Fuel mixture number | Always | Mixture number | Mixture number representing the fuel |
| 5 | MMOD | Moderator mixture number | Always | Mixture number | Mixture number representing the moderator |
| 6 | MMOD2 | 2nd moderator mixture number | Annular cell | Mixture number | Mixture number representing the second moderator |
| 7 | TKMOD2 | 2nd moderator thickness | ASYMSLABCELL | Thickness | Thickness of the second moderator (cm) for ASYMSLABCELL |
| | | or | or | or | or |
| | | 2nd moderator diameter (cm) | annular cell | diameter | diameter of inner moderator (cm) for other annular cells |
| 8 | CLADOD | Outside diameter of clad (cm) | If clad | Clad OD | OMIT IF NO CLAD. For a slab, CLADOD is the sum of thickness of the fuel, gap, and clad |
| 9 | MCLAD | Clad mixture number | If clad | Mixture number | OMIT IF NO CLAD. Mixture number representing the clad |
| 10 | CLADID | Inside diameter of clad (cm) | If gap | Clad ID | OMIT IF NO GAP between the fuel and clad |
| 11 | MGAP | Gap mixture number | If gap | Mixture number | OMIT IF NO GAP between the fuel and clad. A mixture number of zero is often used |
| 12 | END | Terminate LATTICECELL data | Always | END | Terminate the LATTICECELL input data by entering the word END. Do not start in column 1. At least two blanks must follow entry number 12 |

Table C6.4.4 (continued)

**Unit cell specification for MULTIREGION problems (see Sect. C6.4.7 for details)**

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CS | Type of geometry | Always | | Describes the type of geometry. The options are: |
| | | | | SLAB | Use for slab geometry |
| | | | | CYLINDRICAL | Use for cylindrical geometry |
| | | | | SPHERICAL | Use for spherical geometry |
| | | | | BUCKLEDSLAB | Use for slab geometry with a buckling correction for the two transverse directions |
| | | | | BUCKLEDCYL | Use for cylindrical geometry with a buckling correction in the axial direction |
| 2 | BR | Right boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | | Default is VACUUM. Describes the right/outside boundary condition |
| | | | | VACUUM | This provides a non-return condition at the boundary |
| | | | | REFLECTED | Do not use for cylindrical or spherical |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | This provides isotopic return at the boundary |
| 3 | BL | Left boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | | Default is REFLECTED. Describes the left boundary condition |
| | | | | VACUUM | This provides a non-return condition at the boundary |
| | | | | REFLECTED | Recommended for cylindrical or spherical |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | This provides isotropic return at the boundary |
| 4 | ORGN | Location of left boundary on the x-axis (cm) | BUCKLEDSLAB and BUCKLEDCYL optional for other geometries | Appropriate dimension | Default is 0.0. Should not be changed for cylindrical or spherical geometry. A value must be entered if subsequent data are to be entered |
| 5 | DY | Buckling height (cm) | BUCKLEDSLAB and BUCKLEDCYL | Appropriate dimension | Omit for SLAB, CYLINDRICAL, and SPHERICAL. This corresponds to one of the transverse dimensions of an actual 3-D slab assembly or to the length of a finite cylinder |
| 6 | DZ | Buckling depth (cm) | BUCKLEDSLAB | Appropriate dimension | Omit unless BUCKLEDSLAB was specified. This is the buckling depth corresponding to the second transverse dimension of a 3-D slab assembly |
| 7 | END | End geometry parameters | Always | END | Enter the word END. Do not start in column 1. At least two blanks must separate entry 7 from the first entry 8 |
| 8 | MXZ | Mixture number in the zone | Always | Mixture number | Repeat entry numbers 8-10 until all zones are defined. Enter the mixture number for this zone |
| 9 | RZ | Outside radius of the zone (cm) | Always | Appropriate dimension | Repeat entry numbers 8-10 until all zones are defined. Enter the outside dimension of the zone (cm) |
| 10 | XMOD | External moderator index | Optional | | Repeat entry numbers 8-10 until all zones are defined. Entry 10 is optional and can be omitted. If it is omitted, repeat entry numbers 8 and 9 until all zones are defined |
| | | | | NOEXTERMOD | No moderating materials in the adjacent zones |
| | | | | ONEEXTERMOD | A moderating material is present in one adjacent zone |
| | | | | TWOEXTERMOD | Moderating materials are present in two adjacent zones |
| | END ZONE | Terminate zone data | | END ZONE | Enter when all zones have been defined by repeating entry numbers 8 through 10 for each zone. At least two blanks must follow this entry |

Table C6.4.4 (continued)

**Summary of available optional parameter data** (see Sect. C6.4.8)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 1 | MORE DATA | Input flag | | This signals that optional parameter data will be entered. Enter only those parameters you wish to change |
| 2 | ISN= | Order of angular quadrature | XSDRNPM | The default value is 8. This allows using another value |
| 3 | SZF= | Spatial mesh size factor | XSDRNPM | The default value is 1.0.<br>0<SZF<1.0 gives a finer mesh.<br>SZF>1.0 gives a coarser mesh |
| 4 | IIM= | Max. number of inner iterations | XSDRNPM | The default value is 20. This allows using another value |
| 5 | ICM= | Max. number of outer iterations | XSDRNPM | The default value is 25. This allows using another value |
| 6 | EPS= | Overall convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 7 | PTC= | Point convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 8 | BKL= | Buckling factor | XSDRNPM | The default value is 1.420892.<br>Use ONLY for a multiregion problem that specifies BUCKLEDSLAB or BUCKLEDCYL |
| 9 | IUS= | Upscatter scaling factor | XSDRNPM | The default value is zero.<br>IUS=0 doesn't utilize upscatter scaling. IUS=1 uses upscatter scaling to accelerate the solution and/or speed convergence |
| 10 | RES= | Resonance data | BONAMI NITAWL | Enter the mixture number, geometry type (SLAB, CYLINDER, SPHERE), and the thickness of the slab or radius of the sphere or cylinder in cm. Optionally enter the inner radius (cm) to specify an annular cylinder or sphere |
| 11 | DAN(mm)= | Dancoff factor for the specified mixture | BONAMI NITAWL | Enter the mixture number, mm, to which the Dancoff factor applies inside the parentheses; enter the Dancoff factor after the equal sign<br><br>**Repeat entry numbers 10 and 11 for all resonance mixtures used in the problem that are not treated in the LATTICECELL or MULTIREGION description** |

Table C6.4.4 (continued)

Summary of available optional parameter data (continued)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 12 | BAL= | Key to print balance tables | XSDRNPM | The default value is FINE. BAL=NONE suppresses printing the balance table. BAL=ALL prints all balance tables. BAL=FINE prints only the fine-group balance tables |
| 13 | DY= | First transverse dimension | XSDRNPM | The first transverse dimension, in cm, used in a buckling correction to calculate leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder) |
| 14 | DZ= | Second transverse dimension | XSDRNPM | The second transverse dimension, in cm, used for a buckling correction (i.e., the width of a slab) |
| 15 | COF= | Diffusion coefficient option for transverse leakage correction | XSDRNPM | The default is 0. See Sect. F3.5, 3$ array, variable IPN |
| 16 | FRD= | Unit from which fluxes will be read | XSDRNPM | Enter the unit number from which the flux guess for XSDRNPM will be read |
| 17 | FWR= | Unit on which fluxes will be written | XSDRNPM | Enter the unit number where the binary fluxes from XSDRNPM will be written |
| 18 | DAB= | Number of direct access data blocks | MIP[a] | The default is 200. Number of blocks allocated for direct access unit 90 |
| 19 | AXS= | Unit on which a mixed ANISN library will be written | ICE[a] | Enter the unit number where ICE is to write a mixed ANISN library |
| 20 | COLL | Key to activate collapse of thermal groups | MIP[a] | Enter COLL to collapse all thermal groups into one group for the shielding sequences |
| 21 | END | Terminus | | Terminate the optional parameter data. Do not start in column 1. At least two blanks must follow entry number 21 |

[a]MIP is the Material Information Processor.

Table C6.4.4 (continued)

**Outline of KENO-VI data** (see Table C6.4.5 and Sects. C6.4.9 and F17.4 for details)

| Type of data | Starting flag | Comments | Termination flag |
|---|---|---|---|
| Parameters | READ PARM | Enter desired parameter data | END PARM |
| Biasing or weighting | READ BIAS | Enter desired biasing data | END BIAS |
| Geometry | READ GEOM | Enter desired geometry data | END GEOM |
| Array data or unit orientation | READ ARRAY | Enter desired array data | END ARRAY |
| Boundary conditions or albedos | READ BNDS | Enter desired albedo data | END BNDS |
| Start data or initial source | READ START | Enter desired start data | END START |
| Mixing table data | READ MIXT | The Material Information Processor allows only NSCT= and PBXS= | END MIXT |
| Plot data | READ PLOT | Enter desired plot data | END PLOT |
| KENO-VI data terminus | END DATA | Enter to signal the end of all KENO-VI data | |

Table C6.4.5 Summary of KENO-VI input data (continued)

## Summary of parameter data

TITLE: The title must be entered first (80 columns) See Sect. F17.4.3

PARAMETERS: Format: READ PARAM enter parameter data here END PARAM
If parameters are entered, they must follow the title. See Sects. F17.4.3, F17.5.2, and F17.5.3.

| KEY | STD. | DEFINITION |
|---|---|---|
| RND= | Given | Random number |
| TME= | 120 min | Execution time (min) |
| TBA= | 0.5 min | Batch time (min) |
| WTA= | 0.5 | Average weight |
| WTH= | 3.0 | Wt. for splitting |
| WTL= | 1/WTH | Russian Roulette wt. |
| GEN= | 203 | No. of generations |
| NPG= | 1000 | No. per generation |
| NSK= | 3 | Generations skipped |
| RES= | 0 | Gens. between restart |
| NBK= | NPG+25 | Neutron bank positions |
| XNB= | 0 | Extra bank entries |
| NFB= | NPG | Fission bank positions |
| XFB= | 0 | Extra bank entries |
| X1D= | 0 | No. of extra 1-Ds |
| LNG= | 1000000 | Words of storage* |
| BEG= | 1 | Restart at this gen. |
| NG8= | 200 | Blocks for d.a. unit |
| NL8= | 512 | Length of d.a. block |
| SIG= | given | Standard deviation for problem termination |

| KEY | STD. | DEFINITION |
|---|---|---|
| RUN= | YES | Execute problem |
| FLX= | NO | Fluxes |
| FDN= | NO | Fission densities |
| ADJ= | NO | Adjoint calculation |
| AMX= | NO | All mixture xsecs |
| XAP= | NO | Xsec angles & probs. |
| XS1= | NO | 1-D xsecs |
| XS2= | NO | 2-D xsecs |
| PKI= | NO | Fission spectrum |
| P1D= | NO | Extra 1-D xsecs |
| FAR= | NO | Fiss. & abs. |
| MKP= | NO | Matrix by location |
| CKP= | NO | Cofactor k by loc. |
| FMP= | NO | Fiss. prod. by loc. |
| MKU= | NO | Matrix by unit |
| CKU= | NO | Cofactor k by unit |
| FMU= | NO | Fiss. prod. by unit |
| GAS= | NO | Fiss. & Abs by Reg. |

| KEY | STD. | DEFINITION |
|---|---|---|
| MKH= | NO | Matrix by hole |
| CKH= | NO | Cofactor k by hole |
| FMH= | NO | Fiss. prod. by hole |
| HHL= | NO | MKH at highest level |
| MKA= | NO | Matrix by array |
| CKA= | NO | Cofactor k by array |
| FMA= | NO | Fiss. prod. by array |
| HAL= | NO | MKA at highest level |
| PLT= | YES | Printer plots |
| BUG= | NO | Debug print |
| TRK= | NO | Print neutron tracks |
| PWT= | NO | Print avg. weight |
| PGM= | NO | Unprocessed geometry |
| SMU= | NO | Self-multiplication |
| NUB= | NO | Neutrons per fission |
| PAX= | NO | Albedo-xsec array |

| KEY | STD. | DEFINITION |
|---|---|---|
| XSC= | 14 | Mixed xsecs |
| ALB= | 79 | Albedo |
| WTS= | 80 | Weights |
| LTB= | 4 | Working xsecs from NITAWL or XSDRN |
| SKT= | 16 | Scratch |
| RST= | 95 | Read restart |
| WRS= | 95 | Write restart |
| XSC= | 14 | Mixed xsecs |

*The words of storage in CSAS are defaulted to 100,000 and can be overridden by using the size= parameter on the analytical sequence specification which then becomes the default value of LNG in KENO VI.
For example: =CSAS26 PARM=SIZE=200000
or =CSAS26 PARM=(CHECK,SIZE=200000)

Table C6.4.5 (continued)

## Summary of array data

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sects. F17.4.5, F17.5.6, and F17.5.7.

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

| ARRAY PARAMETERS | | | DATA TYPE |
|---|---|---|---|
| KEYWORD | DEFAULT | DEFINITION | FILL LOOP |
| ARA= | none | No. defining the array | |
| TYP= | cuboidal | array type (cuboidal or triangular) | |
| NUX= | none | No. of units in X direction | |
| NUY= | none | No. of units in Y direction | |
| NUZ= | none | No. of units in Z direction | |
| GBL= | none | Global array number** | |
| COM= | none | Delim comment delim optional comment is a maximum of 132 characters | |

**If specified, it need be entered only once per problem.

### ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data utilizing the options in this table, the count field and option field must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

| COUNT FIELD | OPTION FIELD | OPERAND FIELD | COMMENTS |
|---|---|---|---|
| | | j | Stores j at the current position in the array |
| i | R | j | Stores j in the next i positions in the array |
| i | * | j | Stores j in the next i positions in the array |
| i | $ | j | Stores j in the next i positions in the array |
| | F | j | Fills remainder of the array with unit no. j starting with the current array position |
| | A | j | Sets the current position in the array to j |
| i | S | | Increments the current position in the array by i allows skipping i positions. The value of i may be positive or negative |
| i | Q | j | Repeats the previous j entries i times. The default value of i is 1 |
| i | N | j | Repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1 |
| i | B | j | Starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1 |
| i | P | j | Alternately stores j and -j in the next i positions of the array |
| i | I | j k | Provides the end points, j and k, with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the I option should only be used to generate integer steps [i.e., (k-j)/(i+1) should be a whole number]. |
| | T | | Terminates the data reading for the array |

### ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP is terminated by entering END LOOP.

ENTER DATA IN THE FORM:

| DATA ENTRY | COMMENTS |
|---|---|
| LTYPE | The unit or box type. LTYPE must be greater than 1 |
| IX1 | Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX |
| IX2 | Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX |
| INCX | The number of units by which increments are made in the X direction |
| IY1 | The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY |
| IY2 | Ending position int he Y direction. IY2 must be at least 1 and no larger than the value of NUY |
| INCY | The number of units by which increments are made in the positive Y direction |
| IZ1 | Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ |
| IZ2 | Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ |
| INCZ | The number of units by which increments are made in the positive Z direction |

Table C6.4.5 (continued)

## Summary of biasing data

BIAS (weighting)

Format: READ BIAS keyword correlation data auxiliary END BIAS
See Sects. F17.4.7 and F17.5.8

| KEYWORD | DESCRIPTION | |
|---|---|---|
| ID= | CORRELATION DATA will be read next. | |
| | id | Material ID. Enter from table at right to use weighting data from the library |
| | ibgn | Beginning bias ID |
| | iend | Ending bias ID |
| WT= | AUXILIARY DATA will be read next. | |
| WTS= | AUXILIARY DATA will be read next. | |
| | wttitl | Material title (12-character maximum) |
| | id | Material ID |
| | nsets | Number of sets of group structures |
| | REPEAT | (thkinc, numinc, ngpwt, wtavg) nsets times |
| | thkinc | Thickness per increment |
| | numinc | Number of increments |
| | ngpwt | Number of energy groups for this set of wts |
| | wtavg | Enter numinc x ngpwt values of wtavg |

| MATERIAL | ID | ENERGY GROUPS | THICKNESS/ INCREMENT |
|---|---|---|---|
| Concrete | 301 | 16,27,44,218,238 | 5 cm |
| Paraffin | 400 | 16,27,44,218,238 | 3 cm |
| Water | 500 | 16,27,44,218,238 | 3 cm |
| Graphite | 6100 | 16,27,44,218,238 | 20 cm |

For CORRELATION DATA, the material ID is chosen from material ID column above (the keyword is ID=).
For AUXILIARY DATA, the material ID is chosen by the user and the keyword is WT= or WTS=. When AUXILIARY DATA are entered, CORRELATION DATA must also be entered to use the data.
Beginning and ending bias IDs are defined by the user. The geometry specification that has the bias ID equal to the beginning bias ID utilizes the wtavg's from the first interval of material ID.

Table C6.4.5 (continued)

## Summary of boundary condition data

BNDS (albedo or boundary conditions)

Format: READ BNDS face code albedo name END BNDS
See Sect. F17.4.7

The sequence FACE CODE ALBEDO NAME is entered as many times as necessary to define the appropriate albedo boundary conditions. The default for all faces is vacuum.

FACE CODES FOR ENTERING BOUNDARY (ALBEDO) CONDITIONS

| FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION |
|---|---|---|---|---|---|---|---|
| +XB= | Positive X face | XFC= | Both X faces | +YX= | Positive X and Y faces | &ZY= | Positive Y and Z faces |
| | Positive X face | YFC= | Both Y faces | &XY= | Positive X and Y faces | -XY= | Negative X and Y faces |
| | Negative X face | ZFC= | Both Z faces | &YX= | Positive X and Y faces | =XZ= | Negative X and Z faces |
| | Positive Y face | +FC= | All positive faces | +XZ= | Positive X and Z faces | =YZ= | Negative Y and Z faces |
| | Positive Y face | &FC= | All positive faces | +ZX= | Positive X and Z faces | YXF= | All X and Y faces |
| | Negative Y face | -FC= | All negative faces | &XZ= | Positive X and Z faces | ZXF= | All X and Z faces |
| | Positive Z face | XYF= | All X and Y faces | &ZX= | Positive X and Z faces | ZYF= | All Y and Z faces |
| | Positive Z face | XZF= | All X and Z faces | +YZ= | Positive Y and Z faces | -YX= | Negative X and Y faces |
| | Negative Z face | YZF= | All Y and Z faces | +ZY= | Positive Y and Z faces | -ZX= | Negative X and Z faces |
| | All 6 faces | +XY= | Positive X and Y faces | &YZ= | Positive Y and Z faces | -ZY= | Negative Y and Z faces |

Any albedo condition on any face is allowed if the global unit boundary record consists of only one geometry record that is a cuboid.
Any albedo condition, but the same condition on all faces, is allowed if the global unit boundary record consists of one geometry record that is composed of paired planes.
Only void, mirror, or white boundary conditions, the same condition on all faces, if the global unit boundary record consists of only one geometry record of any shape.
Only void boundary condition is allowed if the global unit boundary record contains multiple geometry records. Default is void on all global unit boundary surfaces.

ALBEDO NAMES AVAILABLE ON THE KENO ALBEDO LIBRARY, FOR USE WITH THE FACE CODES

| ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION |
|---|---|---|---|---|---|
| DPOH20 | 12-in. double PO water | CONC-4 | 4-in. concrete differential | VACUUM | Vacuum condition |
| DPOH20 | differential albedo with | CON4 | albedo with 4 incident angles | VOID | |
| DPO | 4 incident angles | CONC4 | | VACU | |
| DPO | | | | VAC | |
| | | CONC-8 | 8-in. concrete differential | | |
| H20 | 12-in. water differential | CON8 | albedo with 4 incident angles | SPECULAR | Mirror image reflection |
| WATER | albedo with 4 incident angles | CONC8 | | MIRROR | |
| | | | | MIRR | |
| PARAFFIN | 12-in. paraffin differential | CONC-12 | 12-in. concrete differential | SPEC | |
| PARA | albedo with 4 incident angles | CON12 | albedo with 4 incident angles | SPE | |
| WAX | | CONC12 | | MIR | |
| CARBON | 200-cm carbon differential | CONC-16 | 16-in. concrete differential | PERIODIC | Periodic boundary condition |
| GRAPHITE | albedo with 4 incident angles | CON16 | albedo with 4 incident angles | PERI | |
| C | | CONC16 | | PER | |
| ETHYLENE | 12-in. polyethylene | CONC-24 | 24-in. concrete differential | WHITE | White boundary Condition |
| POLY | differential albedo with | CON24 | albedo with 4 incident angles | | |
| CH2 | 4 incident angles | CONC24 | | | |

Table C6.4.5 (continued)

## Summary of geometry data

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM
See Sects. F17.4.4, F17.5.1.2, F17.5.6, and F17.5.7.
GEOMETRY REGION DATA consists of SIMPLE GEOMETRY REGION DATA and MODIFICATION GEOMETRY REGION DATA.
ENTER GEOMETRY REGION DATA IN THE FOLLOWING FORM:
OPTIONAL GLOBAL SPECIFICATION
UNIT n
OPTIONAL GEOMETRY COMMENT
GEOMETRY, MODIFICATION, MEDIA, and BOUNDARY DATA
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
DATA IN THE FOLLOWING FORM:
GLOBAL Enter once to specify this unit as the global unit.
UNIT n
COM=delim comment delim This optional comment can be up to 132 characters. It must begin and end with a delimiter.
Geometry Keyword Label Input Parameters Optional origin data (ORIGIN) Optional chord data (CHORD) Optional rotation data (ROTATE degrees)
Media Keyword Input Parameters
Boundary Keyword Input Parameters Each unit must have a boundary record.
Enter as many geometry and media descriptions as necessary to describe the unit and as many units as necessary to describe the system.

GEOMETRY REGION INPUT DATA REQUIREMENTS

| TYPE OF DATA | KEYWORD | IPUT PARAMETERS | NOTES |
|---|---|---|---|
| GEOMETRY | CONE | Lbl Rt Zt Rb Zb | Lbl is an integer assigned to a geometry record, unique within the unit, that relates the geometry record to the media region in the Region Definition Vector I1 I2 ... |
| | CUBOID | Lbl +X -X +Y -Y +Z -Z | |
| | CYLINDER | Lbl R Zt Zb | |
| | DODECAHEDRON | Lbl R | |
| | ECYLINDER | Lbl Rx Ry Zt Zb | |
| | ELLIPSOID | Lbl Rx Ry Rz | |
| | HEXPRISM | Lbl R Zt Zb | |
| | HOPPER | Lbl Lxt Lyt Zt Lxb Lyb Zb | |
| | PARALLELPIPED | Lbl XDIST YDIST ZDIST PHI THETA PHI | |
| | PLANE* | XPL=a YPL=b ZPL=c CON=d | Data is entered after the = following the respective subordinate keyword. Only non-zero data is required. |
| | QUADRATIC* | Lbl AQU=a BQU=b CQU=c DQU=d EQU=e FQU=f GQU=g HQU=h IQU-i JQU=j | Data is entered after the = following the respective subordinate keyword. Only non-zero data is required. |
| | RHOMBOID | Lbl DX PSI | |
| | SPHERE | Lbl R | |
| | WEDGE | Lbl XBASE XPT YPT ZLNG | |
| MODIFICATION* | CHORD | +X= -X= +Y= -Y= +Z= -Z= | Only non-zero data is required. |
| | ORIGIN | X= Y= Z= | Only non-zero data is required. |
| | ROTATE | A1= A2= A3= | The body is rotated about the Origin using the Euler angle x-convention. Only non-zero data is required. |
| MEDIA** | MEDIA | Mid Bld I1 I2 ... | |
| | HOLE | Hid I1 I2 ... | The region boundary containing the hole must exactly match the boundary of the unit being placed in the hole. |
| | ARRAY | Aid I1 I2 ... PLACE nx ny nz xx yy zz | PLACE positions the array by placing the origin of the unit in array position nx ny nz at xx yy zz. The region boundary containing the array must be enclosed in or on the array boundary |
| BOUNDARY | BOUNDARY | I1 I2 ... | Defines unit boundary. |

*Modification Data can be entered in any order. Any combination of modification data can follow any geometry record.
**The Region Definition Vector, I1, I2, ..., should contain references to all geometry records which share a boundary with the region being defined.

Table C6.4.5 (continued)

## Summary of mixing table data

MIXTURES Format: READ MIXT xsec parameters END MIXT

Note: In CSAS6, only xsec parameters should be entered.
KENO-VI mixing table data consists of "xsec parameters" and a "mixing table." The "mixing" table" is used to define the materials used in the KENO-VI problem. This data is automatically created by CSAS6 and provided to KENO-VI. It cannot be overridden.
See Sects. F17.4.10 and F17.5.5.

XSEC PARAMETERS consists of keywords and associated values.
These parameters, if entered, need be entered only once.

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| SCT= | 1 | No. of discrete scattering angles<br>0 is isotropic<br>1 is P1<br>2 is P3<br>3 is P5 |
| EPS= | 0.00003 | Cross-section message cutoff value; use to suppress message No. K6-60 |

## Table C6.4.5 (continued)

## Summary of plot data

PLOT — Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made. See Sects. F17.4.1 and F17.5.9

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| TTL= | Prob. title | delim ptitl delim delim is a one-character delimiter that signals the beginning and end of the title. ptitl is the plot title (max. 132 char.) |
| PIC= | MAT | Type of picture: MIXTURE, UNIT NO. or BIAS ID NO.<br>MIXTURE ---------- MAT<br>MIX<br>MIXT<br>MIXTURE<br>MEDI<br>MEDIA<br>UNIT NO. ---------- BOX<br>BOXT<br>BOXTYPE<br>UNT<br>UNIT<br>UNITTYPE<br>BIAS ID NO. ------- IMP<br>BIAS<br>BIASID<br>WTS<br>WEIG<br>WEIGHTS<br>WGT<br>WGTS |
| XUL= | Prev. plot | X coord. of upper left corner of plot |
| YUL= | Prev. plot | Y coord. of upper left corner of plot |
| ZUL= | Prev. plot | Z coord. of upper left corner of plot |
| XLR= | Prev. plot | X coord. of lower right corner of plot |
| YLR= | Prev. plot | Y coord. of lower right corner of plot |
| ZLR= | Prev. plot | Z coord. of lower right corner of plot |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| UAX= | Prev. plot 0 IF VAX OR WAX is read | X component of direction cosine for the AX axis of the plot (across) |
| VAX= | Prev. plot 0 IF UAX OR WAX is read | Y component of direction cosine for the AX axis of the plot (across) |
| WAX= | Prev. plot 0 IF UAX OR VAX is read | Z component of direction cosine for the AX axis of the plot (across) |
| UDN= | Prev. plot 0 IF VDN OR WDN is read | X component of direction cosine for the DN axis of the plot (down) |
| VDN= | Prev. plot 0 IF UDN OR WDN is read | Y component of direction cosine for the DN axis of the plot (down) |
| WDN | Prev. plot 0 IF UDN OR VDN is read | Z component of direction cosine for the DN axis of the plot (down) |
| DLX= | | Horizontal spacing between points on plot |
| DLD= | | Vertical spacing between points on plot |
| NAX= | | No. of intervals to be printed across page |
| NDN= | | No. of intervals to be printed down page |
| LPI= | 8 | Lines per inch printed down the page |
| NCH= | CHRS* | Delim CHRS delim a one-character delimiter signals the beginning and end of the character string |
| RUN= | YES | YES allows the problem to execute<br>NO terminates problem after data checking |
| PLT= | YES | YES allows the plot(s) to be made<br>NO allows reading the plot data without making a plot |
| SCR= | NO | No specifies a printer plot. If YES, a .gif file is generated for each plot. |
| CLR= | | Used only with SCR=YES to change colors. Input 4 integers: 1st is index into color table. next 3 are the red, green, and blue color values. |

PLOT ORIGIN:

(1) SINGLE UNIT - coincides with origin of geometry description.

(2) BASE ARRAY - at the most negative point of the array (lower left-hand back corner of the global array).

(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

*Default values of CHRS are given below:

| | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
| CHRS | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F | G | H | I | J | K | L | M |

| | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | | |
| CHRS | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | # | , | $ | - | + | ) | \| | | |

| | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MEDIA | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | | | | | | |
| CHRS | £ | > | : | ; | . | - | % | * | " | = | ! | ( | @ | < | / | 0 | | | | | | |

Table C6.4.5 (continued)

## Summary of starting data

START Format: READ START enter start data here END START
The default value of start type is zero. See Sect. F17.4.8.

| START TYPE | REQUIRED DATA | OPTIONAL DATA | STARTING DISTRIBUTION |
|---|---|---|---|
| 0 | None | NST | Uniform |
| | | XSM | |
| | | XSP | |
| | | YSM | |
| | | YSP | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 1 | NST | XSM | Cosine |
| | | XSP | |
| | | YSM | |
| | | YSPO | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 2 | NST | XSM | Cosine with |
| | NXS | XSP | fraction in |
| | NYS | YSM | specified |
| | NZS | YSP | unit |
| | FCT | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 3 | NST | KFS | Spike |
| | TFX | PSP | |
| | TFY | | |
| | TFZ | | |
| | NXS | | |
| | NYS | | |
| | NZS | | |
| 4 | NST | KFS | Multiple |
| | TFX | PSP | spikes |
| | TFY | | |
| | TFZ | | |
| | NBX | | |
| 5 | NST | PSP | In specified |
| | NBX | | units |
| 6 | NST | NXS | Arbitrary |
| | TFX | NYS | points |
| | TFY | NZS | |
| | TFZ | KFS | |
| | LNU* | PS6 | |
| | | PSP | |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| NST= | 0 | Start type |
| TFX= | 0.0 | X coordinate |
| TFY= | 0.0 | Y coordinate |
| TFZ= | 0.0 | Z coordinate |
| NXS= | 0 | X index of unit pos. |
| NYS= | 0 | Y index of unit pos. |
| NZS= | 0 | Z index of unit pos. |
| KFS= | | Fission spectra |
| LNU= | 0 | Number of last neutron |
| NBX= | 0 | Source unit number |
| FCT= | 0 | Fraction |
| XSM= | -X | -X of source cuboid |
| XSP= | +X | +X of source cuboid |
| YSM= | -Y | -Y of source cuboid |
| YSP= | +Y | +Y of source cuboid |
| ZSM= | -Z | -Z of source cuboid |
| ZSP= | +Z | +Z of source cuboid |
| RFL= | NO | Start in reflector |
| PS6= | NO | Print start 6 input |
| PSP= | NO | Print starting points |

*LNU must be the last entry for each set of start 6 data. The LNU of each successive set of data must be larger than the last.
Unless otherwise specified, the starting volume is defined by the first positive geometry label in the global unit boundary region definition vector.

## C6.4.2 INPUT DATA REQUIREMENTS AND DATA CHECKING

Table C6.4.7 summarizes the input data required by Criticality Safety Analysis Sequence No. 6. The types of data needed are:

Table C6.4.7 Summary of data input requirements

| | Type of Data | Analytical Sequence | |
|---|---|---|---|
| 1 | Analytical Sequence Specification | =CSAS26 | =CSAS26x |
| 2 | Material Information Processor Data | Yes | Yes |
| 3 | KENO-VI Data | Yes | Yes |
| 5 | Search Data | No | No |
| | END for the Analytical Sequence | Yes | Yes |

1. Analytical sequence specification. Enter the desired sequence name, starting in column 1. The available options are =CSAS26, =CSAS26X. Enter PARM=CHK in column 11 to check the data.

2. Material Information Processor data. These data are required for processing the cross sections for use in the problem. These data are described in detail in Sects. M7.4 and C6.4.3 through C6.4.8.

3. KENO-VI data. These data are described briefly in Sect. C6.4.9. A detailed description is found in Sect. F17.4 with additional information in Sect. F17.5.

4. Search data. There is currently no search options available using KENO-VI.

5. END for analytical sequence—Enter END, starting in column 1.

To check the input data, specify the desired analytical sequence and specify PARM=CHECK or PARM=CHK starting beyond column 10. For example, to check CSAS26 data, the analytical sequence specification would be:

```
=CSAS26     PARM=CHECK
```

Note that =CSAS26 starts in column 1 and PARM=CHECK starts in or after column 11. This causes the input data to be checked, the problem description to be printed, appropriate warning and error messages to be printed, and if specified, the KENO-VI data and printer plots are printed.

This feature allows checking the input data without executing any of the functional modules. Simply remove the PARM=CHECK to actually execute the problems.

### C6.4.3 MATERIAL INFORMATION PROCESSOR DATA

The Material Information Processor reads the standard compositions specification data and the unit cell geometry specifications. It then produces the mixing table and unit cell information necessary for processing the cross sections. Table C6.4.8 outlines the input data for the Material Information Processor.

The Material Information Processor allows terse data input for all data except the standard composition names and the keyword entries in the optional parameter data (MORE DATA). Terse data entry requires the user to enter only as many characters as are necessary to uniquely define the data entry. However, as many characters as are specified by the user must match the name allowed by the code. For example, refer to Table C6.4.8, **Outline of Material Information Processor data**, data position 3. The available choices for the type of calculation are INFHOMMEDIUM, LATTICECELL, and MULTIREGION. Therefore, it would be sufficient to specify I for an infinite homogeneous medium, L for lattice cell, and M for multiregion. If it is desirable to use more characters, the user is free to do so. However, as many characters as are entered must match the name allowed by the code. For example, if INFHOMMEDIUM is to be specified, INFH is correct, but INFI is not.

WARNING: If several choices begin with the same letters, sufficient characters should be entered to uniquely define the name. For example, ASPHSQUAREP and ASPHTRIANGP are identical through the first four characters. If only ASPH is entered, the code will accept the first name whose first four characters are ASPH. This may not be the one the user intended to use.

The types of data required for the Material Information Processor are given in Table C6.4.8. The individual entries are explained in detail in the text following the table.

1. Title. An 80-character title is required. The title is the first 80 characters of the Material Information Processor data.

2. Cross-Section Library Name. This item specifies the cross-section library that is to be used in the calculation. See Section C6.4, Table C6.4.8, or Section M8, Table M8.2.2, for a list of the available libraries.

3. Type of Calculation. The options are INFHOMMEDIUM, LATTICECELL, and MULTIREGION.

   INFHOMMEDIUM is used for an infinite homogeneous medium. It provides cross sections to represent large homogeneous pieces. The cross sections are treated as if each mixture is infinite. Thus, the self-shielding calculations will not account for any geometrical effects. By default, the smallest mixture number will be used in the cell. A different mixture can be used as described in Section C6.4.5 and Table C6.4.10.

   LATTICECELL is used when the geometry can be described as a lattice. It is especially suited for arrays of cylindrical rods or spherical pellets. The use of LATTICECELL requires the entry of additional geometry information (unit cell specification), which is described in Section C6.4.6 and Table C6.4.11. When the problem consists of an array of fuel bundles or similar geometry, the cross sections are corrected for resonance self-shielding, including geometry effects. If cell-weighted cross sections are desired, CSAS26X must be used. A flux disadvantage factor

Table C6.4.8. Outline of Material Information Processor data

| Data position | Type of data | Data entry | Comments |
|---|---|---|---|
| 1 | Title | Enter a title | 80 characters |
| 2 | Cross-Section Library Name | HANSEN-ROACH<br>27GROUPNDF4<br>44GROUPNDF5<br>238GROUPNDF5<br>27BURNUPLIB<br>etc. | The currently available libraries are listed in Tables C6.4.4 and C6.4.8. |
| 3 | Type of Calculation | INFHOMMEDIUM<br>LATTICECELL<br>MULTIREGION | These are the available options. See the explanation in Sect. C6.4.3 |
| 4 | Standard Composition Specification Data | Enter the appropriate data | Terminate this data block with END COMP. See Sect. C6.4.4 |
| 5 | Unit Cell Geometry Specification | Enter the appropriate data (optional for INFHOMMEDIUM) | Optional for INFHOMMEDIUM<br>See Sect. C6.4.5 for INFHOMMEDIUM<br>See Sect. C6.4.6 for LATTICECELL<br>See Sect. C6.4.7 for MULTIREGION |
| 6 | Optional Parameter Data | Enter the desired data | Precede this data block by MORE DATA if more parameter data are to be entered. Otherwise, omit these data entirely. See Sect. C6.4.8 |

is then applied to the cross sections of the materials that are used in the unit cell. The KENO-VI geometry description may represent a fuel bundle discretely or as a homogeneous region. By using CSAS26X a fuel bundle can be represented as a single homogeneous region by specifying a mixture number of 500 on the KENO-VI geometry card that defines the overall size and shape of the bundle. This utilizes the cell-weighted cross sections that represent the spatial behavior of the unit cell. The use of CSAS26X precludes using individual mixtures from the unit cell in the KENO-VI geometry.

MULTIREGION may be used to define a geometric configuration that is more complicated than that allowed by LATTICECELL. It can also be used for a system involving large geometric regions where geometry effects may be minimal. The cross sections utilized in the unit cell are corrected for resonance self-shielding, and an appropriate Dancoff factor is applied (see Section M7.2.4.3). If a cell-weighted mixture is generated, it is derived from homogenizing the entire MULTIREGION unit cell. The additional geometry information (unit cell specification) required for a MULTIREGION problem is described in Section C6.4.7 and Table C6.4.12.

4. Standard Compositions. These data are used to define the mixtures that will be used in the problem. See Section C6.4.4, and Table C6.4.9, for a description of the standard compositions specification data. These data are required for every problem.

5. Cell Geometry Specification. See Section C6.4.5 and Table C6.4.10 for an explanation of the optional unit cell data associated with an INFHOMMEDIUM problem. See Section C6.4.6 and Table C6.4.11 for an explanation of the data associated with LATTICECELL problems. Section C6.4.7 and Table C6.4.12 explain the data required for a MULTIREGION problem.

6. Optional Parameter Data. This option allows certain defaulted parameters to be respecified by the user. Omit these data unless they are needed. If data are to be entered, see Section C6.4.8 and Table C6.4.13 for assistance.

## C6.4.4 STANDARD COMPOSITIONS SPECIFICATION DATA

The standard compositions specification data are used to define mixtures that are to be utilized in the problem. These data can be considered to be a mixing table that defines mixtures in terms of standard compositions specification data (available from Section M8.2, Table M8.2.1). The required input for the standard compositions specification data varies, depending on the standard composition component name. However, every standard composition specification must include the following:

1. a standard composition component name (SC),

2. the mixture that contains this component (MX),

3. a terminator for the standard compositions specification data (enter the word END).

Enter as many standard composition specifications as necessary to define all the mixtures needed for the problem. When all the necessary mixtures have been described, enter the words END COMP to signal the completion of this type of data.

The data required to define a standard composition specification are contained in Table C6.4.9. In the column entitled "Entry requirement," "ARBM" is used to denote entries required only for arbitrary materials, and "SOLN" is used for entries required only for solutions.

The input data for standard composition specifications are arranged with entry numbers 1 through 7 that apply to all standard composition data. Entries A1 through A7 apply only to arbitrary materials. Entries S1 and S2 apply only to solutions. Entry O1 is optional data that applies to both basic standard compositions and solutions. The individual entries specified in Table C6.4.9 are explained in detail in the text accompanying the table.

1. SC Standard Composition Component Name. The names of the standard composition components (the alphanumeric identifiers) may be (1) chosen from Table M8.2.1 of Section M8.2, which contains the list of elements, compounds, and alloys found in the Standard Composition Library, (2) they may be chosen from the table of available solutions, Section M8.3, or (3) they may be designated as an arbitrary material (ARBM). An error message will be printed if the user enters an invalid standard composition component name.

   For standard compositions taken from Table M8.2.1 of the Standard Composition Library (basic standard compositions), certain data such as the volume fraction or theoretical density and other engineering type data may need to be entered. For standard compositions containing more than one isotope of an element (such as $UO_2$), the user is free to specify the weight percent for each isotope such that they total 100%. See Section M7.5.5.1 for examples of basic standard compositions.

Table C6.4.9 Outline of standard composition specification data

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 1 | SC | Standard composition component name | Always | Enter once for each standard composition. Enter the alphanumeric description from Table M8.2.1. Additional allowed names include those beginning with ARBM for arbitrary materials and SOLN for solutions |
| A1 | ROTH | Theoretical density of material (g/cc) | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A2 | NEL | Number of elements in the material | ARBM | Enter once for each standard composition component that is an arbitrary material |
| A3 | IVIS | No longer used but must still be entered. | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1. |
| A4 | ICP | Compound indicator | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 1 for a compound, 0 for alloys, mixtures, etc. |
| A5 | IRS | No longer used but must still be entered. | ARBM | Enter once for each standard composition component that is an arbitrary material. Enter 0 or 1. |
| A6 | NCZA | ID number (from far right column of Table M8.2.1) | ARBM | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry 2. Enter the number from the far right column of Table M8.2.1. (Premixed standard compositions cannot be used in an arbitrary material definition) |
| A7 | ATPM | Number of atoms of this element per molecule of arbitrary material<br>or<br>weight percent of this element in this arbitrary material | ARBM and ICP=1<br>or<br>ARBM and ICP=0 | Repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry 2.<br>Do not enter a value unless ICP=1<br>or<br>repeat the sequence A6 and A7 for each element in the arbitrary material before entering entry 2. Do not enter a value unless ICP=0 |

Table C6.4.9 (continued)

| Entry number | Variable name | Type of data | Entry requirement | Comments |
|---|---|---|---|---|
| 2 | MX | Mixture ID number | Always | Enter once for each standard composition component |
| S1 | FD | Fuel density (grams of U or Pu per liter of solution) | SOLN | Enter once for a solution |
| S2 | AML | Acid molarity of the solution | SOLN | Enter once for a solution. AML=0 if there is no acid in the solution |
| O1 | SPGR<br>or<br>ROTH | Specific gravity of the solution<br>or<br>density of the basic standard composition | Optional | If the specific gravity (SPGR) of the solution is known, it should be entered as SPG=SPGR<br>or<br>if the density of a basic standard composition (ROTH) is to be entered, use DEN=ROTH |
| 3 | VF | Density multiplier | See comment column | Enter the density multiplier (density fraction, volume fraction, or a combination). Default value is 1. This item can be omitted if entries 4, 5, 6a, and 6b are also omitted. VF=0 is not allowed for SOLN or ARBM |
| 4 | ADEN | Number density (atoms/b-cm) for the nuclide | VF=0 | Enter only if VF=0.0 |
| 5 | TEMP | Temperature, in K | See comment column | Default value is 293 K. This entry can be omitted if entries 6a and 6b are also omitted |
| 6a | IZA | Isotope's ZA number | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 6b | WTP | Weight percent of the isotope | VF≠0 | Enter for each isotope in the standard composition component. Omit if VF=0. Entries 6a and 6b are entered in pairs until each isotope in the component is defined |
| 7[a] | END | Terminate a standard composition | Always | Enter once for each standard composition component. This terminates the data for a standard composition component. Enter END to terminate the component. Repeat entries 1 through 7 until all the mixtures have been defined. At least two blanks must separate entry 7 from the next entry |
| | END COMP | Terminate the data block | Terminus | Enter once for a problem. Enter the words END COMP when all the standard composition components have been described. At least two blanks must follow the END COMP |

[a]NOTE: Entry 7 should not begin in column 1 unless a name is associated with it. At least two blanks should separate the last entry 7 from the END COMP.

Solutions require the specification of the density of the heavy metal in the solution and the acid molarity of the solution. The alphanumeric specification of a solution starts with the four characters "SOLN." Some common fissile solutions are SOLNUO2(NO3)2, SOLNPU(NO3)4, and SOLNUO2F2. See Section M8.3 for a complete listing of the available solutions. Sections M7.5.5.3 and M7.5.9 contain examples of the input data for solutions.

Arbitrary materials require the user to provide all the information normally found in the Standard Composition Library. The arbitrary material option allows specifying a compound or alloy not available in the Standard Composition Library by utilizing the standard compositions available in the library. Table M8.2.1 indicates which nuclides require a resonance self-shielding calculation.

An arbitrary material name must start with the four characters "ARBM." A maximum of twelve characters is allowed for the standard composition component name. Imbedded blanks are not allowed in a standard composition component name. Section M7.5.5.2 contains examples of input data for arbitrary materials.

An additional example of an arbitrary material specification is as follows:

Consider a mixture of boral having 35 wt % $B_4C$, 65 wt % aluminum and an overall density of 2.64 g/cc. If neither BORAL nor B4C were available in the Standard Composition Library, the components BORON, C, and AL could be used as basic standard compositions to describe the boral. The necessary volume fractions would have to be determined as shown in Section M7.5.5.4, examples 3 and 4. By utilizing the arbitrary material option, both $B_4C$ and aluminum can be described as a component of the same mixture (assume it is desired to be mixture 3), both have a density of 2.64 g/cc, and the volume fractions are 0.35 for the $B_4C$ and 0.65 for the aluminum. The resulting standard compositions specification data do not require any calculations (see example 1, Section M7.5.5.5 for a similar situation) and could be entered as

```
ARBMTL-B4C  2.64 2 1 1 0 5000 4 6012 1 3 0.35 END
ARBMTL-AL   2.64 1 0 0 0 13027 100.0 3 0.65 END
```

A1. ROTH Theoretical density of material. Enter ONLY for arbitrary materials. The density of the arbitrary material is entered in units of grams per cubic centimeter.
NOTE: ROTH can be entered for a basic standard composition as shown by entry O1.

A2. NEL Number of elements in the material. Enter ONLY for arbitrary materials. Enter the number of components from the Standard Composition Library that are to be used to define this arbitrary material. In the above example, the ARBMTL-B4C contains two components, the B and the C. Therefore, NEL is 2 for the $B_4C$ arbitrary material. The ARBMTL-AL contains only one component; so NEL is 1 for it.

A3. IVIS Unused Variable. Enter ONLY for an arbitrary material. Enter 0 or 1.

A4. ICP Compound indicator. Enter ONLY for arbitrary materials. A compound is a substance containing more than one constituent element and having properties different from those which their constituents had as elementary substances. The composition of a pure compound is perfectly definite and is always the same, regardless of the method of formation. Examples of

compounds are $H_2O$, $B_4C$, $CO_2$, $Ca(OH)_2$, $C_2H_6$, $Fe_2S_3$, $Cr_2O_3$, etc. ICP is 1 for a compound. ICP is 0 for alloys (stainless steel, monel, etc.) and conglomerates or mixtures such as concrete.

In the $B_4C$ arbitrary material example, ICP is 1; in the aluminum arbitrary material example, ICP is 0. These examples are given in the explanation of entry number 1, SC.

A5. IRS Unused Variable. Enter ONLY for arbitrary materials. Enter 0 or 1.

A6. NCZA ID number. Enter ONLY for arbitrary materials. This is the "ZA" ID number for the element or isotope. Usually, NCZA=A+1000*Z, where A is the atomic mass or weight of the nuclide, and Z is the atomic number. For example, NCZA is 1001 for hydrogen, 8016 for oxygen, and 92235 for $^{235}U$. For elements for which the natural abundance is acceptable, NCZA can be entered as 1000*Z. For example, 92000 is used for uranium and 5000 is used for boron.

Arbitrary materials may now contain more than one element that has multiple isotopes. Multiple isotope elements need not be the first element specified in the arbitrary material. The multiple isotope element MUST be listed in the Isotope Distribution Table, Table M8.4.1. Uranium (92000), plutonium (94000), and boron (5000) are multiple isotope elements.

For example:
ARBMTL-B4C 2.64 2 1 1 0 5000 4 6012 1 3 0.35 END is correct;
ARBMTL-B4C 2.64 2 1 1 0 6012 1 5000 4 3 0.35 END is also correct even though the multiple isotope element is not listed first.

REPEAT THE SEQUENCE NCZA and ATPM (A6 and A7) FOR EVERY ELEMENT IN THE ARBITRARY MATERIAL.

A7. ATPM Enter ONLY for arbitrary materials. Repeat the sequence A6 and A7 for every element in the arbitrary material before going on to entry 2.

If ICP is 1 (entry A4), enter the NUMBER OF ATOMS PER MOLECULE.
If the arbitrary material is a compound (entry A4, ICP is 1), enter the number of atoms of this element per molecule of arbitrary material.

If ICP is 0 (entry A4), enter the WEIGHT PERCENT OF THIS ELEMENT IN THE ARBITRARY MATERIAL. If the arbitrary material is a mixture, alloy, or conglomerate rather than a compound (entry A4, ICP is 0), enter the weight percent of this element in the arbitrary material. If ICP is 0, the sum of all the weight percents for the arbitrary material MUST be 100.0.

2. MX Mixture ID number. A mixture number is required on every standard composition specification. It defines the mixture that contains the material defined by the standard composition specifications data. The mixture numbers are utilized in the unit cell specification for INFHOMMEDIUM, LATTICECELL, or MULTIREGION problems and the KENO-VI geometry data.

S1. FD Fuel density. Enter ONLY for a solution. The standard composition component name for a solution starts with the four characters "SOLN." The available solution names are listed in Section M8.3. Enter the fuel density in units of grams of heavy metal (fissile material) per liter of solution.

S2. AML Acid molarity. Enter ONLY for a solution. If there is no acid in the solution, enter zero. If acid is present in the solution, enter the molarity of the solution.

O1. SPGR or ROTH Optional data. Does NOT apply to ARBITRARY MATERIALS. These optional data can be entered for a solution or for a basic standard composition using a keyword in the form SPG=SPGR, DEN=SPGR, DEN=ROTH, or SPG=ROTH. The keywords SPG= and DEN= are interchangeable. SPGR is the specific gravity of the fuel solution. If the standard composition name begins with SOLN and if the specific gravity of the solution is known, it should be entered. Otherwise the code has to interpolate in a table to obtain a specific gravity to use in the calculation. ROTH is the specified density of the standard composition in grams per cubic centimeter and can be entered if desired. The interaction between ROTH, the density of the standard composition, and VF (entry 3) is demonstrated in Section M7.5.5.

3. VF Volume fraction. The default value of the volume fraction is 1.0. It can be omitted if items 4, 5, 6a, and 6b are also omitted. The volume fraction can be interpreted as:

a. the volume fraction of this standard composition component in the mixture,

b. the density of the standard composition component in this application divided by the theoretical density listed in the Standard Composition Library (Section M8.2, Table M8.2.1), or a product of (a) and (b).

Section M7.5.5 discusses the interaction between ROTH and VF. For example, assume a homogenized mixture representing the water moderator and zircalloy cladding around a fuel pin is to be described. If the volume of the clad is 5.32 cc and if the volume of the water moderator is 44.68 cc, the mixture can be described using $H_2O$ with a volume fraction of 0.8936 [i.e., 44.68/(44.68+5.32)] and ZIRCALLOY with a volume fraction of 0.1064 [i.e., 5.32/ (44.68+5.32)].

For $UO_2$ fuel at 95% theoretical density, the volume fraction is 0.95. For the coolant water in an operating pressurized water reactor, the volume fraction might typically be 0.71.

To describe a mixed-oxide fuel pin with a density of 10.5 g/cc, that is 17.8 wt % $PuO_2$ and 82.2 wt % $UO_2$, two standard composition specifications must be entered: one for the $PuO_2$component and one for the $UO_2$ component. From the Standard Composition Library, Table M8.2.1, the theoretical density of the $PuO_2$ is 11.46 g/cc, and that of the $UO_2$ is 10.96 g/cc. Therefore, the volume fraction for the $PuO_2$ is 0.1631 (i.e., 0.178*10.50/11.46), and the volume fraction for the $UO_2$ is 0.7875 (i.e., 0.822*10.50/10.96).

Occasionally, a special mixture will be desired for which the determination of the component volume fractions will not be so straightforward. Consider a mixture of boral having 35 wt % $B_4C$, 65 wt % aluminum, and an overall density of 2.64 g/cc. Assume that neither boral nor $B_4C$ are in the Standard Composition Library. The volume fractions corresponding to boron, carbon, and aluminum may be computed as illustrated in Section M7.5.5.4, examples 3 and 4. Note that the calculation is

complicated by the fact that $B_4C$ is a compound and boron has two isotopes. It might be simpler to enter $B_4C$ and aluminum as arbitrary materials as illustrated in example 1 of Section M7.5.5.5. Examples 1 and 2 of Section M7.5.5.4 illustrate the use of $B_4C$ and aluminum to describe a boral mixture. **If the volume fraction is set to 0.0, the user can enter the number density for the nuclide (in atoms per barn-cm) for item 4. Number densities cannot be entered for standard compositions that represent multiple nuclides.**

4. ADEN Number density. Enter a value for the number density ONLY if the volume fraction (VF, entry 3) was entered as zero. The number density is entered in units of atoms per barn-cm.

5. TEMP Temperature. The default value of the temperature is 293 K. The temperature can be omitted if items 6a and 6b are also omitted.

   The temperature is used for Doppler broadening and/or the selection of the proper set of thermal scattering data.

   The resonance self-shielding calculation performed by NITAWL-II uses the Nordheim integral method and can account for Doppler broadening of the resonances at any specified temperature. If this material is known to contain a resonance nuclide, the user should enter a rough estimate of the temperature of the material. To determine whether resonance data are available for any of the nuclides in a material,

   a. check Table M8.2.1 if entry number 1 is not a solution, SOLN;

   b. check Table M8.3.1 if entry number 1 denotes a solution, SOLN.

   For each of the light nuclides (A ≤ 20 a.m.u.), the master cross-section library may contain one or more sets of thermal scattering data, each set corresponding to a different temperature. Scattering matrices for nuclides in media at elevated temperatures are generally fuller than those at lower temperatures. Tables M8.2.1 and M8.3.1 indicate the isotopes, elements, compounds, solutions, and alloys for which multiple sets of thermal scattering data are currently available. If multiple sets of thermal scattering data are available, the user should enter a rough estimate of the temperature of the material. The code will then interpolate between the appropriate data to produce a set of cross-section data that is most appropriate.

6a. IZA Isotope ZA number. Enter a value for each isotope in the standard composition component, item 1. Do not enter a value if the volume fraction, VF, is zero (entry 3).

   The ZA number of the isotope is entered if the user wishes to specify the isotopic distribution. This is done by entering 6a and 6b for each isotope until all the desired isotopes have been described. The "ZA" ID number is (A+1000*Z), where A is the atomic mass or weight of the isotope, and Z is the atomic number.

   Entries 6a and 6b can be skipped if the default values listed in the Isotope Distribution Table, Table M8.4.1, are acceptable.

6b. WTP Weight percent of the isotope. Do not enter a value if the volume fraction, VF, item 3 is zero. If entry 6a is entered, a value must also be entered for 6b. The weight percent of the isotope is the percent of this isotope in the element.

The user need not enter entries 6a and 6b for each isotope listed for the element in Table M8.4.1. The only requirement is that the sum of entries 6b for the element must sum to 100.0. Thus, uranium could be specified as 92235 3.2 92238 96.8 to represent 3.2% enriched uranium and ignore the 234 and 236 isotopes. However, 92235 3.0 and 92238 96.8 would not be correct because the 3.0 and 96.8 do not sum to 100.0. But 92235 3.0, 92234 0.2, and 92238 96.8 would be correct.

REPEAT THE SEQUENCE 6a and 6b until all the isotopes of the element have been defined as desired.

7. END The word END is entered to terminate the input data for a standard composition component. This END can have a label associated with it that can be as long as 12 characters. The label is optional and, if entered, is separated from the END by a single blank. At least two blanks must separate item 7 from the next data entry.

END COMP The words END COMP are entered only once for a problem. Their purpose is to signal the end of all the standard composition specification data that will be used in the problem. At least two blanks should separate entry 7 from the END COMP. Note that COMP is a label associated with this END. This label can be as long as 12 characters, but the first four characters must be COMP.

## C6.4.5 OPTIONAL UNIT CELL SPECIFICATION FOR INFINITE HOMOGENEOUS PROBLEMS

This section describes the unit cell data that can be entered for an INFHOMMEDIUM problem (data position 3 of Table C6.4.8). By default, the smallest mixture number is the mixture that will be used for the cell in XSDRNPM. In order to calculate the eigenvalue of the system, the mixture should contain fissionable material. Unit cell data are required for an infinite homogeneous medium problem ONLY if XSDRNPM is utilized by the control sequence and a mixture number other than the smallest mixture number is to be used in the cell. Additional information is available in Section M7.5.12 (see also Sections M7.5.4.1 and M7.5.6.1).

The data required to specify the unit cell for an INFHOMMEDIUM problem are given in Table C6.4.10. The individual entries are explained in the text following the table.

Table C6.4.10 Optional unit cell specification for INFHOMMEDIUM problems

| Entry number | Variable name | Type of data | Data entry | Comments |
|---|---|---|---|---|
| 1 | NAME | Keyword | CELLMIX | Initiates reading cell data for INFHOMMEDIUM |
| 2 | MFUEL | Mixture number in the cell | Mixture number | Specifies the mixture number to be used in the cell. Defaults to the smallest mixture number entered in the standard composition data |

NOTE: Unit cell data are necessary for INFHOMMEDIUM only if a mixture number other than the smallest mixture number is to be used in the cell and if a control sequence that executes XSDRN has been specified. If mixtures 1, 2, and 3 are specified and if mixture 3 is to be used in the cell, enter CELLMIX 3.

1. NAME Keyword. The keyword CELLMIX is entered to indicate that unit cell data will be entered. If this entry is entered, entry 2 must also be entered. These data can be entered ONLY if item 3, the "type of calculation" from Table C6.4.8, is entered as INFHOMMEDIUM.

2. MFUEL Mixture number. The mixture number that defines the mixture to be used in the cell. The default value is the smallest mixture number specified in the standard compositions specification data. If either entry 1 or 2 is entered, the other must also be entered.

To enter the above data, enter the keyword CELLMIX, one or more blanks, and the desired mixture number, MFUEL, as shown below:

CELLMIX MFUEL

To specify mixture 5 to be used in the cell, enter the following data:

CELLMIX 5

**WARNING: If unit cell data are to be entered as in example 3 below, the word CELLMIX must be spelled correctly. If it is misspelled, the code will not give an error message and will attempt to use the smallest mixture number. The input data will get out of phase for CSAS and SAS sequences that expect to read data after the unit cell specification. For example, CSAS26 expects to read KENO-VI data after the unit cell specification. If CELLMIX is misspelled, the code assumes that it is reading KENO-VI data and the data reading gets out of phase.**

## C6.4.6 UNIT CELL SPECIFICATION FOR LATTICECELL PROBLEMS

This section describes the unit cell data that must be entered for a LATTICECELL problem (data position 3 of Table C6.4.8). Additional information is available in Sections M7.5.4, M7.5.4.2, M7.5.6.2, and M7.5.8.2. The lattice cell description is especially suited to large arrays of identical cells. The code is limited to a single unit cell description for each problem. The unit cell data are used to provide the lump shape and dimensions for resonance cross-section processing, to provide lattice corrections for cross-section processing,

and to provide data used by XSDRNPM to create cell-weighted cross sections. Thus the unit cell specification plays a major role in providing problem-dependent cross sections. The configuration of the unit cell is determined by the specified "type of lattice." The unit cells are limited to infinitely long cylindrical rods in a square or triangular lattice, spheres in a cubic or triangular lattice, a symmetric array of slabs, or an asymmetric array of slabs. Two distinct types of cells ("Regular" and "Annular") are available to be used in a lattice cell problem. "Regular" cells allow a concentric spherical, cylindrical, or symmetric slab configuration constrained to a central fuel region surrounded by an optional gap, an optional clad, and an external moderator. The outer boundary of the cell is limited to provide an array having a square or triangular pitch. "Annular" cells allow a concentric spherical, cylindrical, or asymmetric slab configuration constrained to a central (second) moderator region surrounded by a fuel region having an optional gap and optional clad on each side of the fuel. The gap, if present, is identical on both sides of the fuel. The clad, if present, is identical in dimension and composition on both sides of the fuel. The outer region of the cell is filled with an external (first) moderator. The outer boundary of the cell is limited to provide an array having a square or triangular pitch.

Regular cells are: SQUAREPITCH, TRIANGPITCH, SPHSQUAREP, SPHTRIANGP, and SYMMSLABCELL.

Annular cells are: ASQUAREPITCH, ASQP, ATRIANGPITCH, ATRP, ASPHSQUAREP, ASSP, ASPHTRIANGP, ASTP, and ASYMSLABCELL.

The unit cell data are utilized by the code to define the geometric and resonance self-shielding corrections that will be applied to the cross sections. Cell-weighted cross sections are created only if XSDRNPM is invoked by executing a control sequence ending in X, (CSAS26X.). The cell-weighted cross sections have a flux disadvantage factor applied to them. Because this weighting is applied to each mixture used in the cell, these cell-weighted mixtures should not be used in the geometry description of a subsequent code. However, the cell-weighted mixture that is created is always assigned a mixture number of 500 and can be used in a subsequent code. KENO-VI is the subsequent code in CSAS26X. Note that cell-weighted cross sections are not generated if CSAS26 is used.

The unit cell geometry data required to specify a LATTICECELL problem are given in Table C6.4.11. The individual entries are explained in the text accompanying the table.

1. CTP Type of lattice. This defines the type of lattice or array configuration. Any one of the following alphanumeric descriptions may be used. Note that the alphanumeric description must be separated from subsequent data entries by one or more blanks. Figures C6.4.1 through C6.4.6 are used to illustrate the type of lattice configurations available in a LATTICECELL problem. The fuel region is labeled F; the gap, G; the clad, C; and the moderator, M. If two moderators are present, M1 is the first moderator, and M2 is the second moderator. Both "regular" and "annular" unit cells are available. The arrangement of the materials in the unit cell is strictly constrained. Regular cells are limited to a minimum of two regions and a maximum of four regions. The innermost regions must be fuel, which is surrounded by an optional gap and/or optional clad. The outer region is moderator.

   Annular cells are limited to a minimum of three regions. The innermost region must be the second moderator. The annular fuel region is outside the second moderator. If a clad is present, it must be identical in thickness and composition on both sides of the fuel. If a gap is present, it is the same thickness on both sides of the fuel. If the first and second moderators are the same material at the same density and temperature and do not contain resonance nuclides, the same mixture number can be used for both moderators. Otherwise, different mixture numbers must be used for the two moderators.

## Table C6.4.11 Unit cell specification for LATTICECELL problems

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CTP | Type of lattice | Always | | Describes the type of lattice or array configuration |
| | | | | SQUAREPITCH | Use for cylindrical rods in a square pitch |
| | | | | ASQUAREPITCH | Use for annular cylindrical rods in a square pitch |
| | | | | ASQP | Use for annular cylindrical rods in a square pitch |
| | | | | TRIANGPITCH | Use for cylindrical rods in a triangular pitch |
| | | | | ATRIANGPITCH | Use for annular cylindrical rods in a triangular pitch |
| | | | | ATRP | Use for annular cylindrical rods in a triangular pitch |
| | | | | SPHSQUAREP | Use for spherical pellets in a cubic lattice |
| | | | | ASPHSQUAREP | Use for annular spherical pellets in a cubic lattice |
| | | | | ASSP | Use for annular spherical pellets in a cubic lattice |
| | | | | SPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASPHTRIANGP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | ASTP | Use for spherical pellets in a bi-centered or face centered hexagonal close-packed lattice |
| | | | | SYMMSLABCELL | Use for a symmetric array of slabs |
| | | | | ASYMSLABCEL | Use for a periodic, but asymmetric array of slabs |
| 2 | PITCH | Array pitch (cm) | Always | Appropriate dimension | The center-to-center spacing (cm) between fuel lumps<br>For asymmetric slab cell, enter the distance from the center of one moderator to the center of the other moderator (cm) |
| 3 | FUELOD | Outside dimension of fuel (cm) | Always | Appropriate dimension | Outside diameter of fuel (cm), or the thickness of the fuel in a slab |
| 4 | MFUEL | Fuel mixture number | Always | Mixture number | Mixture number representing the fuel |
| 5 | MMOD | Moderator mixture number | Always | Mixture number | Mixture number representing the moderator |
| 6 | MMOD2 | 2nd moderator mixture number | ASYMSLABCELL annular cells | Mixture number | Mixture number representing the second moderator |
| 7 | TKMOD2 | 2nd moderator thickness<br>or<br>2nd moderator diameter (cm) | ASYMSLABCELL<br>or<br>annular cells | Thickness<br>or<br>diameter | Thickness of the second moderator (cm) for ASYMSLABCELL<br>or<br>diameter of inner moderator (cm) for other annular cells |
| 8 | CLADOD | Outside diameter of clad (cm) | If clad | Clad OD | OMIT IF NO CLAD. For a slab, CLADOD is the sum of thicknesses of the fuel, gap, and clad |
| 9 | MCLAD | Clad mixture number | If clad | Mixture number | OMIT IF NO CLAD. Mixture number representing the clad |
| 10 | CLADID | Inside diameter of clad (cm) | If gap | Clad ID | OMIT IF NO GAP between the fuel and clad |
| 11 | MGAP | Gap mixture number | If gap | Mixture number | OMIT IF NO GAP between the fuel and clad. A mixture number of zero is often used |
| 12 | END | Terminate LATTICECELL data | Always | END | Terminate the LATTICECELL input data by entering the word END. Do not start in column 1. At least two blanks must follow entry 12 |

Regular and annular cell configurations are specified as shown below.

**Regular Cells**

SQUAREPITCH is used for an array of cylinders arranged in a square lattice as shown in Fig. C6.4.1. The clad and/or gap can be omitted.

TRIANGPITCH is used for an array of cylinders arranged in a triangular pitch lattice as shown in Fig. C6.4.2. The clad and/or gap can be omitted.

SPHSQUAREP is used for an array of spheres arranged in a square pitch lattice. A cross-section view through a cell is represented by Fig. C6.4.1. The clad and/or gap can be omitted.

SPHTRIANGP is used for an array of spheres arranged in a triangular pitch (dodecahedral) lattice. A cross-section view through a cell is represented by Fig. C6.4.2. The clad and/or gap can be omitted.

SYMMSLABCELL is used for an infinite array of symmetric slab cells as shown in Fig. C6.4.3. The clad and/or gap can be omitted.

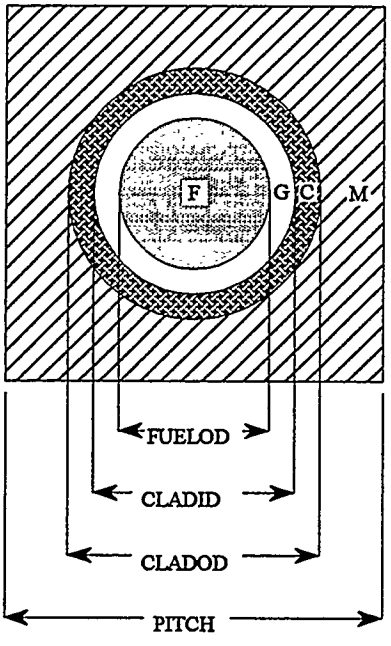



Figure C6.4.1 Arrangement of materials in a SQUAREPITCH and SPHSQUAREP unit cell

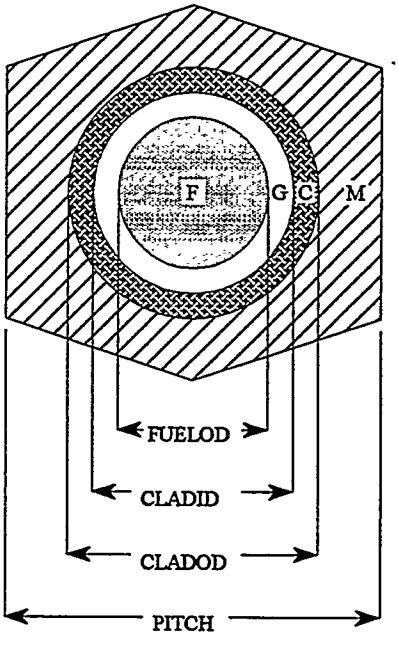



Figure C6.4.2 Arrangement of materials in a TRIANGPITCH and SHPTRIANGP unit cell

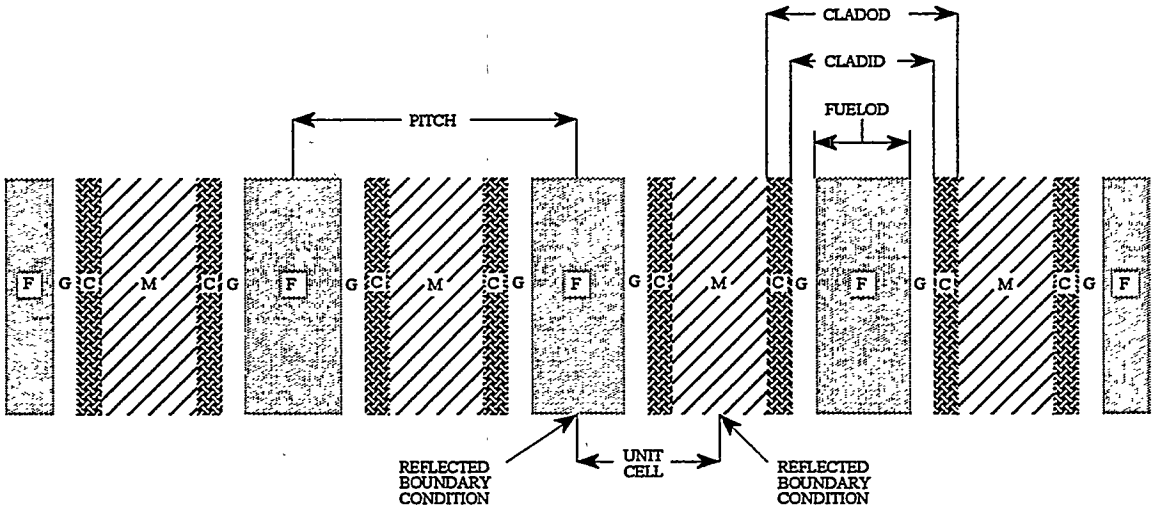



Figure C6.4.3 Arrangement of materials in a SYMMSLABCELL unit cell having reflected left and right boundary conditions

**Annular Cells**

ASQUAREPITCH or ASQP is used for annular cylindrical rods in a square pitch lattice as shown in Fig. C6.4.4. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

ATRIANGPITCH or ATRP is used for annular cylindrical rods in a triangular pitch lattice as shown in Fig. C6.4.5. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

ASPHSQUAREP or ASSP is used for spherical shells in a square pitch lattice as shown in Fig. C6.4.4. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

ASPHTRIANGP or ASTP is used for spherical shells in a triangular pitch (dodecahedral) lattice as shown in Fig. C6.4.5. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

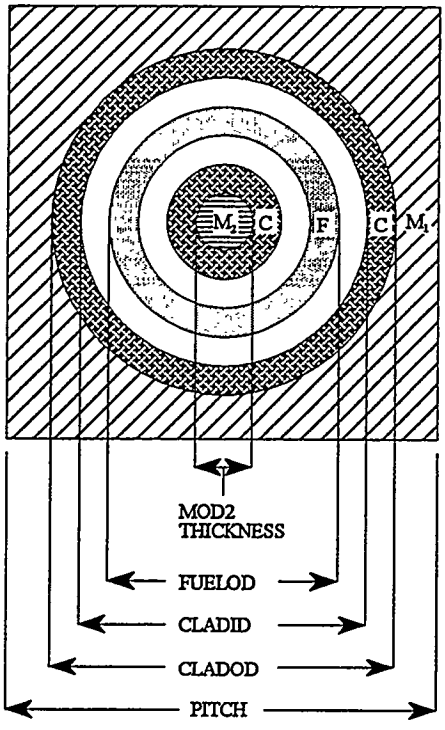



Figure C6.4.4 Arrangement of materials in an ASQUAREPITCH and ASPHSQUAREP unit cell

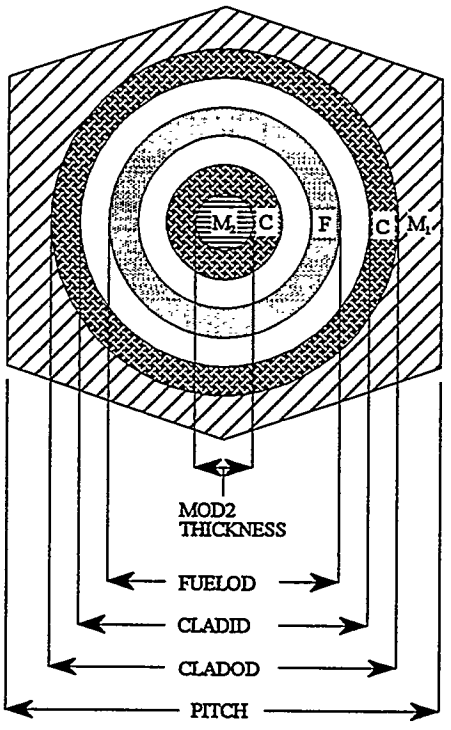



Figure C6.4.5 Arrangement of materials in an ATRIANGPITCH and ASPHTRIANGP unit cell

ASYMSLABCELL is used for a periodic, but asymmetric, array of slabs as shown in Fig. C6.4.6. The gap on both sides of the fuel must be identical, as must the clad. The clad and/or gap can be omitted.

2. PITCH Array pitch. This is the center-to-center spacing between the fuel lumps (rods, pellets, or slabs) in centimeters shown in Figs. C6.4.1 through C6.4.6. For an ASYMSLABCELL, enter the distance from the center of one moderator to the center of the other moderator. See Fig. C6.4.6.

3. FUELOD Outside dimension of fuel. This is the outside diameter of the fuel in centimeters as shown in Figs. C6.4.1 through C6.4.6. In slab geometry, enter the thickness of the fuel (see Figs. C6.4.1 through C6.4.6).

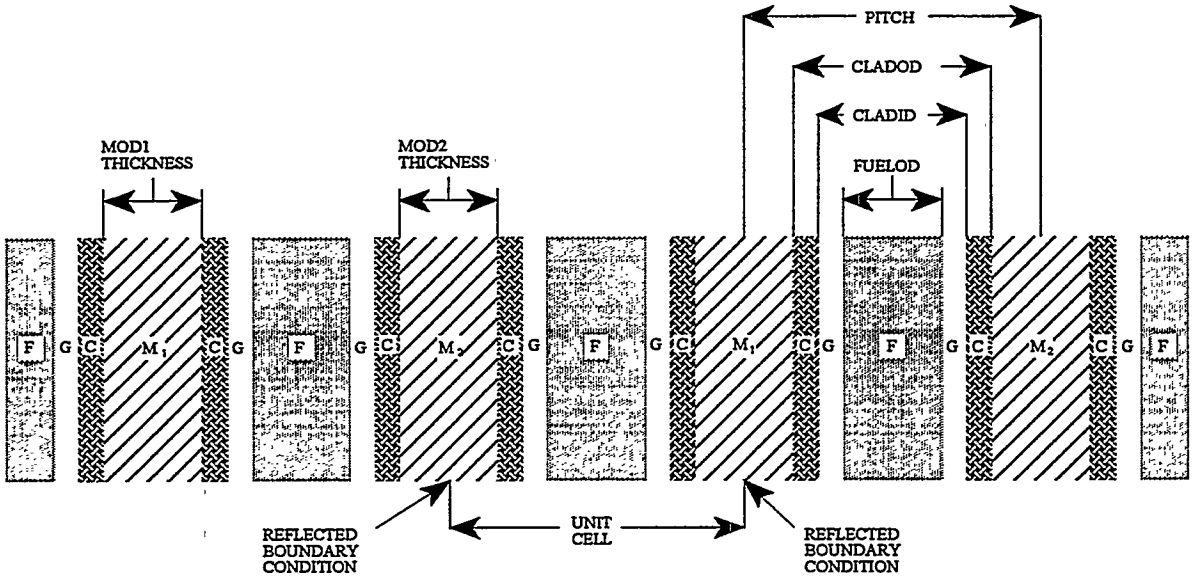



Figure C6.4.6 Arrangement of materials in an ASYMSLABCELL unit cell

4. MFUEL Fuel mixture number. This is the mixture number representing the fuel (F in Figs. C6.4.1 through C6.4.6).

5. MMOD Moderator mixture number. This is the mixture number representing the moderator (M or M1 in Figs. C6.4.1 through C6.4.6).

6. MMOD2 Second moderator mixture number. Enter ONLY for annular cells. This is the mixture number representing the second moderator (M2 in Figs. C6.4.4 through C6.4.6).

7. TKMOD2 Second moderator thickness. Enter ONLY for annular cells. This is the thickness or diameter of the second moderator in centimeters. See Figs. C6.4.4 through C6.4.6.

8. CLADOD Outside diameter of clad. Enter ONLY if a clad is present. Enter the thickness of the clad in centimeters. For a slab, CLADOD is the sum of the fuel thickness, twice the gap thickness, and twice the clad thickness. See Figs. C6.4.1 through C6.4.6.

9. MCLAD Clad mixture number. Enter ONLY if a clad is present. Enter the mixture number that represents the clad (C in Figs. C6.4.1 through C6.4.6).

10. CLADID Inside diameter of clad. Enter ONLY if a gap is present between the fuel and the clad. Omit if there is no gap. If a gap is present, enter the inside diameter of the clad in centimeters.

This corresponds to the outside diameter of the gap. In slab geometry, CLADID is the sum of the fuel thickness and twice the gap thickness. See Figs. C6.4.1 through C6.4.6.

11. MGAP — Gap mixture number. Enter ONLY if a gap is present between the fuel and the clad. Omit if there is no gap. Enter the mixture number representing the gap (G in Figs. C6.4.1 through C6.4.6). Zero is often used to represent a void.

12. END — The word END is entered to terminate the LATTICECELL data. An optional label can be associated with this END. The label can be as many as 12 characters long and is separated from the END by a single blank. At least two blanks must follow entry 12.

## C6.4.7 UNIT CELL SPECIFICATION FOR MULTIREGION PROBLEMS

These data are entered only if the problem is defined as a MULTIREGION problem. They describe the additional geometry data that are required for a MULTIREGION problem (data position 3 of Table C6.4.8). A MULTIREGION problem can be used to define a geometric arrangement that is more complicated than is allowed by a lattice cell. It can also be used for large geometric regions where the geometry effects for the cross sections are minimal. Additional information is available in Sections M7.5.4, M7.5.4.3, M7.5.6.3, and M7.5.8.3. BE FULLY AWARE THAT CROSS SECTIONS GENERATED USING MULTIREGION IGNORE THE LATTICE GEOMETRY EFFECTS IN DOING THE RESONANCE CORRECTIONS.

The additional data required for a MULTIREGION problem are given in Table C6.4.12; individual entries are explained in the text accompanying the table.

1. CS — Type of geometry. The type of geometry must always be specified for a MULTIREGION problem. The cross-section corrections do not account for lattice effects. The available geometry options are listed below.

   SLAB. This is used to describe a slab geometry.

   CYLINDRICAL. This is used to describe cylindrical geometry.

   SPHERICAL. This is used to describe spherical geometry.

   BUCKLEDSLAB. This is used for slab geometry with a buckling correction for the two transverse directions.

   BUCKLEDCYL. This is used for cylindrical geometry with a buckling correction in the axial direction.

2. BR — Right boundary condition. This is defaulted to vacuum. A value for BR **MUST** be entered for BUCKLEDSLAB and BUCKLEDCYL but is optional for SLAB, CYLINDRICAL, or SPHERICAL. The available options and their qualifications are listed below:

   VACUUM. This imposes a vacuum at the boundary of the system.

Table C6.4.12 Geometry specification for MULTIREGION problems

| Entry number | Variable name | Type of data | Entry requirement | Data entry | Comments |
|---|---|---|---|---|---|
| 1 | CS | Type of geometry | Always | | Describes the type of geometry. The options are listed below |
| | | | | SLAB | Use for slab geometry |
| | | | | CYLINDRICAL | Use for cylindrical geometry |
| | | | | SPHERICAL | Use for spherical geometry |
| | | | | BUCKLEDSLAB | Use for slab geometry with a buckling correction for the two transverse directions |
| | | | | BUCKLEDCYL | Use for cylindrical geometry with a buckling correction in the axial direction |
| 2 | BR | Right boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | VACUUM | Default is VACUUM. Describes the right/outside boundary condition |
| | | | | REFLECTED | This provides a non-return condition at the boundary |
| | | | | PERIODIC | Do not use for cylindrical or spherical |
| | | | | WHITE | Do not use for cylindrical or spherical<br>This provides isotopic return at the boundary |
| 3 | BL | Left boundary condition | Required for BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | VACUUM | Default is REFLECTED. Describes the left boundary condition |
| | | | | REFLECTED | This provides a non-return condition at the boundary |
| | | | | PERIODIC | Recommended for cylindrical or spherical |
| | | | | WHITE | Do not use for cylindrical or spherical<br>This provides isotropic return at the boundary |
| 4 | ORGN | Location of left boundary on the x-axis (cm) | BUCKLEDSLAB and BUCKLEDCYL; optional for other geometries | Appropriate dimension | Default is 0.0. Should not be changed for cylindrical or spherical geometry. A value must be entered if subsequent data are to be entered |
| 5 | DY | Buckling height (cm) | BUCKLEDSLAB and BUCKLEDCYL | Appropriate dimension | Omit for SLAB, CYLINDRICAL, and SPHERICAL.<br>This corresponds to one of the transverse dimensions of an actual 3-D slab assembly or to the length of a finite cylinder |
| 6 | DZ | Buckling depth (cm) | BUCKLEDSLAB | Appropriate dimension | Omit unless BUCKLEDSLAB was specified.<br>This is the buckling depth corresponding to the second transverse dimension of a 3-D slab assembly |
| 7 | END | End geometry parameters | Always | END | Enter the word END. Do not start in column 1 |
| 8 | MXZ | Mixture number in the zone | Always | Mixture number | Repeat entry numbers 8-10 until all zones are defined.<br>Enter the mixture number for this zone |
| 9 | RZ | Outside radius of the zone (cm) | Always | Appropriate dimension | Repeat entry numbers 8-10 until all zones are defined.<br>Enter the outside dimension of the zone (cm) |
| 10 | XMOD | External moderator index | Optional | | Repeat entry numbers 8-10 until all zones are defined.<br>Entry 10 is optional and can be omitted. If it is omitted, repeat entries 8 and 9 until all zones are defined |
| | | | | NOEXTERMOD | No moderating materials in the adjacent zones |
| | | | | ONEEXTERMOD | A moderating material is present in one adjacent zone |
| | | | | TWOEXTERMOD | Moderating materials are present in two adjacent zones |
| | END ZONE | Terminate zone data | | END ZONE | Enter when all zones have been defined by repeating entries 8 through 10 for each zone |

REFLECTED. This imposes mirror image reflection at the boundary. Do not use for CYLINDRICAL or SPHERICAL.

PERIODIC. This imposes periodic reflection at the boundary. Do not use for CYLINDRICAL or SPHERICAL.

WHITE. This imposes isotropic return at the boundary.

3. BL Left boundary condition. This is defaulted to reflected. A value for BL **MUST** be entered for BUCKLEDSLAB but is optional for SLAB, CYLINDRICAL, or SPHERICAL. The available options and their qualifications are listed below:

VACUUM. This imposes a vacuum at the boundary of the system.

REFLECTED. This imposes mirror image reflection at the boundary. For CYLINDRICAL or SPHERICAL, this is the only valid boundary condition because the left boundary corresponds to the centerline.

PERIODIC. This imposes periodic reflection at the boundary.

WHITE. This causes isotropic return at the boundary.

4. ORGN Location of left boundary on the origin. The default value of ORGN is 0.0. This is the only value allowed for CYLINDRICAL or SPHERICAL geometry. Enter the location of the left boundary on the axis perpendicular to the slab (in cm).

5. DY Buckling height. Enter *ONLY* for BUCKLEDSLAB or BUCKLEDCYL. This is the buckling height in centimeters. It corresponds to one of the transverse dimensions of an actual 3-D slab assembly or the length of a finite cylinder.

6. DZ Buckling depth. Enter *ONLY* for BUCKLEDSLAB. This is the buckling depth in centimeters. It corresponds to the second transverse dimension of an actual 3-D slab assembly.

7. END The word END is entered to terminate these data before entering the zone description data. It must not be entered in columns 1 through 3. At least two blanks are required between items 1 and 7 if they are the only entries. A label can be associated with this END. The label can be a maximum of 12 characters and is separated from the END by a single blank. At least two blanks must follow entry 7.

The zone description data are entered at this point. Entries 8, 9, and 10 are entered for each zone, and the sequence is repeated until all the desired zones have been described. To terminate the data, enter the words END ZONE.

8. MXZ Mixture number in the zone. Enter the mixture number of the material that is present in the zone. Enter a zero for a void. Repeat the sequence of entries 8 through 10 for each zone.

9. RZ Outside radius of the zone. Enter the outside dimension of the zone in centimeters. In slab geometry, RZ is the location of the zone's right boundary on the x-axis. Repeat the sequence of entries 8 through 10 for each zone.

10. XMOD Optional external moderator index. This optional entry describes the conditions existing in the zones adjacent to the present zone. Appropriate conditions are provided by default. The available options are listed below.

NOEXTERMOD. This means there are no moderating materials between this zone and either boundary.

ONEEXTERMOD. This means one of the zones between this zone and one of the boundaries contains a material.

TWOEXTERMOD. This indicates that at least one zone on either side of this zone contains a material.

CAUTION: User-provided entries do not always supersede the default values: (1) For SLAB geometry, user-supplied values will be used everywhere except zone 1. If the left boundary is reflected, the default value of ONEEXTERMOD will be used unless NOEXTERMOD is specified. (2) For SPHERICAL and CYLINDRICAL geometry, default values will be used for all zones unless NOEXTERMOD is specified. NOEXTERMOD allows the user to "turn off" geometric corrections.

**Repeat the sequence of entries 8 and 9 or 8 through 10 for each zone. When all the zones have been described, enter the words END ZONE.**

END ZONE This is used to terminate the MULTIREGION zone data. Enter the words END ZONE when all the zones have been described. Note that ZONE is a label associated with this END. This label can be as long as 12 characters, but the first four characters **must** be ZONE. At least two blanks must follow this entry.

## C6.4.8 OPTIONAL PARAMETER DATA

Most of the parameter data for Criticality Safety Analysis Sequence No. 6 are determined by the control module and cannot be adjusted. However, certain parameters can be altered. If the default parameters are acceptable, this section of input data should be omitted in its entirety. Nondefault values for one or more of the parameters can be specified by entering the words MORE DATA followed by the desired keyworded parameters and their associated values. One or more of the parameters can be entered in any order. Default values are used for parameters that are not entered. Each parameter is entered by spelling its name, followed immediately by an equal sign and the value to be entered. There should not be a blank between the parameter name and the equal sign. Each parameter specification must be separated from the rest by at least one blank. For example

MORE DATA ISN=16 EPS=0.00008 END

would result in using an S16 angular quadrature set and tightening the convergence criteria to 0.00008 in the XSDRNPM calculation. Note that the END is not the end for the analytical sequence, entry 5 of Table C6.4.7.

Table C6.4.13 describes the optional parameters that can be entered as data and the code or codes the parameters apply to. A description of each entry is contained in the text accompanying the table.

Table C6.4.13 Optional parameter data

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 1 | MORE DATA | Input flag | | This signals that optional parameter data will be entered. Enter only those parameters you wish to change |
| 2 | ISN= | Order of angular quadrature | XSDRNPM | The default value is 8. This allows using another value |
| 3 | SZF= | Spatial mesh size factor | XSDRNPM | The default value is 1.0.<br>0<SZF<1.0 gives a finer mesh.<br>SZF>1.0 gives a coarser mesh |
| 4 | IIM= | Max. number of inner iterations | XSDRNPM | The default value is 20. This allows using another value |
| 5 | ICM= | Max. number of outer iterations | XSDRNPM | The default value is 25. This allows using another value |
| 6 | EPS= | Overall convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 7 | PTC= | Point convergence criteria | XSDRNPM | The default value is 0.0001. This allows using another value |
| 8 | BKL= | Buckling factor | XSDRNPM | The default value is 1.420892.<br>Use ONLY for a multiregion problem that specifies BUCKLEDSLAB or BUCKLEDCYL |
| 9 | IUS= | Upscatter scaling factor | XSDRNPM | The default value is zero.<br>IUS=0 doesn't utilize upscatter scaling. IUS=1 uses upscatter scaling to accelerate the solution and/or speed convergence |
| 10 | RES= | Resonance data | BONAMI NITAWL-II | Enter the mixture number, geometry type (SLAB, CYLINDER, SPHERE), and the thickness of the slab or radius of the sphere or cylinder, in cm. Optionally enter the inner radius (cm) to specify an annular cylinder or sphere. |
| 11 | DAN(mm)= | Dancoff factor for the specified mixture | BONAMI NITAWL-II | Enter the mixture number, mm, to which the Dancoff factor applies inside the parentheses; enter the Dancoff factor after the equal sign<br><br>Repeat entries 10 and 11 for all resonance mixtures used in the problem that are not treated in the LATTICECELL or MULTIREGION description |

Table C6.4.13 (continued)

| Entry number | Keyword name | Type of data | Applicable module | Comments |
|---|---|---|---|---|
| 12 | BAL= | Key to print balance tables | XSDRNPM | The default value is FINE. BAL=NONE suppresses printing the balance table. BAL=ALL prints all balance tables. BAL=FINE prints only the fine-group balance tables |
| 13 | DY= | First transverse dimension | XSDRNPM | The first transverse dimension, in centimeters, used in a buckling correction to calculate leakage normal to the principal calculation direction (i.e., the height of a slab or a cylinder) |
| 14 | DZ= | Second transverse dimension | XSDRNPM | The second transverse dimension, in centimeters, used for a buckling correction (i.e., the width of a slab) |
| 15 | COF= | Diffusion coefficient option for transverse leakage correction | XSDRNPM | The default is 0. See Section F3.5, 3$ array, variable IPN |
| 16 | FRD= | Unit from which fluxes will be read | XSDRNPM | Enter the unit number from which the flux guess for XSDRNPM will be read |
| 17 | FWR= | Unit on which fluxes will be written | XSDRNPM | Enter the unit number where the binary fluxes from XSDRNPM will be written |
| 18 | DAB= | Number of direct access data blocks | MIP[a] | The default is 200. Number of blocks allocated for direct-access unit 90 |
| 19 | AXS= | Unit on which a mixed ANISN library will be written | ICE | Enter the unit number where ICE is to write a mixed ANISN library |
| 20 | COLL | Key to activate collapse of thermal groups | MIP[a] | Enter COLL to collapse all thermal groups into one group for the shielding sequences |
| 21 | END | Terminus | | Terminate the optional parameter data. Do not start in column 1. At least two blanks must follow entry number 21 |

[a]MIP is the Material Information Processor.

1. MORE DATA...These words, followed by one or more blanks, are entered ONLY if optional parameter data are to be entered. They must precede all other optional parameter data. Entries 2 through 19 can be entered in any order.

2. ISN Order of angular quadrature for XSDRNPM. Quadrature sets are geometry-dependent quantities that are defaulted to order 8 by the Material Information Processor. See Sections M7.2.5.7 and M7.5.7 for a more detailed explanation.

3. SZF Spatial mesh size factor for XSDRNPM. The size of the largest mesh interval can be adjusted by entering a value for SZF. The default value is 1.0. A value between zero and 1.0 yields a finer mesh; a value greater than 1.0 yields a coarser mesh. See Sections M7.2.5.6 and M7.5.7 for more details.

4. IIM Maximum number of inner iterations for XSDRNPM. This is the maximum number of inner iterations to be used in the XSDRNPM calculation. The default value is 20. See Section M7.5.7 for a more detailed explanation.

5. ICM Maximum number of outer iterations for XSDRNPM. This is the maximum number of outer iterations to be used in the XSDRNPM calculation. The default value is 25. If the calculation reaches the outer iteration limit, a larger value should be used. See Section M7.5.7 for a more detailed explanation.

6. EPS Overall convergence criteria for XSDRNPM. This is used by XSDRNPM after each outer iteration to determine if the problem has converged. The default value of EPS is 0.0001. A value less than 0.0001 tightens the convergence criteria; a larger value loosens the convergence criteria. See Sections M7.2.5.8 and M7.5.7 for additional information.

7. PTC Pointwise convergence criteria for XSDRNPM. This is the point flux convergence criteria used by XSDRNPM to determine if convergence has been achieved after an inner iteration. The default value is 0.0001. A smaller value tightens convergence; a larger value loosens it. See Sections M7.2.5.8 and M7.5.7 for more details.

8. BKL Buckling factor for XSDRNPM. A buckling factor should be used ONLY for a MULTIREGION BUCKLEDSLAB or BUCKLEDCYL problem. Because cylinders are assumed to be infinitely long and slabs are assumed to be infinite in both transverse directions, the analytic sequence may tend to overestimate the total flux for a finite system. A buckling correction can be used to approximate the leakage from the system in the transverse direction(s). The extrapolation distance factor, BKL, is defaulted to 1.420892. See Sections M7.2.5.4 and M7.5.7 for additional details.

9. IUS Upscatter scaling flag for XSDRNPM. This option allows the use of upscatter scaling to accelerate the solution or force convergence. The default value is zero, in which case upscatter scaling is not used. IUS=1 facilitates upscatter scaling. Guidelines are not available to indicate when upscatter scaling is needed. Some problems will not converge with it, and some will not converge without it. See Section M7.5.7 for a more detailed explanation.

10. RES Resonance data for BONAMI and NITAWL-II. This parameter allows overriding the default resonance data for **materials that are not used in the unit cell**. The default resonance data are infinite homogeneous media that may not accurately represent the geometry (for example, a thin clad or container region). The resonance data are entered in the form:

RES= mixture number geometry type thickness [inner radius]

The mixture number is the material for which the resonance data apply. The geometry type can be a sphere, slab, or cylinder and defines the third entry of the 3* array in NITAWL-II. The thickness is the thickness of a slab or the radius of a sphere or cylinder in centimeters and defines the fourth

entry of the 3* array in NITAWL-II. The inner radius is the optionally entered inner radius of an annular cylinder or sphere and defines the sixth entry of the 3* array in NITAWL-II. This entire data sequence should be entered for each material for which the default data are inadequate. See Section M7.5.7 for additional details.

**WARNING: If this parameter is used to enter resonance data for a mixture that is used in the unit cell specification, it will NOT be used. Information provided in the unit cell data will be used instead.**

11. DAN(mm)= Dancoff factor for the specified mixtures. This allows entering a Dancoff factor to be used in the resonance correction for the specified mixture. The Dancoff data are entered in the form:

    DAN(mm) = Dancoff factor.

    Note that the parentheses must be entered as part of the data. The mixture number, mm, to which the Dancoff factor applies, must be enclosed in the parentheses. The Dancoff factor to be used is entered after the equal sign. See Section M7.5.7 for additional details.

    Items 10 and 11 should be entered for each resonance mixture that is not specified in the LATTICECELL or MULTIREGION unit cell specification data. See Sections M7.5.10 and C6.5.1 for additional information.

12. BAL Balance table print flag for XSDRNPM. This allows control of the balance table print from XSDRNPM. The default value is FINE. BAL=NONE suppresses the balance table print. BAL=ALL prints all of the balance tables. BAL=FINE prints only the fine-group balance tables. See Section M7.5.7 for additional details.

13. DY= First transverse dimension for XSDRNPM. This is the first transverse dimension, in centimeters, used in a buckling correction to calculate the leakage normal to the principal calculation direction (the height of a slab or cylinder). It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a cylinder or slab system of finite height for a LATTICECELL problem. DY= is defaulted to an infinite height, or is set to DY for a buckled MULTIREGION cell description. See Sects M7.2.5.4 and M7.5.7 for additional details.

14. DZ= Second transverse dimension for XSDRNPM. This is the second transverse dimension, in centimeters, used for a buckling correction for a slab of finite width. It should only be entered if XSDRNPM is to create cell-weighted cross sections and/or calculate the eigenvalue of a LATTICECELL slab of finite width. DZ= is defaulted to an infinite width, or is set to DZ for a buckled MULTIREGION slab cell of finite width. See Sections M7.2.5.4 and M7.5.7 for additional details.

15. COF= Diffusion coefficient for transverse leakage corrections in XSDRNPM. The default value is 0. The available options are

    COF=0 sets a transport-corrected cross section for each zone.

    COF=1 use a spatially averaged diffusion coefficient for each zone.

COF=2 use a diffusion coefficient for all zones that is one-third the diffusion coefficient determined from the spatially averaged transport cross section for all zones.

COF=3 use a flux and volume weighting across all zones.

See Section M7.5.7 or Section F3.5, 3$ array, variable IPN for more details.

16. FRD= Unit from which fluxes are read for XSDRNPM. If fluxes are to be used for the initial flux guess, enter the unit number from which they are to be read. The user is responsible for supplying job control language data that is necessary for reading the flux dataset. See Section M7.5.7 for more details.

17. FWR= Unit on which fluxes will be written for XSDRNPM. If binary fluxes are to be saved from this XSDRNPM calculation, enter the unit number where they will be written. The user is responsible for supplying job control language data that is necessary for creating and saving the flux dataset. See Section M7.5.7 for more details.

18. DAB= Number of direct-access data blocks for the Material Information Processor. This is the number of blocks allocated for direct access on unit 90. The default is 200. See Section M7.5.7 for additional details. If the job fails in the Material Information Processor with an LMP005 DA ERROR, then DAB must be used to enter a larger value. See Section C6.7.4 for additional details.

19. AXS= Logical unit number for ICE. Write a mixed ANISN library on this unit. The default is zero (no library). If the unit number is input as 7, the ANISN library will be in free-form card image. For any other unit number, the library will be binary. The user is responsible for supplying job control language data that is necessary for creating and saving the ANISN cross-section library. An ANISN cross section can only be created by the control sequences CSASI, CSASIX, and SAS3. See Section M7.5.7 for additional details.

20. COLL Activate collapse of thermal groups. This parameter should only be used with the shielding analysis sequences (SAS1, etc.). It causes the Material Information Processor to instruct XSDRN to collapse all thermal groups into one group.

21. END The word END is entered to terminate the optional parameter data. A label can be associated with this END. The label can be as long as 12 characters, but must be preceded by a single blank. If this END is entered without a label, it must not begin in column 1. At least two blanks must follow this entry.

### C6.4.9 SUMMARY OF KENO-VI DATA

KENO-VI data are required for CSAS26 and CSAS26X. A brief outline of the KENO-VI data is given in Table C6.4.14. The input data are summarized in Table C6.4.15. The input data for KENO-VI are discussed in detail in Sects. F17.4 and F17.5. Data need not be supplied for entries whose default value is acceptable. KENO-VI mixing table data cannot be entered or used in CSAS26 because they are automatically supplied by the control module.

Table C6.4.14 Outline of KENO-VI data

| Type of data | Starting flag | Comments | Termination flag |
|---|---|---|---|
| Title | None | 80 columns, must be entered first | None |
| Parameters | READ PARM | Enter desired parameter data | END PARM |
| Biasing or weighting | READ BIAS | Enter desired biasing data | END BIAS |
| Geometry | READ GEOM | Enter desired geometry data | END GEOM |
| Array data or unit orientation | READ ARRAY | Enter desired array data | END ARRAY |
| Boundary conditions or albedos | READ BNDS | Enter desired albedo data | END BNDS |
| Start data or initial source | READ START | Enter desired start data | END START |
| Mixing table data | READ MIXT | The CSAS6 modules allow only SCT= and EPS= | END MIXT |
| Plot data | READ PLOT | Enter desired plot data | END PLOT |
| KENO-VI data terminus | END DATA | Enter to signal the end of all KENO -VI data | |

## Table C6.4.15 Summary of KENO-VI data

### Summary of parameter data

TITLE: The title must be entered first (80 columns) See Sect. F17.4.3

PARAMETERS: Format: READ PARAM enter parameter data here END PARAM
If parameters are entered, they must follow the title. See Sects. F17.4.3, F17.5.2, and F17.5.3.

| KEY | STD. | DEFINITION |
|---|---|---|
| RND= | given | Random number |
| TME= | 30 min | Execution time (min) |
| TBA= | 0.5 min | Batch time (min) |
| WTA= | 0.5 | Average weight |
| WTH= | 3.0 | Wt. for splitting |
| WTL= | 1/WTH | Russian Roulette wt. |
| GEN= | 103 | No. of generations |
| NPG= | 300 | No. per generation |
| NSK= | 3 | Generations skipped |
| RES= | 0 | Gens. between restart |
| NBK= | NPG+25 | Neutron bank positions |
| XNB= | 0 | Extra bank entries |
| NFB= | NPG | Fission bank positions |
| XFB= | 0 | Extra bank entries |
| XID= | 0 | No. of extra 1-D's |
| LNG= | 1000000 | Words of storage* |
| BEG= | 1 | Restart at this gen. |
| NB8= | 200 | Blocks for d.a. unit |
| NL8= | 512 | Length of d.a. block |
| SIG= | given | Standard deviation for problem termination |

| KEY | STD. | DEFINITION |
|---|---|---|
| RUN= | YES | Execute problem |
| FLX= | NO | Fluxes |
| FDN= | NO | Fission densities |
| ADJ= | NO | Adjoint calculation |
| AMX= | NO | All mixture xsecs |
| XAP= | NO | Xsec angles and probs. |
| XS1= | NO | 1-D xsecs |
| XS2= | NO | 2-D xsecs |
| PKI= | NO | Fission spectrum |
| P1D= | NO | Extra 1-D xsecs |
| FAR= | NO | Fiss. and abs. |
| MKP= | NO | Matrix by location |
| CKP= | NO | Cofactor k by loc. |
| FMP= | NO | Fiss. prod. by loc. |
| MKU= | NO | Matrix by unit |
| CKU= | NO | Cofactor k by unit |
| FMU= | NO | Fiss. prod. by unit |
| GAS= | NO | Fiss. & Abs. by Reg. |

| KEY | STD. | DEFINITION |
|---|---|---|
| MKH= | NO | Matrix by hole |
| CKH= | NO | Cofactor k by hole |
| FMH= | NO | Fiss. prod. by hole |
| HHL= | NO | MKH at highest level |
| MKA= | NO | Matrix by array |
| CKA= | NO | Cofactor k by array |
| FMA= | NO | Fiss. prod. by array |
| HAL= | NO | MKA at highest level |
| PLT= | YES | Printer plots |
| BUG= | NO | Debug print |
| TRK= | NO | Print neutron tracks |
| PWT= | NO | Print avg. weight |
| PGM= | NO | Unprocessed geometry |
| SMU= | NO | Self-multiplication |
| NUB= | NO | Neutrons per fission |
| PAX= | NO | Albedo-xsec array |

| KEY | STD. | DEFINITION |
|---|---|---|
| XSC= | 14 | Mixed xsecs |
| ALB= | 79 | Albedo |
| WTS= | 80 | Weights |
| LTB= | 4 | Working xsecs from NITAWL or XSDRN |
| SKT= | 16 | Scratch |
| RST= | 95 | Read restart |
| WRS= | 95 | Write restart |
| XSC= | 14 | Mixed xsecs |

*The words of storage in CSAS6 are defaulted to 100,000 and can be overridden by using the size= parameter on the analytical sequence specification which then becomes the default value of LNG in KENO-VI.
For example =CSAS26 PARM=SIZE=200000
=CSAS26 PARM=(CHECK,SIZE=200000)

Table C6.4.15 (continued)

## Summary of array data

ARRAY Format: READ ARRAY array parameters data type orientation data END ARRAY See Sects. F17.5.5, F17.5.6, and F17.5.7.

Repeat the sequence ARRAY PARAMETERS DATA TYPE ORIENTATION DATA for each array used in the problem.

| ARRAY PARAMETERS | | | DATA TYPE |
|---|---|---|---|
| KEYWORD | DEFAULT | DEFINITION | FILL<br>LOOP |
| ARA= | none | No. defining the array | |
| TYP= | cuboidal | Array Type (cuboidal or triangular) | |
| NUX= | none | No. of units in X direction | |
| NUY= | none | No. of units in Y direction | |
| NUZ= | none | No. of units in Z direction | |
| GBL= | none | Global array number** | |
| COM= | None | Delim comment delim Optional comment is a maximum of 132 characters | |

**If specified, it need be entered only once per problem.

ORIENTATION DATA FOR FILL

Enter unit numbers to define every position in the array. When entering data utilizing the options in this table, the count field and option field must be adjacent with no imbedded blanks. The operand field may be separated from the option field by one or more blanks. Orientation data for FILL is terminated by entering END FILL.

| COUNT FIELD | OPTION FIELD | OPERAND FIELD | COMMENTS |
|---|---|---|---|
| | | j | Stores j at the current position in the array |
| i | R | j | Stores j in the next i positions in the array |
| i | * | j | Stores j in the next i positions in the array |
| i | $ | j | Stores j in the next i positions in the array |
| | F | j | Fills remainder of the array with unit no. j starting with the current array position |
| | A | j | Sets the current position in the array to j |
| i | S | | Increments the current position in the array by i allows skipping i positions. The value of i may be positive or negative |
| i | Q | j | Repeats the previous j entries i times. The default value of i is 1 |
| i | N | j | Repeats the previous j entries i times, inverting the sequence each time. The default value of i is 1 |
| i | B | j | Starting with the entry at -i from the current position, store entries in inverse order until position -(i+j) is reached. Default value of i=1 |
| i | P | j | Alternately stores j and -j in the next i positions of the array |
| i | I | j k | Provides the end points, j and k, with i entries linearly interpolated between them (i.e., a total of i+2 points). At least one blank must separate j and k. When used for an integer array, the I option should only be used to generate integer steps [i.e., (k-j)/(i+1) should be a whole number]. |
| | T | | Terminates the data reading for the array |

ORIENTATION DATA FOR LOOP

Enter the unit number and nine numbers that define the position(s) of that unit. Data for each of these ten entries are repeated until every position in the array has been defined. Orientation data for LOOP is terminated by entering END LOOP.

ENTER DATA IN THE FORM:

| DATA ENTRY | COMMENTS |
|---|---|
| LTYPE | The unit or box type. LTYPE must be greater than 1 |
| IX1 | Starting position in the X direction. IX1 must be at least 1 and no larger than the value entered for NUX |
| IX2 | Ending position in the X direction. IX2 must be at least 1 and no larger than the value of NUX |
| INCX | The number of units by which increments are made in the X direction |
| IY1 | The starting position in the Y direction. IY1 must be at least 1 and less than the value entered for NUY |
| IY2 | Ending position int he Y direction. IY2 must be at least 1 and no larger than the value of NUY |
| INCY | The number of units by which increments are made in the positive Y direction |
| IZ1 | Starting position in the Z direction. IZ1 must be at least 1 and no larger than NUZ |
| IZ2 | Ending position in the Z direction. IZ2 must be at least 1 and no larger than NUZ |
| INCZ | The number of units by which increments are made in the positive Z direction |

Table C6.4.15 (continued)

## Summary of biasing data

BIAS (weighting)

Format: READ BIAS keyword correlation data auxiliary END BIAS
See Sects. F17.4.7 and F17.5.8

| KEYWORD | DESCRIPTION | |
|---|---|---|
| ID= | CORRELATION DATA will be read next. | |
| | id | Material ID. Enter from table at right to use weighting data from the library |
| | ibgn | Beginning bias ID |
| | iend | Ending bias ID |
| WT= | AUXILIARY DATA will be read next. | |
| WTS= | AUXILIARY DATA will be read next. | |
| | wttitl | Material title (12-character maximum) |
| | id | Material ID |
| | nsets | Number of sets of group structures |
| | REPEAT | (thkinc, numinc, ngpwt, wtavg) nsets times |
| | thkinc | Thickness per increment |
| | numinc | Number of increments |
| | ngpwt | Number of energy groups for this set of wts |
| | wtavg | Enter numinc x ngpwt values of wtavg |

| MATERIAL | ID | ENERGY GROUPS | THICKNESS/ INCREMENT |
|---|---|---|---|
| Concrete | 301 | 16,27,44,218,238 | 5 cm |
| Paraffin | 400 | 16,27,44,218,238 | 3 cm |
| Water | 500 | 16,27,44,218,238 | 3 cm |
| Graphite | 6100 | 16,27,44,218,238 | 20 cm |

For CORRELATION DATA, the material ID is chosen from material ID column above (the keyword is ID=).
For AUXILIARY DATA, the material ID is chosen by the user and the keyword is WT= or WTS=.
When AUXILIARY DATA are entered, CORRELATION DATA must also be entered to use the data.
Beginning and ending bias ID's are defined by the user. The geometry specification that has the bias ID equal to the beginning bias ID utilizes the wt avg's from the first interval of material ID.

Table C6.4.15 (continued)

## Summary of boundary condition data

BNDS (albedo or boundary conditions)

Format: READ BNDS face code albedo name END BNDS
See Sect. F17.4.7

The sequence FACE CODE ALBEDO NAME is entered as many times as necessary to define the appropriate albedo boundary conditions. The default for all faces is vacuum.

FACE CODES FOR ENTERING BOUNDARY (ALBEDO) CONDITIONS

| FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION | FACE CODE | DEFINITION |
|---|---|---|---|---|---|---|---|
| +XB= | Positive X face | XFC= | Both X faces | +YX= | Positive X and Y faces | &ZY= | Positive Y and Z faces |
| &XB= | Positive X face | YFC= | Both Y faces | &XY= | Positive X and Y faces | -XY= | Negative X and Y faces |
| -XB= | Negative X face | ZFC= | Both Z faces | &YX= | Positive X and Y faces | =XZ= | Negative X and Z faces |
| +YB= | Positive Y face | +FC= | All positive faces | +XZ= | Positive X and Z faces | =YZ= | Negative Y and Z faces |
| &YB= | Positive Y face | &FC= | All positive faces | +ZX= | Positive X and Z faces | YXF= | All X and Y faces |
| -YB= | Negative Y face | -FC= | All negative faces | &XZ= | Positive X and Z faces | ZXF= | All X and Z faces |
| +ZB= | Positive Z face | XYF= | All X and Y faces | &ZX= | Positive X and Z faces | ZYF= | All Y and Z faces |
| &ZB= | Positive Z face | XZF= | All X and Z faces | +YZ= | Positive Y and Z faces | -YX= | Negative X and Y faces |
| -ZB= | Negative Z face | YZF= | All Y and Z faces | +ZY= | Positive Y and Z faces | -ZX= | Negative X and Z faces |
| ALL= | All 6 faces | +XY= | Positive X and Y faces | &YZ= | Positive Y and Z faces | -ZY= | Negative Y and Z faces |

Any albedo condition on any face is allowed if the global unit boundary record consists of only one geometry record that is a cuboid.
Any albedo condition, but the same condition on all faces, is allowed if the global unit boundary record consists of one geometry record that is composed of paired planes.
Only void, mirror, or white boundary conditions, the same condition on all faces, if the global unit boundary record consists of only one geometry record of any shape.
Only void boundary condition is allowed if the global unit boundary record contains multiple geometry records. Default is void on all global unit boundary surfaces.

ALBEDO NAMES AVAILABLE ON THE KENO V ALBEDO LIBRARY, FOR USE WITH THE FACE CODES

| ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION | ALBEDO NAME | DESCRIPTION |
|---|---|---|---|---|---|
| DPOH20<br>DPOH20<br>DPO<br>DPO | 12-in. double PO water differential albedo with 4 incident angles | CONC-4<br>CON4<br>CONC4 | 4-in. concrete differential albedo with 4 incident angles | VACUUM<br>VOID<br>VACU<br>VAC | Vacuum condition |
| H20<br>WATER | 12-in. water differential albedo with 4 incident angles | CONC-8<br>CON8<br>CONC8 | 8-in. concrete differential albedo with 4 incident angles | SPECULAR<br>MIRROR<br>MIRR | Mirror image reflection |
| PARAFFIN<br>PARA<br>WAX | 12-in. paraffin differential albedo with 4 incident angles | CONC-12<br>CON12<br>CONC12 | 12-in. concrete differential albedo with 4 incident angles | SPEC<br>SPE<br>MIR | |
| CARBON<br>GRAPHITE<br>C | 200-cm carbon differential albedo with 4 incident angles | CONC-16<br>CON16<br>CONC16 | 16-in. concrete differential albedo with 4 incident angles | PERIODIC<br>PERI<br>PER | Periodic boundary condition |
| ETHYLENE<br>POLY<br>CH2 | 12-in. polyethylene differential albedo with 4 incident angles | CONC-24<br>CON24<br>CONC24 | 24-in. concrete differential albedo with 4 incident angles | WHITE | White boundary condition |

Table C6.4.15 (continued)

## Summary of geometry data

GEOMETRY (region)

Format: READ GEOM enter geometry region data here END GEOM
See Sects. F17.4.4, F17.5.1.2, F17.5.6, and F17.5.7.

GEOMETRY REGION DATA consists of SIMPLE GEOMETRY REGION DATA and MODIFICATION GEOMETRY REGION DATA.

ENTER GEOMETRY REGION DATA IN THE FOLLOWING FORM:
OPTIONAL GLOBAL SPECIFICATION
UNIT n
OPTIONAL GEOMETRY COMMENT
GEOMETRY, MODIFICATION, MEDIA, and BOUNDARY DATA
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
SIMPLE GEOMETRY REGION DATA IN THE FOLLOWING FORM:

GLOBAL Enter only to specify this unit as the global unit.
UNIT n
COM=delim comment delim This optional comment can be up to 132 characters. It must begin and end with a delimiter.
Geometry Keyword Input Parameters optional origin data (ORIGIN) optional chord data (CHORD) optional rotation data (ROTATE degrees)
Media Keyword Input Parameters
Boundary Keyword Input Parameters Each unit must have a boundary record.
Enter as many geometry description specifications as necessary to describe the unit and as many units as necessary to describe the system.

GEOMETRY REGION INPUT DATA REQUIREMENTS

| TYPE OF DATA | KEYWORD | INPUT PARAMETERS | NOTES |
|---|---|---|---|
| GEOMETRY | CONE | LblRt Zt Rb Zb | Lbl is an integer assigned to a geometry record, unique within the unit, that relates the geometry record to the media region in the Region Definition Vector i1 i2 ... |
| | CUBOID | Lbl+X -X +Y -Y +Z -Z | |
| | CYLINDER | Lbl R Zt Zb | |
| | DODECAHEDRON | Lbl R | |
| | ECYLINDER | LblRx Ry Zt Zb | |
| | ELLIPSOID | Lbl Rx Ry Rz | |
| | HEXPRISM | Lbl R Zt Zb | |
| | HOPPER | Lbl Lxt Lyt Lzt Lxb Lyb Lzb | |
| | PARALLELPIPED | Lbl XDIST YDIST ZDIST PHI THETA PHI | |
| | PLANE* | Lbl XPL=a YPL=b ZPL=c CON=d | Data is entered after the = following the respective subordinate keyword. Only non-zero data is required. |
| | QUADRATIC* | Lbl AQU=a BQU=b CQU=c DQU=d EQU=e Lbl FQU=f GQU=g HQU=h IQU=i JQU=j | Data is entered after the = following the respective subordinate keyword. Only non-zero data is required. |
| | RHOMBOID | Lbl DX PSI | |
| | SPHERE | Lbl R | |
| | WEDGE | Lbl XBASE XPT YPT ZLNG | |
| MODIFICATION* | CHORD | +X= -X= +Y= -Y= +Z= -Z= | Only non-zero data is required. |
| | ORIGIN | X= Y= Z= | Only non-zero data is required. |
| | ROTATE | A1= A2= A3= | the body is rotated about the Origin using the Euler angle x-convention. Only non-zero data is required. |
| MEDIA** | MEDIA | Mid Bid i1 i2 ... | |
| | HOLE | Hid i1 i2 ... | the region boundary containing the hole must exactly match the boundary of the unit being placed in the hole. |
| | ARRAY | Aid i1 i2... PLACE nx ny nz xx yy zz | PLACE positions the array by placing the origin of the unit in array position nx ny nz at xx yy zz. The region boundary containing the array must be enclosed in or on the array boundary. |
| BOUNDARY | BOUNDARY | i1 i2 ... | Defines unit boundary. |

*Modification Data can be entered in any order. Any combination of modification data can follow any geometry record.
**The Region Definition Vector, i1, i2, ..., should contain references to all geometry records which share a boundary with the region being defined.

Table C6.4.15 (continued)

**Summary of mixing table data**

MIXTURES — Format: READ MIXT xsec parameters END MIXT

Note: In CSAS6, only xsec parameters should be entered.
KENO-VI mixing table data consists of "xsec parameters" and a "mixing table." The "mixing" table" is used to define the materials used in the KENO-VI problem. These data are automatically created by CSAS6 and provided to KENO-VI. It cannot be overridden. See Sects. F17.4.10 and F17.5.5.

XSEC PARAMETERS — consists of keywords and associated values.
These parameters, if entered, need be entered only once.

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| SCT= | 1 | No. of discrete scattering angles<br>0 is isotropic<br>1 is P1<br>2 is P3<br>3 is P5 |
| EPS= | .00003 | Cross-section message cutoff value<br>Use to suppress message No. K5-60 |

Table C6.4.15 (continued)

## Summary of plot data

PLOT — Format: READ PLOT plot parameters END PLOT plot parameters must be entered for each plot that is to be made.
See Sects. F17.4.11 and F17.5.9

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| TTL= | Prob. title | delim ptitl delim delim is a one-character delimiter that signals the beginning and end of the title. ptitl is the plot title (max. 132 char.) |
| PIC= | MAT | Type of picture: MIXTURE, UNIT NO. or BIAS ID NO. |
| | | MIXTURE ---------- MAT |
| | | MIX |
| | | MIXT |
| | | MIXTURE |
| | | MEDI |
| | | MEDIA |
| | | UNIT NO. ---------- BOX |
| | | BOXT |
| | | BOXTYPE |
| | | UNT |
| | | UNIT |
| | | UNITTYPE |
| | | BIAS ID NO. ------- IMP |
| | | BIAS |
| | | BIASID |
| | | WTS |
| | | WEIG |
| | | WEIGHTS |
| | | WGT |
| | | WGTS |
| XUL= | Prev. plot | X coord. of upper left corner of plot |
| YUL= | Prev. plot | Y coord. of upper left corner of plot |
| ZUL= | Prev. plot | Z coord. of upper left corner of plot |
| XLR= | Prev. plot | X coord. of lower right corner of plot |
| YLR= | Prev. plot | Y coord. of lower right corner of plot |
| ZLR= | Prev. plot | Z coord. of lower right corner of plot |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| UAX= | Prev. plot 0 IF VAX OR WAX is read | X component of direction cosine for the AX axis of the plot (across) |
| VAX= | Prev. plot 0 IF UAX OR WAX is read | Y component of direction cosine for the AX axis of the plot (across) |
| WAX= | Prev. plot 0 IF UAX OR VAX is read | Z component of direction cosine for the AX axis of the plot (across) |
| UDN= | Prev. plot 0 IF VDN OR WDN is read | X component of direction cosine for the DN axis of the plot (down) |
| VDN= | Prev. plot 0 IF UDN OR WDN is read | Y component of direction cosine for the DN axis of the plot (down) |
| WDN | Prev. plot 0 IF UDN OR VDN is read | Z component of direction cosine for the DN axis of the plot (down) |
| DLX= | | Horizontal spacing between points on plot |
| DLD= | | Vertical spacing between points on plot |
| NAX= | | No. of intervals to be printed across page |
| NDN= | | No. of intervals to be printed down page |
| LPI= | 8 | Lines per inch printed down the page |
| NCH= | CHRS** | Delim CHRS delim, a one character delimiter signals the beginning and end of the character string |
| RUN= | YES | YES allows the problem to execute<br>NO terminates problem after data checking |
| PLT= | YES | YES allows the plot(s) to be made<br>NO allows reading the plot data without making a plot |
| SCR= | NO | No specifies a printer plot. If YES, a .gif file is generated for each plot. |
| CLR= | | Used only with SCR=YES to change colors. Input 4 integers: 1st is index into color table, next 3 are the red, green, and blue color values. |

PLOT ORIGIN:

(1) SINGLE UNIT - coincides with origin of geometry description.

(2) BASE ARRAY - at the most negative point of the array (lower left-hand back corner of the global array).

(3) REFLECTED ARRAY - coincides with the origin of the CORE or ARRAY description of the global array.

**default values of CHRS are given below:

MEDIA 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
CHRS 1 2 3 4 5 6 7 8 9 A B C D E F G H I J K L M

MEDIA 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42
CHRS N O P Q R S T U V W X Y Z # , $ - + ) |

MEDIA 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58
CHRS £ > : ; . - % * " = | ( @ < / 0

Table C6.4.15 (continued)

## Summary of starting data

START Format: READ START enter start data here END START
The default value of start type is zero. See Sect. F17.4.8.

| START TYPE | REQUIRED DATA | OPTIONAL DATA | STARTING DISTRIBUTION |
|---|---|---|---|
| 0 | None | NST | Uniform |
| | | XSM | |
| | | XSP | |
| | | YSM | |
| | | YSP | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 1 | NST | XSM | Cosine |
| | | XSP | |
| | | YSM | |
| | | YSPO | |
| | | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |
| 2 | NST | XSM | Cosine with |
| | NXS | XSP | fraction in |
| | NYS | YSM | specified |
| | NZS | YSP | unit |
| | FCT | ZSM | |
| | | ZSP | |
| | | RFL | |
| | | PSP | |

| START TYPE | REQUIRED DATA | OPTIONAL DATA | STARTING DISTRIBUTION |
|---|---|---|---|
| 3 | NST | KFS | Spike |
| | TFX | PSP | |
| | TFY | | |
| | TFZ | | |
| | NXS | | |
| | NYS | | |
| | NZS | | |
| 4 | NST | KFS | Multiple |
| | TFX | PSP | spikes |
| | TFY | | |
| | TFZ | | |
| | NBX | | |
| 5 | NST | PSP | In specified |
| | NBX | | units |
| 6 | NST | NXS | Arbitrary |
| | TFX | NYS | points |
| | TFY | NZS | |
| | TFZ | KFS | |
| | LNU* | PS6 | |
| | | PSP | |

| KEYWORD | DEFAULT | DEFINITION |
|---|---|---|
| NST= | 0 | Start type |
| TFX= | 0.0 | X coordinate |
| TFY= | 0.0 | Y coordinate |
| TFZ= | 0.0 | Z coordinate |
| NXS= | 0 | X index of unit pos. |
| NYS= | 0 | Y index of unit pos. |
| NZS= | 0 | Z index of unit pos. |
| KFS= | | Fission spectra |
| LNU= | 0 | Number of last neutron |
| NBX= | 0 | Source unit number |
| FCT= | 0 | Fraction |
| XSM= | -X | -X of source cuboid |
| XSP= | +X | +X of source cuboid |
| YSM= | -Y | -Y of source cuboid |
| YSP= | +Y | +Y of source cuboid |
| ZSM= | -Z | -Z of source cuboid |
| ZSP= | +Z | +Z of source cuboid |
| RFL= | NO | Start in reflector |
| PS6= | NO | Print start 6 input |
| PSP= | NO | Print starting points |

*LNU must be the last entry for each set of start 6 data. The LNU of each successive set of data must be larger than the last.
Unless otherwise specified, the starting volume is defined by the first positive geometry label in the global unit boundary region definition vector.

## C6.5 NOTES FOR CSAS6 USERS

This section provides tips concerning the use of CSAS6 and its associated sequences. Section M7.5 provides information about the use of the Material Information Processor, and Section F17.7 provides assistance about the use of the KENO-VI data required by CSAS6.

### C6.5.1 CSAS6 MODULES

Several CSAS6 modules are contained in the CSAS6 family. These modules allow the user to generate and save working format cross sections for use with KENO V.a, XSDRNPM, or KENO-VI, or a MORSE/KENO cross-section library for use with MORSE, KENO V.a, or KENO-VI. Therefore, the user can run different codes using the same cross-section library for comparison purposes. A list of the available modules is given below. Table C6.5.1 lists the I/O devices utilized by CSAS6, and Table C6.5.2 lists the type of I/O files utilized by the functional modules invoked by CSAS6.

The CSAS6 input data allow the user to define optional I/O files. The CSAS6 optional parameter data, Section C6.4.8, allow the user to define the following optional files: (1) the unit from which XSDRNPM may read fluxes, (2) the unit on which XSDRNPM may write fluxes, and (3) the unit on which ICE may write an ANISN library. The KENO-VI input data used by CSAS6 allow the user to define optional logical units used by KENO-VI. See "Logical Unit Numbers" in Section F17.4.3. The unit on which a KENO-VI restart data file is written should not be set to 95.

CSAS26 Runs BONAMI, NITAWL-II, and KENO-VI. A working format library is written on unit 4, and a MORSE/KENO mixture cross-section library is written on unit 14. Either or both of these libraries can be saved for future use. Because KENO-VI data are entered as part of the CSAS26 data, LIB= and XSC= should not be specified in the KENO-VI parameter data, and the mixing table should be omitted from the KENO-VI data. All KENO-VI options are available through CSAS26.

CSAS26X Runs BONAMI, NITAWL-II, XSDRNPM, and KENO-VI. A working format library is written on unit 3, and a MORSE/KENO mixture cross-section library is written on unit 14. The cell-weighted mixture 500 is included on unit 14. Either or both of these libraries can be saved for future use. Because KENO-VI data are entered as part of the CSAS26 data, LIB= and XSC= should not be specified in the KENO-VI parameter data, and the mixing table should be omitted from the KENO-VI data. All KENO-VI options are available through CSAS26X. To process cross sections and run KENO-VI, CSAS26 should be used unless use of a cell-weighted mixture in KENO-VI is planned.

Table C6.5.1 Description of units utilized by the Criticality Safety Analysis Sequence No. 6 control module

| Unit number | Type of data | Creating module | User module | Type of file |
|---|---|---|---|---|
| 5 | Input | DRIVER | CSAS6 | Sequential |
| 8 | Scratch file | CSAS6 | CSAS6 | Direct or random access data |
| 11 | Short master cross-section library | CSAS6 | BONAMI | Sequential |
| 16 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 17 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 18 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 19 | Scratch file | CSAS6 | CSAS6 | Sequential |
| 70 | User-supplied cross sections | | CSAS6 | AMPX master format library |
| 81 | 16-group cross sections | | CSAS6 | AMPX master format library |
| 82 | 27-group cross sections | | CSAS6 | AMPX master format library |
| 83 | 44-group cross sections | | CSAS6 | AMPX master format library |
| 84 | 238-group cross sections | | CSAS6 | AMPX master format library |
| 85 | 22 neutron-18 gamma cross sections | | CSAS6 | AMPX master format library |
| 86 | 18-gamma cross sections | | CSAS6 | AMPX master format library |
| 87 | 27-group cross sections for burnup | | CSAS6 | AMPX master format library |
| 88 | 27 neutron-18 gamma cross sections | | CSAS6 | AMPX master format library |
| 89 | Standard Composition Library | CSAS6 | CSAS6 | Direct- or random-access data file |
| 92 | ICE input data | CSAS6 | ICE | Sequential |
| 95 | KENO-VI restart data | CSAS6<br>KENO-VI | KENO-VI | Sequential |
| 96 | BONAMI input data | CSAS6 | BONAMI | Sequential |
| 97 | NITAWL-II input data | CSAS6 | NITAWL-II | Sequential |
| 98 | XSDRNPM input data | CSAS6 | XSDRNPM | Sequential |
| 99 | Printed output | CSAS6 | CSAS6 | Sequential |

Table C6.5.2 Description of units used by the functional modules of CSAS6

| Unit No. | Module | Action | Type of file |
|---|---|---|---|
| 1 | BONAMI | WRITE | Microscopic master working format cross sections |
| | NITAWL-II | READ | Microscopic master working format cross sections |
| 2 | ICE | WRITE | Macroscopic working format cross sections |
| 3 | XSDRNPM | WRITE | Microscopic cell-weighted working format cross sections |
| | ICE | READ | Microscopic cell-weighted working format cross sections |
| | KENO-VI | READ | Microscopic cell-weighted working format cross sections |
| 4 | NITAWL-II | WRITE | Microscopic working format cross sections |
| | ICE | READ | Microscopic working format cross sections |
| | KENO-VI | READ | Microscopic working format cross sections |
| | XSDRNPM | READ | Microscopic working format cross sections |
| 6 | BONAMI | WRITE | Printed output |
| | ICE | WRITE | Printed output |
| | KENO-VI | WRITE | Printed output |
| | NITAWL-II | WRITE | Printed output |
| | XSDRNPM | WRITE | Printed output |
| 8 | BONAMI | WRITE/READ | Direct- or random-access scratch files |
| | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 9 | BONAMI | WRITE/READ | Direct- or random-access scratch files |
| | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct- or random-access scratch files |
| | NITAWL-II | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 10 | ICE | WRITE/READ | Direct- or random-access scratch files |
| | KENO-VI | WRITE/READ | Direct- or random-access scratch files |
| | XSDRNPM | WRITE/READ | Direct- or random-access scratch files |
| 11 | BONAMI | READ | Short AMPX master format cross sections |
| 14 | ICE | WRITE | Monte Carlo-formatted macroscopic cross sections |
| | KENO-VI* | READ/WRITE | Monte Carlo-formatted macroscopic cross sections |
| 16 | KENO-VI | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Angular flux file |
| 17 | XSDRNPM | WRITE/READ | Sequential scratch file |
| 18 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 19 | BONAMI | WRITE/READ | Sequential scratch file |
| | NITAWL-II | WRITE/READ | Sequential scratch file |
| | XSDRNPM | WRITE/READ | Sequential scratch file |
| 79 | KENO-VI | READ | KENO-VI albedo file |
| 80 | KENO-VI | READ | KENO-VI weighting file |
| 92 | ICE | READ | Binary input file |
| 95 | KENO-VI | READ | Binary input file |
| 96 | BONAMI | READ | Binary input file |
| 97 | NITAWL-II | READ | Binary input file |
| 98 | XSDRNPM | READ | Binary input file |

*KENO-VI always reads from unit 14. If microscopic input cross sections are used, KENO-VI prepares macroscopic cross sections and writes them on unit 14 to be read later.

## C6.5.2 AUTOMATIC CROSS-SECTION PROCESSING FOR CRITICALITY CALCULATIONS

The CSASI, CSASIX, CSAS1X, and CSASN moduled, used to prepare an AMPX working format problem-dependent cross-section library, have been disabled in CSAS26. These modules, which can be used to prepare cross-section libraries for use in KENOVI, are contained in CSAS4. CSAS26 and CSAS26X can be used to prepare an AMPX working format cross-section library and run KENO-VI without the necessity of having to enter the mixing table data in KENO-VI. Examples are given in Section C6.5.2.1 through C6.5.2.4.

### C6.5.2.1 Run KENO-VI using CSAS26

CSAS26 creates a microscopic working format library and a mixing table that is passed to KENO-VI. CSAS26 executes KENO-VI and calculates k-effective for the problem.

EXAMPLE 1. CSAS26 - Determine the k-effective of a system.

Consider a problem consisting of four uranium metal cylinders that are 93.2% wt enriched, having a density of 18.76 g/cc. The cylinders are arranged in a 2 × 2 × 2 array. Each has a radius of 5.748 cm and a height of 10.765 cm. The center-to-center spacing in the horizontal plane (X-Y) is 13.74 cm and the vertical center-to-center spacing is 13.01 cm. The input data for this problem follow. The calculated k-effective for this critical experiment was 0.9934 ± 0.0045.

```
=CSAS26
SET UP 2C8 IN CSAS26
HANSEN-ROACH  INFHOMMEDIUM
URANIUM  1 DEN=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
KENO-VI SAMPLE PROBLEM 1  CASE 2C8 BARE
READ PARAMETERS  TME=1.0 FLX=YES FDN=YES FAR=YES
END PARAMETERS
READ GEOMETRY
UNIT 1
CYLINDER 10  5.748 5.3825 -5.3825
CUBOID    20  6.87 -6.87 6.87 -6.87 6.505 -6.505
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY 20
GLOBAL UNIT 2
CUBOID 10 4P13.74  2P13.010
ARRAY  1  10  PLACE 1 1 1 -6.87 -6.87 -6.505
BOUNDARY 10
END GEOMETRY
READ ARRAY  GBL=1 ARA=1 NUX=2 NUY=2 NUZ=2  FILL  F1 END FILL
END ARRAY
END DATA
END
```

EXAMPLE 2. CSAS26 - Determine the k-effective of an array of fuel assemblies.

Consider a 4 × 4 × 1 array of fuel assemblies in a square aluminum cask. Each assembly consists of a 17 × 17 × 1 array of zirconium-clad, 2.35%-enriched $UO_2$ fuel pins in a square pitched array. The $UO_2$ has a density of 9.21 g/cc. The pin diameter is 0.8 cm and is 366 cm long. The clad is 0.07 cm thick, and the pitch is 1.3 cm. Each fuel bundle is contained in a 0.65-cm-thick boral sheath. The bundles are separated by an edge-to-edge spacing of 1 cm. The array of bundles is centered in a 10-cm-thick aluminum cask whose inside dimensions are 0.5 cm beyond the outer edges of the fuel bundles. Determine the k-effective of a dry cask. Input data for this cask follow. The calculated k-effective for this system is 0.7940 ± 0.0035.

```
=CSAS26
SQUARE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2  1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR  2 1 END
H2O  3 1 END
B4C  4 0.367 END
AL  4 0.636 END
AL  5 1 END
END COMP
SQUAREPITCH  1.3 .8 1 3 .94 2 END
SQUARE FUEL CASK EXAMPLE USING DETAILED MOCKUP
READ PARAM  TME=12.0  NUB=YES FAR=YES GEN=153
END PARAM
READ GEOM
UNIT  1
COM='FUEL PIN'
CYLINDER 10  0.4  2P183.0
CYLINDER 20  0.47 2P183.07
CUBOID    30  4P0.65 2P183.07
MEDIA 1 1 10
MEDIA 2 1 20 -10
MEDIA 3 1 30 -20 -10
BOUNDARY 30
UNIT  2
COM='FUEL ASSEMBLY'
CUBOID  10  4P11.05  2P183.07
CUBOID  20  4P11.7   2P183.72
CUBOID  30  4P12.2   2P184.22
ARRAY  1  10  PLACE 9 9 1 3*0.0
MEDIA  4  1  20 -10
MEDIA  3  1  20 -20 -20
BOUNDARY  30
GLOBAL  UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
CUBOID  10  4P48.8  2P184.22
CUBOID  20  4P58.8  2P194.22
```

```
ARRAY 2 10 PLACE 1 1 1 -36.6 -36.6 0.0
MEDIA 5 1 20 -10
BOUNDARY 20
END GEOM
READ ARRAY ARA=1 NUX=17 NUY=17 NUZ=1 FILL F1 END FILL
GBL= 2 ARA=2 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
END
```

### C6.5.2.2 Run KENO-VI using CSAS26X

CSAS26X create a microscopic working format library and a mixing table that is passed to KENO-VI. The microscopic cross sections of the nuclides used in the unit cell geometry description are cell-weighted for use in mixture 500. Mixture 500 utilizes the cell-weighted cross sections that represent the heterogeneous system. CSAS26X executes KENO-VI and calculates k-effective for the problem.

EXAMPLE 1. CSAS26X - Calculate the k-effective of an array of fuel assemblies using cell-weighted cross sections.

Consider the 4 × 4 × 1 array of fuel assemblies in a square aluminum cask described in Section C6.5.2.1, example 3. Calculate the k-effective of the system by using the cell-weighted mixture 500 to represent the fuel pins in the fuel assembly. Note that mixtures 1, 2, and 3, representing $UO_2$, zirconium, and water, respectively, are used in the unit cell description. Cell-weighting is applied to the microscopic cross sections that are used in the cell, making them incorrect for use elsewhere. Because water is used both inside the cell and between the fuel assemblies, an additional mixture, mixture 6, has been added to represent the water between the fuel assemblies. The input data for this problem follow. The calculated k-effective is 0.79400 ± 0.0030.

```
=CSAS26X
SQUARE FUEL CASK EXAMPLE
HANSEN-ROACH LATTICECELL
UO2 1 DEN=9.21 1.0 293. 92235 2.35 92238 97.65 END
ZR 2 1 END
H2O 3 1 END
B4C 4 0.367 END
AL 4 0.636 END
AL 5 1 END
H2O 6 1 END
END COMP
SQUAREPITCH 1.3 .8 1 3 .94 2 END
SAMPLE SQUARE FUEL CASK USING HOMOGENEOUS MOCKUP
READ PARAM TME=12.0 NUB=YES FAR=YES GEN=153
END PARAM
READ GEOM
UNIT 2
COM='FUEL ASSEMBLY'
CUBOID 10 4P11.05 2P183.07
```

```
CUBOID 20 4P11.70 2P183.72
CUBOID 30 4P12.20 2P183.22
MEDIA 500 1 10
MEDIA 4 1 20 -10
MEDIA 6 1 30 -20 -10
BOUNDARY 30
GLOBAL UNIT 3
COM='FUEL CASK CONTAINING 4X4 ARRAY OF ASSEMBLIES'
CUBOID 10 4P48.8 2P184.22
CUBOID 20 4P58.8 2P194.22
ARRAY 1 10 PLACE 1 1 1 -36.6 -36.6 0.0
MEDIA 5 1 20 -10
BOUNDARY 20
END GEOM
READ ARRAY ARA=1 NUX=4 NUY=4 NUZ=1 FILL F2 END FILL
END ARRAY
END DATA
END
```

### C6.5.2.3 Run XSDRNPM using CSAS1X

XSDRNPM can be utilized to calculate the eigenvalue of a system without the necessity of manually preparing the cross sections and mixing table data by executing CSAS1X as shown in example 1 below.

EXAMPLE 1. CSAS1X - Calculate the eigenvalue of a system.

This example is a critical experiment of plutonium nitrate solution in a spherical stainless steel tank reflected by water. The plutonium is 95.4% wt $^{239}Pu$, the remainder being $^{240}Pu$. The fuel is a 0.52-M nitric acid solution of plutonium-nitrate, containing 24.4 g of heavy metal per liter of solution. Optional parameters were entered to decrease the number of mesh intervals and to use an $S_4$ calculation rather than $S_8$. The input data for this problem follow. The eigenvalue calculated by XSDRNPM is 1.00946.

```
=CSAS1X
CRITICAL PU(NO3)4 SOLUTION IN A SS SPHERICAL TANK REFLECTED BY WATER
HANSEN-ROACH MULTIREGION
SOLNPU(NO3)4  1 24.4 .52 1 295 94239 95.4 94240 4.6 END
SS304 2 1.0 END
H2O 3 1 END
END COMP
SPHERICAL VACUUM REFLECTED 0 END
1 19.304 ONEEXTERMOD 2 19.426 ONEEXTERMOD 3 39.746 NOEXTERMOD
END ZONE
MORE DATA ISN=4 SZF=1.5 END
END
```

## C6.5.3 AUTOMATED CROSS-SECTION PREPARATION FOR USE BY SUBSEQUENT CODES

Many of the control modules available through the CSAS4 family of codes are used primarily to process cross sections for use by other codes. Although these modules are not part of CSAS6, the cross sections they produce can be used with KENO VI which is in the CSAS6 family of codes. Some of these applications are described in the following subsections.

### C6.5.3.1 Create a Microscopic, AMPX Working Format, Cross-Section Library for Use by KENO-VI

This example demonstrates the use of CSASN to create a microscopic, AMPX working format library on unit 4. KENO-VI can be executed as part of the same job and use the created cross sections, or the cross-section data set on unit 4 can be saved for later use. It should be noted that the cross-section library created by CSASN is a problem-dependent cross-section library containing only those nuclides specified in the problem. The resonance corrections applied to these cross sections are appropriate only for the specified problem. A simple demonstration of the use of CSASN to create the library and its use by KENO-VI is given in example 1.

EXAMPLE 1

Consider a problem consisting of four uranium metal cylinders that are 93.2% wt enriched, having a density of 18.76 g/cc. The cylinders are arranged in a 2 × 2 × 2 array. Each has a radius of 5.748 cm and a height of 10.765 cm. The center-to-center spacing in the horizontal plane (X-Y) is 13.74 cm, and the vertical center-to-center spacing is 13.01 cm. Use CSASN to create the microscopic, AMPX working format library for use by KENO-VI. The input data for this problem are:

```
=CSASN
CREATE A MICROSCOPIC AMPX WORKING FMT LIBRARY FOR 2C8
HANSEN-ROACH  INFHOMMEDIUM
URANIUM  1 DEN=18.76 1 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
END
=KENOVI
KENO-VI SAMPLE PROBLEM 1  CASE 2C8 BARE
READ PARAMETERS  TME=1.0 FLX=YES FDN=YES FAR=YES LIB=4
END PARAMETERS
READ MIXT MIX=1 92235 4.48006-2 92238 2.6578-3 92234 4.827-4
92236 9.57-5 END MIXT
READ GEOMETRY
UNIT 1
CYLINDER 10  5.748 5.3825 -5.3825
CUBOID    20  6.87 -6.87 6.87 -6.87 6.505 -6.505
MEDIA  1 1 10
MEDIA  0 1 20 -10
BOUNDARY  20
GLOBAL UNIT 2
CUBOID  10  4P13.74  2P13.01
ARRAY  1 10 PLACE 1 1 1 -6.87 -6.87 -6.505
BOUNDARY  10
```

```
END GEOMETRY
READ ARRAY ARA=1  NUX=2 NUY=2 NUZ=2 FILL F1 END FILL
END ARRAY
END DATA
END
```

Note that the KENO-VI data given above are the same as those given in Section C6.5.2.1 except (1) the parameter data specify LIB=4 to instruct KENO-VI where to obtain the cross-section library and (2) the mixing table data must be supplied to KENO-VI. The number densities in the above data were hand-calculated. An easy method of obtaining the mixing table is to run the CSASN problem with PARM=CHECK, capture the output, and delete everything except the mixing table, which is shown in Fig. C6.5.1. This mixing table can then be edited into the correct format for the KENO-VI mixing table as shown in Figs. C6.5.2 and C6.5.3. Either of these mixing tables can be used in the place of the mixing table shown in example 1.

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY | N.I.T. POINTER |
|---|---|---|---|
| 1 | 1092235 | 4.48072E-02 | 274 |
| 1 | 1092238 | 2.65827E-03 | 276 |
| 1 | 1092234 | 4.82825E-04 | 273 |
| 1 | 1092236 | 9.57448E-05 | 275 |

Figure C6.5.1 Mixing table from CSASN with PARM=CHECK

```
READ MIXT
MIX=1       1092235     4.48072E-02
            1092238     2.65827E-03
            1092234     4.82825E-04
            1092236     9.57448E-05
END MIXT
```

Figure C6.5.2 Mixing table from CSASN edited to KENO-VI format

```
READ MIXT
  MIX=1        1092235      4.48072E-02
  MIX=1        1092238      2.65827E-03
  MIX=1        1092234      4.82825E-04
  MIX=1        1092236      9.57448E-05
END MIXT
```

Figure C6.5.3 Mixing table from CSASN edited to KENO-VI format

### C6.5.3.2 Create a Microscopic, Cell-Weighted, AMPX Working Format, Cross-Section Library

EXAMPLE 1. Use CSAS1X to create a microscopic cell-weighted library.

This example demonstrates the use of CSAS1X to create a microscopic, cell-weighted, AMPX working format, cross-section library for use with KENO-VI. CSAS1X produces the microscopic, cell-weighted AMPX working format library on unit 3. All nuclides that are used in the mixtures specified in the unit cell are flux-weighted according to the characteristics of the unit cell and therefore should not be used in any mixtures utilized in KENO-VI other than the cell-weighted mixture. To utilize this cross-section library in KENO-VI, LIB=3 must be specified in the parameter data and an appropriate mixing table must be supplied. The input data for a CSAS1X problem follow:

```
=CSAS1X
PWR 15x15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH LATTICECELL
UO2 1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY 2 1.0 605.0 END
H2O 3 0.719 581.0 END
END COMP
SQUAREPITCH 1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

Because the printed mixing tables from CSASN and CSAS1X with PARM=CHECK do not contain mixing table data for the cell-weighted mixture, the number densities can be calculated by hand or another sequence (CSASIX or CSAS26X with PARM=CHECK) can be used to obtain the mixing table.

EXAMPLE 2. Use CSASIX to create a microscopic cell-weighted library.

A microscopic, cell-weighted, AMPX format library is created on unit 3 when CSASIX is run. The number densities of the cell-weighted mixture are volume averaged. To obtain the mixing table, CSASIX can be run with PARM=CHECK. The second mixing table (labeled COMPLETE MIXING TABLE) will contain the data for the cell-weighted mixture, but the mixture number will not be designated as mixture 500. Instead, it will be one larger than the largest mixture specified in the standard compositions data for the problem. In example 2, the cell-weighted mixture will be designated as mixture 4 in the complete mixing table. The CSASIX data are the following:

```
=CSASIX
PWR 15x15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH  LATTICECELL
UO2  1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY  2 1.0 605.0 END
H2O  3 0.719 581.0 END
END COMP
SQUAREPITCH  1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

The second mixing table from CSASIX is shown in Fig. C6.5.4. In this problem, mixtures 1, 2, and 3 are the only mixtures defined in the input data, and all of them are used in the unit cell description. The cell-weighted mixture is always one larger than the largest defined mixture number; in this case the cell-weighted mixture is mixture 4. All of the mixtures defined in the standard compositions data are used in the unit cell description. Therefore, all of the nuclides used in this problem are cell-weighted, and it is not valid to use them in any mixture other than the cell-weighted mixture. In Fig. C6.5.5, the cell-weighted mixture is mixture 4. In KENO-VI, LIB=3 must be specified in the parameter data in order to access the cell-weighted cross sections. The cell-weighted mixture is usually defined as mixture 500 in KENO-VI.

The mixing table shown in Fig. C6.5.4 can be edited for use in KENO-VI as shown in Fig. C6.5.5. Because mixtures 1, 2, and 3 were used in the unit cell description, they should not be used in KENO-VI. The mixing table shown in Fig. C6.5.4 can be edited by eliminating the ENTRY column and inserting MIX= in front of each mixture number, and inserting a READ MIXT at the beginning of the data and an END MIXT at the end of the data. Since mixtures 1, 2, and 3 should not be used in KENO-VI, they can be edited out as shown in Fig. C6.5.5, or they can be left in the mixing table and not used in the KENO-VI geometry. It should be noted that the easiest way to use a cell-weighted mixture in KENO-VI is to run CSAS26X and utilize mixture number 500 to designate the cell-weighted mixture in the KENO-VI geometry data so the user does not need to create a mixing table.

COMPLETE MIXING TABLE

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY |
|---|---|---|
| 1 | 1092235 | 7.76007E-04 |
| 4 | 1092235 | 2.57581E-04 |
| 1 | 1092238 | 2.24522E-02 |
| 4 | 1092238 | 7.45257E-03 |
| 1 | 1008016 | 4.64564E-02 |
| 4 | 1008016 | 1.54203E-02 |
| 3 | 3008016 | 2.39971E-02 |
| 4 | 3008016 | 1.34078E-02 |
| 2 | 2040302 | 4.25156E-02 |
| 4 | 2040302 | 4.07230E-03 |
| 3 | 3001001 | 4.79943E-02 |
| 4 | 3001001 | 2.68156E-02 |

Figure C6.5.4 Second mixing table from CSASIX

```
READ MIXT   MIX=500
1092235             2.57581E-04
1092238             7.45257E-03
1008016             1.54203E-02
3008016             1.34078E-02
2040302             4.07230E-03
3001001             2.68156E-02
END MIXT
```

Figure C6.5.5 Mixing table from CSASIX edited to KENO-VI format

EXAMPLE 3. Use CSASIX to create a microscopic cell-weighted library with extra mixtures.

This example is identical to example 2 except water and Zircaloy are to be used in the KENO-VI geometry description in addition to the cell-weighted mixture. The temperature of this water and Zircaloy is 450 K. Note that even if the temperatures of the water and Zircaloy were identical to the temperatures specified for the mixtures that are used in the LATTICECELL description, they would have to be specified again with a different mixture number in order to use them in the KENO-VI geometry data. The input data for this example are the following:

```
=CSASIX
PWR 15X15 LATTICECELL CALCULATION AT OPER. T&P
HANSEN-ROACH  LATTICECELL
UO2  1 0.95 1000.0 92235 3.3 92238 96.7 END
ZIRCALLOY  2 1.0 605.0 END
H2O  3 0.719 581.0 END
ZIRCALLOY  4 1.0 450.0 END
H2O  5 0.847 450.0 END
END COMP
SQUAREPITCH  1.430 0.92964 1 3 1.07188 2 0.94844 0 END
END
```

In this example, the cell-weighted mixture will be identified as mixture 7. The mixing tables from CSASIX are shown in Fig. C6.5.6.

MIXING TABLE FROM STANDARD COMPOSITIONS DATA

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY | N.I.T. POINTER |
|---|---|---|---|
| 1 | 1092235 | 7.76007E-04 | 274 |
| 1 | 1092238 | 2.24522E-02 | 276 |
| 1 | 1008016 | 4.64564E-02 | 21 |
| 3 | 3008016 | 2.39971E-02 | 21 |
| 5 | 5008016 | 2.39971E-02 | 21 |
| 2 | 2040302 | 4.25156E-02 | 95 |
| 4 | 4040302 | 4.25156E-02 | 95 |
| 3 | 3001001 | 4.79943E-02 | 4 |
| 5 | 5001001 | 4.79943E-02 | 4 |

COMPLETE MIXING TABLE

| MIXTURE NO. | NUCLIDE NO. | ATOM DENSITY |
|---|---|---|
| 1 | 1092235 | 7.76007E-04 |
| 6 | 1092235 | 2.57581E-04 |
| 1 | 1092238 | 2.24522E-02 |
| 6 | 1092238 | 7.45257E-03 |
| 1 | 1008016 | 4.64564E-02 |
| 6 | 1008016 | 1.54203E-02 |
| 3 | 3008016 | 2.39971E-02 |
| 6 | 3008016 | 1.34078E-02 |
| 5 | 5008016 | 2.82692E-02 |
| 2 | 2040302 | 4.25156E-02 |
| 6 | 2040302 | 4.07230E-03 |
| 4 | 4040302 | 4.25156E-02 |
| 3 | 3001001 | 4.79943E-02 |
| 6 | 3001001 | 2.68156E-02 |
| 5 | 5001001 | 5.65385E-02 |

Figure C6.5.6 Mixing tables from CSASIX

In order to use the microscopic, cell-weighted, cross-section library in KENO-VI, LIB=3 must be specified in the KENO-VI parameter data because the CSAS6 sequences whose names contain an X write the microscopic, cell-weighted, cross-section library on unit 3. The appropriate mixing table for KENO-VI can be created by capturing and editing the COMPLETE MIXING TABLE. The order of scatter should be specified in the KENO-VI mixing table. KENO-VI always defaults to P1 scattering, but CSAS6 sets the order of scattering based on the cross-section library (presently P1 for the Hansen-Roach 16-group library and P3 for all other libraries). In KENO-VI the order of scattering is set using the parameter SCT= in the mixing table data. The resultant mixing table is shown in Figs. C6.5.7 through C6.5.9 below.

```
READ MIXT       SCT=1
 MIX=1          1092235        7.76007E-04
 MIX=500        1092235        2.57581E-04
 MIX=1          1092238        2.24522E-02
 MIX=500        1092238        7.45257E-03
 MIX=1          1008016        4.64564E-02
 MIX=500        1008016        1.54203E-02
 MIX=3          3008016        2.39971E-02
 MIX=500        3008016        1.34078E-02
 MIX=5          5008016        2.82692E-02
 MIX=2          2040302        4.25156E-02
 MIX=500        2040302        4.07230E-03
 MIX=4          4040302        4.25156E-02
 MIX=3          3001001        4.79943E-02
 MIX=500        3001001        2.68156E-02
 MIX=5          5001001        5.65385E-02
END MIXT
```

Figure C6.5.7 KENO-VI mixing table using all CSASIX entries

In Fig. C6.5.7 all of the entries from the CSASIX COMPLETE MIXING TABLE have been edited to conform to the KENO-VI format. In the CSASIX output, mixture 6 is the cell-weighted mixture. Its mixture number has been changed to 500, but it is not necessary to do so. CSASIX and CSAS26X automatically label the cell-weighted mixture cross section as mixture 500 on the macroscopic, cross-section library. This example arbitrarily redefined mixture 6 to be mixture 500 so a user would recognize that cell-weighted cross sections were used in the geometry. MIXTURES 1, 2, AND 3 SHOULD NOT BE USED IN THE KENO-VI GEOMETRY DATA BECAUSE THEY HAVE BEEN FLUX-WEIGHTED OVER THE CELL TO CREATE A CELL-WEIGHTED CROSS SECTION THAT ALLOWS THE HETEROGENEOUS SYSTEM TO BE REPRESENTED AS A HOMOGENEOUS MATERIAL.

```
READ MIXT       SCT=1
 MIX=500        1092235        2.57581E-04
 MIX=500        1092238        7.45257E-03
 MIX=500        1008016        1.54203E-02
 MIX=500        3008016        1.34078E-02
 MIX=5          5008016        2.82692E-02
 MIX=500        2040302        4.07230E-03
 MIX=4          4040302        4.25156E-02
 MIX=500        3001001        2.68156E-02
 MIX=5          5001001        5.65385E-02
END MIXT
```

Figure C6.5.8 KENO-VI mixing table using only mixtures 4, 5 and 6 from CSASIX

In Fig C6.5.8, mixtures 1, 2, and 3 have been removed because they should not be used in the KENO-VI geometry description.

```
READ MIXT       SCT=1
 MIX=4          4040302     4.25156E-02
 MIX=5          5008016     2.82692E-02
                5001001     5.65385E-02
 MIX=500        1092235     2.57581E-04
                1092238     7.45257E-03
                1008016     1.54203E-02
                3008016     1.34078E-02
                2040302     4.07230E-03
                3001001     2.68156E-02
END MIXT
```

Figure C6.5.9 Condensed KENO-VI mixing table using only mixtures 4, 5, and 6 from CSASIX

The mixing table shown in Fig. C6.5.9 has been abbreviated by reordering the data to group all of the components of a mixture together and specifying the mixture number only once for each mixture.

### C6.5.3.3 Create a Macroscopic, AMPX Working Format, Cross-Section Library and a Macroscopic, MORSE/KENO, Cross-Section Library

Example 1 illustrates the use of CSASI to produce a macroscopic, AMPX working format, cross-section library on unit 2 for use by KENO-VI. The ID numbers in this library are the mixtures specified in the standard composition data. To utilize this library in KENO-VI, LIB=2 must be entered in the parameter data, and a mixing table must be provided. The mixing table nuclide ID numbers are the mixture numbers used in the CSASI problem. The number density is a multiplication factor applied to the macroscopic cross sections. Entering a number density of 1.0 causes the cross sections to remain unchanged. Consider a CSASI problem that defines three mixtures and mixture 2 is full-density water. If KENO-VI is to use these cross sections and needs to use half-density water in addition to all of the other defined mixtures, mixture 4 can be defined as half-density water, and the mixing table for KENO-VI would be:

```
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 MIX=4 2 0.5 END MIXT
```

LIB=2 would be entered in the parameter data.

EXAMPLE 1

Consider a problem that consists of four 20.96-kg, 93.2% enriched uranium metal cylinders, density 18.76 g/cc, and four 5-liter Plexiglas bottles filled with highly enriched uranyl nitrate solution at 415 g/l, a specific gravity of 1.555, and 0.39 mg of excess nitrate per gram of solution. The isotopic content of the uranium metal is 93.2% wt $^{235}U$, 5.6% wt $^{238}U$, 1.0% wt $^{234}U$, and 0.2% wt $^{236}U$. The uranium isotopic content of the nitrate solution is 92.6% wt $^{235}U$, 5.9% wt $^{238}U$, 1.0% wt $^{234}U$, and 0.5% wt $^{236}U$. Each metal unit has a diameter of 11.496 cm, a

height of 10.765 cm, and an edge-to-edge spacing of 1.684 cm horizontally and 1.685 cm vertically. The uranyl nitrate solution is contained in a Plexiglas cylinder having an ID of 19.05 cm and an inside height of 17.78 cm. The Plexiglas walls are 0.635 cm thick. The solution units have an edge-to-edge spacing of 1.43 cm horizontally and vertically. The center-to-center spacing between the metal and solution units is 17.465 cm.

CSASI creates a macroscopic, AMPX working format library on unit 2 and a macroscopic, MORSE/KENO library on unit 14. To use the macroscopic, AMPX working format library in KENO-VI, enter LIB=2 in the parameter data, and enter the following mixing table:

```
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 END MIXT
```

The total input data to run CSASI, followed by KENO-VI are given below:

```
=CSASI
SET UP 4AQUEOUS 4 METAL IN CSAS26
HANSEN-ROACH  INFHOMMEDIUM
URANIUM      1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
SOLNUO2(NO3)2  2 415 9.783-3 SPG=1.555 1.0 293 92235 92.6 92238 5.9
                                    92234 1.0  92236 0.5 END
PLEXIGLAS    3 END
END COMP
END
=KENOVI
KENO VI  4 AQUEOUS 4 METAL MIXED UNITS USING CSASI MACRO WORKING LIBR
READ PARAM  FLX=YES FDN=YES  TME=2.0 LIB=2
END PARAM
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 END MIXT
READ GEOM
UNIT 1
CYLINDER 10  9.525 8.89 -8.89
CYLINDER 20  10.16 9.525 -9.525
CUBOID    30  10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 2 1 10
MEDIA 3 1 20 -10
MEDIA 0 1 30 -20
BOUNDARY 30
UNIT 2
CYLINDER 10  5.748 5.3825 -5.3825
CUBOID    10  6.59 -15.16 6.59 -15.16 6.225 -14.255
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY 20
UNIT 3
CYLINDER 10  5.748 5.3825 -5.3825
CUBOID    10  6.59 -15.16 15.16 -6.59 6.225 -14.255
MEDIA 1 1 10
MEDIA 0 1 20 -10
```

```
BOUNDARY 20
UNIT 4
CYLINDER 10 5.748 5.3825 -5.3825
CUBOID    10 6.59 -15.16 6.59 -15.16 14.255 -6.225
MEDIA 1 0 10
MEDIA 0 1 20 -10
BOUNDARY 20
UNIT 5
CYLINDER 10 5.748 5.3825 -5.3825
CUBOID    10 6.59 -15.16 15.16 -6.59 14.255 -6.225
MEDIA 1 1 10
MEDIA 0 1 20 -10
BOUNDARY 20
GLOBAL UNIT 6
CUBOID 10 43.5 0.0 43.5 0.0 40.96 0.0
ARRAY 1 10 PLACE 1 1 1 15.16 15.16 14.255
BOUNDARY 10
END GEOM
READ ARRAY NUX=2 NUY=2 NUZ=2 LOOP
1 3R2 1 2 1 1 2 1 2 9R1 3 3R1 2 2 1 3R1 4 6R1 2 2 1 5 3R1 2 2 1 2 2 1
END ARRAY
END DATA
END
```

Example 2 illustrates the use of CSASI to produce a macroscopic, MORSE/KENO format, cross-section library on unit 14 for use by KENO-VI. The ID numbers in this library are the mixtures specified in the standard composition data. To utilize this library in KENO-VI, XSC=14 must be entered in the parameter data, and a mixing table is not provided. The mixtures used in the KENO-VI geometry must correspond to the mixtures defined in the CSASI data.

EXAMPLE 2

Consider the problem described in example 1. Produce a macroscopic, MORSE/KENO format, cross-section library on unit 14 for use by KENO-VI. To utilize this library in KENO-VI, XSC=14 must be entered in the parameter data, and a mixing table is not provided. The mixture numbers used in the KENO-VI geometry must correspond to the mixtures defined in the standard composition data of CSASI. The input data required to run this problem are the following:

```
=CSASI
SET UP 4AQUEOUS 4 METAL IN CSAS26
HANSEN-ROACH  INFHOMMEDIUM
URANIUM      1 0.985 293  92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
SOLNUO2(NO3)2  2 415 9.783-3 SPG=1.555 1.0 293 92235 92.6 92238 5.9
                                    92234 1.0  92236 0.5 END
PLEXIGLAS    3 END
END COMP
END
=KENOVI
KENO-VI  4 AQUEOUS 4 METAL MIXED UNITS USING CSASI MACRO WORKING LIBR
```

```
READ PARAM  FLX=YES FDN=YES  TME=2.0 XSC=14
END PARAM
READ GEOM
UNIT 1
CYLINDER  10  9.525 8.89 -8.89
CYLINDER  20  10.16 9.525 -9.525
CUBOID     30  10.875 -10.875 10.875 -10.875 10.24 -10.24
MEDIA 2  1  10
MEDIA 3  1  20  -10
MEDIA 0  1  30  -20
BOUNDARY  30
UNIT 2
CYLINDER  10  5.748 5.3825 -5.3825
CUBOID     10  6.59 -15.16 6.59 -15.16 6.225 -14.255
MEDIA 1  1  10
MEDIA 0  1  20  -10
BOUNDARY  20
UNIT 3
CYLINDER  10  5.748 5.3825 -5.3825
CUBOID     10  6.59 -15.16 15.16 -6.59 6.225 -14.255
MEDIA 1  1  10
MEDIA 0  1  20  -10
BOUNDARY  20
UNIT 4
CYLINDER  10  5.748 5.3825 -5.3825
CUBOID     10  6.59 -15.16 6.59 -15.16 14.255 -6.225
MEDIA 1  0  10
MEDIA 0  1  20  -10
BOUNDARY  20
UNIT 5
CYLINDER  10  5.748 5.3825 -5.3825
CUBOID     10  6.59 -15.16 15.16 -6.59 14.255 -6.225
MEDIA 1  1  10
MEDIA 0  1  20  -10
BOUNDARY  20
GLOBAL UNIT  6
CUBOID  10  43.5  0.0  43.5  0.0  40.96  0.0
ARRAY  1  10 PLACE  1  1  1  15.16  15.16  14.255
BOUNDARY 10
END GEOM
READ ARRAY  NUX=2 NUY=2 NUZ=2  LOOP
1 3R2 1 2 1 1 2 1  2 9R1  3 3R1 2 2 1 3R1  4 6R1 2 2 1  5 3R1 2 2 1 2 2 1
END ARRAY
END DATA
END
```

The cross-section libraries created by CSASI are

1. a macroscopic AMPX working format library on unit 2,
2. a macroscopic MORSE/KENO format library on unit 14, and
3. a microscopic AMPX working format library on unit 4.

### C6.5.3.4. Create a Macroscopic, Cell-Weighted, AMPX Working Format Library and a Macroscopic, Cell-Weighted, MORSE/KENO Library

This section illustrates the use of CSASIX to produce cell-weighted, macroscopic libraries for use with a subsequent code. The problem-dependent cross-section libraries created by CSASIX are:

1. A macroscopic, cell-weighted, AMPX working format library on unit 2.
   To use this library in KENO-VI, set LIB=2 in the parameter data and enter a mixing table using the mixture numbers as the nuclide ID's and number densities of 1.0.

2. A microscopic, cell-weighted, AMPX working format library on unit 3.
   To use this library in KENO-VI, set LIB=3 in the parameter data and enter a mixing table using the appropriate nuclide IDs and number densities. If the cell-weighted mixture is to be used, its component nuclides and correct number densities must be included in the mixing table.

3. A microscopic, AMPX working format library on unit 4.
   To use this library in KENO-VI, set LIB=4 in the parameter data and enter the mixing table data. None of the cross sections in this library are cell weighted.

4. A macroscopic cell-weighted MORSE/KENO format library on unit 14.
   To use this library in KENO-VI, set XSC=14 in the parameter data and omit LIB. Do not provide a mixing table. The cell-weighted cross section is mixture 500.

Example 1 shows the use of the macroscopic, cell-weighted, AMPX working format library. Example 2 shows the use of the macroscopic, cell-weighted, MORSE/KENO library.

EXAMPLE 1

This example illustrates the use of CSASIX to create a cell-weighted macroscopic library for use in KENO-VI. The CSASIX data define the materials used to describe a flooded 3 × 3 array of PWR fuel assemblies. The center-to-center spacing between the assemblies is 12 cm. Each fuel assembly is a 17 × 17 array of fuel pins contained in a 0.29-cm-thick boral sheath. The fuel is 3.2% enriched $UO_2$, density = 10.41 g/cc. The fuel OD is 0.819 cm, the gap OD is 0.836 cm, and the clad OD is 9.50 cm. The fuel pin pitch is 1.26 cm. The fuel pin length is 375 cm.

Note that the cell-weighted macroscopic library is specified in KENO-VI by entering LIB=2 in the parameter data and entering mixing table data. The KENO-VI geometry uses mixture 500 to represent the homogenized pin cells inside each fuel assembly. Unit 1 in the geometry describes the fuel assembly and the spacing cuboid that define the center-to-center spacing between assemblies. Unit 2 defines the water-reflected array of fuel assemblies. Because mixture 3 was used in the unit cell definition in CSASIX, another water, mixture 5, was defined to be

used in the KENO-VI geometry because mixtures 1, 2, and 3 have been cell-weighted and cannot be used in the geometry. The input data are the following:

```
=CSASIX
FLOODED PWR FUEL ASSEMBLY (17x17) AT ROOM TEMP
27GROUPNDF4  LATTICECELL
UO2  1  DEN=10.41 1.0 293. 92235 3.2 92238 96.8 END
ZIRCALLOY  2  1.0  END
H2O  3  1.0  END
ARBMTL-B4C 2.64 2 1 1 0 5000 4 6012 1 4 0.015 END
ARBMTL-AL 2.64 1 0 0 0 13027 100.0 4 0.985 END
H2O   5 1.0 END
END COMP
SQUAREPITCH  1.26 0.819 1 3 0.950 2 0.836 0 END
END
=KENOVI
3 X 3 ARRAY OF FLOODED FUEL ASSEMBLIES
READ PARAM LIB=2  TME=5.0  END PARAM
READ MIXT MIX=1 1 1.0 MIX=2 2 1.0 MIX=3 3 1.0 MIX=4 4 1.0 MIX=5 5 1.0
END MIXT
READ GEOM
UNIT 1
COM='HOMOGENIZED FUEL ASSEMBLY'
CUBOID  10  4P10.71 375.0 0.0
CUBOID  20  4P11.0  375.29 -0.29
CUBOID  30  4P12.0  375.29 -0.29
MEDIA  500  1  10
MEDIA  4  1  20  -10
MEDIA  5  1  30  -20  -10
BOUNDARY  30
GLOBAL UNIT 2
COM='WATER REFLECTED 3X3 ARRAY OF FUEL ASSEMBLIES'
CUBOID   10  4P18.0  375.29  -0.29
CUBOID   20  4P21.0  378.29  -3.29
CUBOID   30  4P24.0  381.29  -6.29
CUBOID   40  4P27.0  384.29  -9.29
CUBOID   50  4P30.0  387.29  -12.29
CUBOID   60  4P33.0  390.29  -25.29
CUBOID   70  4P36.0  393.29  -18.29
CUBOID   80  4P39.0  396.29  -21.29
CUBOID   90  4P42.0  399.29  -24.29
CUBOID  100  4P45.0  402.29  -27.29
CUBOID  110  4P48.0  405.29  -30.29
ARRAY  1  10  PLACE 2  2  1  3*0.0
MEDIA  5  2    20   -10
MEDIA  5  3    30   -20
MEDIA  5  4    40   -30
```

```
MEDIA 5 5    50  -40
MEDIA 5 6    60  -50
MEDIA 5 7    70  -60
MEDIA 5 8    80  -70
MEDIA 5 9    90  -80
MEDIA 5 10 100  -90
MEDIA 5 11 110 -100
BOUNDARY 110
END GEOM
READ ARRAY  ARA=1 NUX=3 NUY=3 NUZ=1 FILL F1 END FILL END ARRAY
READ BIAS ID=500 2 11 END BIAS
END
```

EXAMPLE 2

This example is the same as example 1 except the KENO-VI problem will use the macroscopic, MORSE/KENO library. The input data for CSASIX and KENO-VI are given below. Note that XSC=14 is specified in the KENO-VI data, and a mixing table is not entered. However, the mixture numbers used in the geometry correspond to the mixture numbers defined in the standard composition library. Because water, mixture 3, was used in the unit cell definition in CSASIX, another water, mixture 5, has to be defined to be used in the KENO-VI geometry because mixture 3 has been cell-weighted along with mixtures 1 and 2. The input data follow:

```
=CSASIX
FLOODED PWR FUEL ASSEMBLY (17x17) AT ROOM TEMP
27GROUPNDF4  LATTICECELL
UO2  1  DEN=10.41 1.0 293. 92235 3.2 92238 96.8 END
ZIRCALLOY  2  1.0  END
H2O  3  1.0  END
ARBMTL-B4C  2.64  2  1  1  0  5000  4  6012  1  4  0.015  END
ARBMTL-AL  2.64  1  0  0  0  13027  100.0  4  0.985  END
H2O   5  1.0  END
END COMP
SQUAREPITCH  1.26 0.819 1 3 0.950 2 0.836 0 END
END
=KENOVI
3 X 3 ARRAY OF FLOODED FUEL ASSEMBLIES USING MORSE/KENO LIBRARY
READ PARAM XSC=14 TME=5.0  END PARAM
READ GEOM
UNIT 1
COM='HOMOGENIZED FUEL ASSEMBLY'
CUBOID  10  4P10.71 375.0 0.0
CUBOID  20  4P11.0  375.29 -0.29
CUBOID  30  4P12.0  375.29 -0.29
MEDIA  500  1  10
MEDIA  4  1  20  -10
MEDIA  5  1  30  -20  -10
BOUNDARY  30
```

```
GLOBAL UNIT 2
COM='WATER REFLECTED 3X3 ARRAY OF FUEL ASSEMBLIES'
CUBOID   10 4P18.0 375.29 -0.29
CUBOID   20 4P21.0 378.29 -3.29
CUBOID   30 4P24.0 381.29 -6.29
CUBOID   40 4P27.0 384.29 -9.29
CUBOID   50 4P30.0 387.29 -12.29
CUBOID   60 4P33.0 390.29 -25.29
CUBOID   70 4P36.0 393.29 -18.29
CUBOID   80 4P39.0 396.29 -21.29
CUBOID   90 4P42.0 399.29 -24.29
CUBOID  100 4P45.0 402.29 -27.29
CUBOID  110 4P48.0 405.29 -30.29
ARRAY 1 10 PLACE 2 2 1 3*0.0
MEDIA 5 2    20   -10
MEDIA 5 3    30   -20
MEDIA 5 4    40   -30
MEDIA 5 5    50   -40
MEDIA 5 6    60   -50
MEDIA 5 7    70   -60
MEDIA 5 8    80   -70
MEDIA 5 9    90   -80
MEDIA 5 10 100   -90
MEDIA 5 11 110 -100
BOUNDARY 110
END GEOM

READ ARRAY  ARA=1 NUX=3 NUY=3 NUZ=1 FILL F1 END FILL END ARRAY
READ BIAS ID=500 2 11 END BIAS
END
```

## C6.5.4 CREATE AND USE TWO CELL-WEIGHTED MIXTURES IN A PROBLEM

Consider a problem in which a stainless steel cylinder with an ID of 56 cm and an inside height of 91 cm is filled with pellets of $UO_2$ in borated water. The steel is 0.125 cm thick. The spherical 2.57%-enriched $UO_2$ pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.13 cm. The spherical 2.96%-enriched $UO_2$ pellets have a diameter of 1.07 cm and are arranged in a triangular pitch array with a pitch of 1.12 cm. The cylindrical tank is filled half full of the 2.96% pellets in borated water, and the remainder is filled with the 2.57%-enriched pellets in borated water.

Example 1 demonstrates how this problem can be calculated by running a CSASIX case for each of the pellet arrays, combining the macroscopic, AMPX working format, cell-weighted libraries using the AMPX module WAX to combine the two libraries. The combined AMPX working format library can then be used to run the KENO-VI problem. It should be noted that this procedure cannot be stacked together to run in a single SCALE execution because the second CSASIX would destroy the library created by the first CSASIX. Example 2 demonstrates how to add an additional WAX step to move the library created by the first CSASIX so that the steps can be stacked together to run in a single SCALE execution.

EXAMPLE 1

STEP 1. Run CSASIX to obtain a macroscopic, cell-weighted, AMPX working format library on unit 2. The mixture 500 on this library represents the 2.57%- enriched pellets in borated water. The input data for this step are given below. The user must save the library that is created on unit 2. The input data are

```
=CSASIX
2.57% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.57 92238 97.43 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
END COMP
SPHTRIANGP 1.13 1.07  1  2 END
END
```

STEP 2. Run CSASIX to obtain a macroscopic, cell-weighted, AMPX working format library on unit 2. The mixture 500 on this library represents the 2.96%-enriched pellets in borated water. The input data for this step are given below. The user must save the library that is created on unit 2. Note that the problem that is to be run by KENO-VI must have stainless steel available to define the cylindrical container. Therefore, stainless steel must be defined in one of the CSASIX problems. The input data are the following:

```
=CSASIX
2.96% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.96 92238 97.04 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
SS304 3 1.0 283 END
END COMP
SPHTRIANGP 1.12   1.07  1  2 END
END
```

STEP 3. Run WAX to combine the macroscopic, cell-weighted, AMPX working format libraries and rename the mixture 500s to distinguish between them. In this case, mixture 500 for the 2.57%-enriched pellets has been renamed to 501 and mixture 500 for the 2.96%-enriched pellets has been renamed to 502. The 2.57% library is mounted on unit 33, the 2.96% library is mounted on unit 34, and the combined library is written out on unit 4. It is not necessary to rename both of the mixture 500s. One of them can be carried through as 500; the other one, as some other number. WAX can be used to rename all of the mixtures on the library. The cell-weighted mixture created by the CSAS modules is always mixture 500, but WAX can be used to rename it to any desired number. The mixture numbers on the WAX library are the nuclide ID numbers that must be used in the KENO-VI mixing

table when multiple cell-weighted mixtures are used. The user must save the library that is created on unit 4 so that it will be available for use in KENO-VI. The input data follow:

```
=WAX
0$$ 4 34
1$$ 2  1T
2$$ 33 1  2T
3$$ 500
4$$ 501   3T
2$$ 34 2  2T
3$$ 3 500
4$$ 3 502  3T
END
```

STEP 4. Use the combined macroscopic, AMPX working format library to run KENO-VI. Use mixture 501 for the homogenized, cell-weighted, 2.57%-enriched pellets, and use mixture 502 for the homogenized, cell-weighted, 2.96%-enriched pellets. The KENO-VI data are:

```
=KENOV
PROBLEM USING TWO CELL_WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=501 501 1.0  MIX=502 502 1.0 END MIXT
READ GEOM
GLOBAL UNIT 1
CYLINDER  10 38.0 45.5 0.0
CYLINDER  20  38.0 91.0 0.0
CYLINDER  30  38.125 91.0 -0.125
MEDIA  502  1  10
MEDIA  501  1  20  -10
MEDIA    3  1  30  -20  -10
BOUNDARY  30
END GEOM
END DATA
END
```

Alternatively, KENO-VI could have been set up to define mixtures 1 and 2 as the cell-weighted mixtures as follows:

```
=KENOV
PROBLEM USING TWO CELL_WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=1 501 1.0  MIX=2 502 1.0 END MIXT
READ GEOM
GLOBAL UNIT 1
CYLINDER  10 38.0 45.5 0.0
CYLINDER  20  38.0 91.0 0.0
CYLINDER  30  38.125 91.0 -0.125
```

```
MEDIA 2 1 10
MEDIA 1 1 20 -10
MEDIA 3 1 30 -20 -10
END GEOM
END DATA
END
```

EXAMPLE 2

This example is the same as example 1 except a WAX step has been added to move the library created by the first CSASIX from unit 2 to unit 12 so that the steps can be stacked together and run in a single SCALE execution. Because all of the steps are stacked together, unit 12 is used instead of unit 33 because unit 12 is defined in the SCALE procedure and will be automatically passed to the next step. The SCALE input data are:

```
=CSASIX
2.57% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.57 92238 97.43 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
END COMP
SPHTRIANGP 1.13 1.07  1  2 END
END
=WAX
0$$ 12 2
1$$ 1  1T
2$$ 2  1  2T
3$$ 500
4$$ 500    3T
END
=CSASIX
2.96% ENR UO2 PELLETS IN 3500 PPM BORATED WATER
27GROUPNDF4  LATTICECELL
UO2   1  .925  283 92235 2.96 92238 97.04 END
H2O  2 1.0 283 END
ARBMBACID  2.0017-2 3 1 1 0 5000 1 1001 3 8016 3  2  1.0
          283 5010 18.32 5011 81.68 END
SS304 3 1.0 283 END
END COMP
SPHTRIANGP 1.12   1.07  1  2 END
END
=WAX
0$$ 4  2
1$$ 2  1T
'RENAME MIXTURE 500 TO MIXTURE 501
2$$ 12 1  2T
```

```
3$$ 500
4$$ 501   3T
'RENAME MIXTURE 500 TO MIXTURE 502.  MIXTURE 3 REMAINS MIXTURE 3
2$$ 2  2  2T
3$$ 3 500
4$$ 3 502  3T
END
=KENOV
PROBLEM USING TWO CELL-WEIGHTED MIXTURES (502 IS 2.96% & 501 IS 2.57%)
READ PARAM  FLX=YES LIB=4   END PARAM
READ MIXT  MIX=3 3 1.0 MIX=501 501 1.0  MIX=502 502 1.0 END MIXT
READ GEOM
GLOBAL UNIT 1
CYLINDER  10 38.0 45.5 0.0
CYLINDER  20  38.0 91.0 0.0
CYLINDER  30  38.125 91.0 -0.125
MEDIA  502  1  10
MEDIA  501  1  20  -10
MEDIA    3  1  30  -20  -10
END GEOM
END DATA
END
```

## C6.5.5. CREATE A CELL-WEIGHTED MIXTURE REPRESENTING A NONHOMOGENEOUS FUEL ASSEMBLY

Calculate the k-effective of a 3 × 3 × 3 array of the fuel assembly shown in Fig. C6.5.10. The spacing between the assemblies is 2 cm and is filled with water. The array is reflected by 31 cm of water on all faces. Each assembly is a 15 × 15 array of fuel pins with 16 water holes and a central instrumentation hole. The fuel pins are 1.98%-enriched $UO_2$ at 10.138 g/cc and are 0.94 cm in diameter. The active fuel length is 365.76 cm. The zircaloy-4 clad is 0.545 cm OR × 0.4875 cm IR. The fuel rod pitch is 1.44 cm.

The fuel assembly shrouds are 0.16-cm-thick 304 stainless steel. The IDs are 21.6 cm in cross section. The interstices between the rods are filled with water.

To obtain the correct cell-weighted cross sections for this assembly, run CSASIX to define the cell-weighted fuel pin cell. Then run XSDRNPM to account for the water holes.

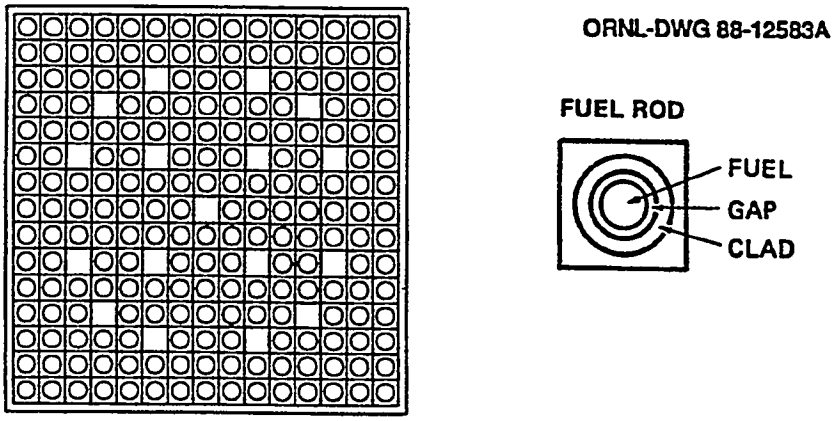



Figure C6.5.10 1.98% enriched $UO_2$ fuel assembly

The input data for CSASIX are given below. Mixtures 1, 2, and 3 are used in the unit cell description, causing them to be cell-weighted and therefore unacceptable for use by a subsequent code. The cross sections for the stainless steel shroud and water holes must be defined so they will be available for use by a subsequent code, XSDRNPM in this case. Mixture 4 is the stainless steel for the fuel assembly shroud, and mixture 5 is the water for the instrumentation hole and water holes. XSDRNPM will create a cell-weighted cross section representing the interior of the fuel assembly. This procedure will use the cell-weighted cross section from CSASIX and the mixture 5 water. Therefore mixture 5 will be cell-weighted and cannot be used in a subsequent code. Since water is to be used in the KENO-VI calculation that uses the cell-weighted cross sections from XSDRNPM, a mixture 6 that is identical to mixtures 3 and 5 must be added to the CSASIX data. The input data follow:

```
=CSASIX
CREATE CELL WEIGHTED CROSS SECTION FOR FUEL PIN CELL
27GROUPNDF4  LATTICECELL
UO2      1 .925   293 92235 1.98 92238 98.02 END
ZIRCALLOY 2 1.0    293 END
H2O      3 1.0    293 END
SS304     4 1.0    293 END
H2O      5 1.0    293 END
H2O      6 1.0    293 END
END COMP
SQUAREPITCH  1.44 .94 1 3 1.09 2 .975 0 END
END
```

The input data for XSDRNPM are given below. Each assembly contains 225 units, 17 of which are water holes (the instrumentation hole is treated as a water hole). The geometry for XSDRNPM is an infinite cylinder of water having an area equal to the area of the central instrumentation hole. The ratio of fuel to water

in the assembly is (225-17)/17; so the area of the fuel region surrounding the water is (225-17)/17 times the area of the water region. The OR of the water hole is 0.81243 cm, and the OR of the fuel region is 2.95566 cm. XSDRNPM will use the macroscopic, cell-weighted, cross-section library that was created by CSASIX on unit 2 and will create a new macroscopic cell-weighted cross-section library on unit 3 as specified in the 0$$ array. Note that XSDRNPM will process only those mixtures that are used in the problem. All mixtures that are to be passed through XSDRNPM for use by a subsequent code must be included in every mixture used in XSDRNPM and should have a very small number density (1.0–20). Therefore, mixtures 4 and 6 must be added to both the water (mixture 5 from CSASIX) and the cell-weighted pin cell mixture (mixture 500 from CSASIX) with a negligible number density. The macroscopic, cell-weighted, cross-section library from CSASIX is read from unit 2, and the resultant cell-weighted cross-section library is written on unit 3. The 3$$ array must specify that a weighting calculation is to be done.

```
=XSDRNPM
USE CELL WTD PIN CELL XSECS AND CREATE CELL WTD FUEL ASSMBLY XSECS
0$$ A3 2 E
1$$ 2 2 20 1 3 4 6 8 3 1 E
2$$ A7 -1 E
3$$ 1 E
4$$ -1 27 E 1T
13$$ 1 2 1 2 1 2
14$$ 5 500 4 4 6 6
15** 1.0 1.0 4R1.-20 2T
33## F1.0 4T
35** 9I0 9I0.812433 2.95566
36$$ 10R1 10R2
51$$ 25I1 27 5T
END
```

The input data for KENO-VI are given below. The cell-weighted cross-section library is read from unit 3. This library contains mixtures 5, 500, 4, and 6. Mixtures 5 and 500 have been cell-weighted to represent the heterogeneous fuel assembly. To create a cell-weighted mixture that represents the fuel assembly, the number densities associated with mixtures 5 and 500 are their respective volume fractions in the fuel assembly. In this problem, mixture 1 is the homogenized fuel assembly mixture. Mixture 4 is the stainless steel shroud, and mixture 6 is water. The number density is 1.0 for mixtures 4 and 6.

```
=KENOVI
WATER REFLECTED 3X3X3 ARRAY OF HOMOGENIZED FUEL ASSEMBLIES
READ PARAM LIB=3 TME=10 END PARAM
READ MIXT MIX=1 5 0.075556 500 0.924444  MIX=4 4 1.0 MIX=6 6 1.0 END MIXT
READ GEOM
UNIT 1
COM='CELL-WEIGHTED HOMOGENIZED FUEL ASSEMBLY WITH ZIRCALLOY SHROUD
AND H2O'
CUBOID  10  4P10.8 365.76 0.0
CUBOID  20  4P10.96  365.92  -0.16
CUBOID  30  4P11.96  366.92  -1.16
MEDIA  1  1  10
```

```
MEDIA 4 1 20 -10
MEDIA 6 1 30 -20
BOUNDARY 30
GLOBAL UNIT 2
CUBOID 10 4P32.88 735.5 -369.24
CUBOID 20 4P62.88 765.5 -399.24
ARRAY 1 10 PLACE 2 2 2 3*0.0
MEDIA 6 1 20 -10
BOUNDARY 20
END GEOM
READ ARRAY NUX=3 NUY=3 NUZ=3 FILL F1 END FILL END ARRAY
END DATA
END
```

## C6.6 DESCRIPTION OF OUTPUT

This section describes the output from the CSAS6 control sequences. The output files that are created when the CSAS6 control sequences are executed consist of (1) output from the control module, (2) output from the functional modules invoked by the control module, and (3) a listing of the input data. The order in which these output files appear is system dependent. If PARM=CHECK is specified on the analytical sequence indicator, there will be no output from the functional modules. The output files from the functional mudules are described in the section pertaining to the module (see Table C6.6.1).

Table C6.6.1 Location of functional module output

| Functional module | Section |
|---|---|
| BONAMI | F1.6 |
| NITAWL-II | F2.8.2 |
| XSDRN | F3.7 |
| ICE | F8.5 |
| KENO-VI | F17.6 |

### C6.6.1 CONTROL MODULE HEADER PAGE

The control module header page, shown in Fig. C6.6.1, prints the name of the control module in block letters. The job name from the job control language, the date (month/day.year), and the time at execution (hour:minute:second) are also printed. This time is given in terms of a 24-h clock, with midnight being 2400 h.



Figure C6.6.1 Sample control module header page

## C6.6.2 CONTROL MODULE PROGRAM VERIFICATION TABLE

The control module program verification information table, shown in Fig. C6.6.2, is printed after the header page. It lists the name of the control module that is being executed, the date the load module was created, the library that contains the load module, the computer code name from the configuration control table, and the revision number. Some computers do not store load modules in a library, in which case the library entry is left blank. The job name, date, and time of execution are also printed. This information may be useful for quality assurance purposes.

```
*********************************************************************************
*********************************************************************************
*********************************************************************************
*****                                                                       *****
*****                  program verification information                     *****
*****                                                                       *****
*****                 code system:    scale  version:    4.2              *****
*****                                                                       *****
*********************************************************************************
*********************************************************************************
*****                                                                       *****
*****                                                                       *****
*****              program:  csas6                                          *****
*****                                                                       *****
*****         creation date:  06/27/95                                      *****
*****                                                                       *****
*****              library:  /scale/exe                                     *****
*****                                                                       *****
*****                                                                       *****
*****       production code:  csas6                                         *****
*****                                                                       *****
*****              version:  2.0                                            *****
*****                                                                       *****
*****              jobname:  dfz                                            *****
*****                                                                       *****
*****     date of execution:  06/28/95                                      *****
*****                                                                       *****
*****     time of execution:  08:53:12                                      *****
*****                                                                       *****
*****                                                                       *****
*********************************************************************************
*********************************************************************************
*********************************************************************************
```

Figure C6.6.2 Sample control module program verification table

## C6.6.3 MATERIAL INFORMATION PROCESSOR OUTPUT

The Material Information Processor output is printed immediately after the program verification table. This output is discussed in detail in Sections M7.6.1 through M7.6.6. An example of this output is given in Figs. C6.6.3 and C6.6.4.

```
 sample problem 1  case 2c8 bare

**** problem parameters ****

lib 27groupndf4  library
mxx            1 mixtures
msc            1 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           0 fuel solutions

**** problem composition description ****

sc  uranium      standard composition
mx             1 mixture no.
vf        1.0000 volume fraction
roth     18.7600 specified density
nel            1 no. elements
icp            1 0/1 mixture/compound
temp       293.0 deg kelvin
           92000      1.00 atom/molecule
                           92235    93.200 wt%
                           92238     5.600 wt%
                           92234     1.000 wt%
                           92236      .200 wt%
end

**** problem geometry ****

**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

Figure C6.6.3 Part 1 of Material Information Processor printout

```
                              sample problem 1  case 2c8 bare

                                  ********** data library information **********

      unit                                                          volume
     number                 data set name                          name        unit function

       89     /scale/datalib/scale.rev10.sclib                                standard composition library

       82     /scale/datalib/scale.rev04.xn27                                 cross section library

       11     ft11f001                                                        short cross section library

       90     ft90f001                                                        input data direct access


                                      standard composition library data
                                      ---------------------------------

          unit number  :  89

          dataset name :  /scale/datalib/scale.rev10.sclib

          library title:  scale-4 standard composition library
                          637 standard compositions, 490 nuclides
                          90 elements with variable isotopic distrbutions.

          creation date:   5/ 3/95


                                        cross section library data
                                        --------------------------

          unit number  :  82

          dataset name :  /scale/datalib/scale.rev04.xn27

          library title:  scale 4.2 - 27 group neutron group library
                             based on endf-b version 4 data
                                compiled for nrc     1/27/89
                                   last updated                                  08/12/94
                                      l.m.petrie   -   ornl
```

Figure C6.6.4 Part 2 of Material Information Processor printout

## C6.6.4 KENO-VI OUTPUT

Output from KENO-VI is printed if the control sequence expects to execute KENO-VI and PARM=CHECK is specified on the analytical sequence indicator. See Sects. F17.6.3 through F17.6.20 for an explanation of the KENO-VI output.

If the control sequence expects to execute KENO-VI and if PARM=CHECK is not specified on the analytical sequence indicator, the KENO-VI output consists of a set of cryptic statements indicating the number of I/Os used to process the KENO-VI data as shown in Fig. C6.6.5. On some platforms the number of I/Os used is not summed so the number of I/Os used is reported as zero.

```
........    0 io's were used before reading keno v data     ........
........    0 io's were used reading the keno-vi parameter data     ........

*************** data reading completed ***************
........    0 io's were used preparing the keno-vi input data     ........
........    0 io's were used loading the keno-vi data     ........
........    0 io's were used loading the data     ........
........    0 io's were used checking the keno-vi geometry data     ........
*****  restart data has been written on unit 95  *****
........    0 io's were used writing the keno-vi - csas data     ........
........    0 io's were used processing csas input data     ........
```

Figure C6.6.5 Summary of KENO-VI I/O usage by CSAS control module

The first line of Fig. C6.6.5, the number of I/Os used before reading KENO-VI data, is always printed for the analytical sequences contained in the CSAS family of control modules. It is the only line of Fig. C6.6.5 that is printed for CSASN, CSAS1X, CSASI, and CSASIX. All other lines of Fig. C6.6.5 are printed by control modules that expect to execute KENO-VI (i.e., CSAS26 and CSAS26X).

## C6.7 WARNING AND ERROR MESSAGES

CSAS6 contains two types of warning and error messages. The first type of message is from the Material Information Processor which is common to many of the SCALE analytical sequences. These messages are prefixed by MP- followed by a number. The Material Information Processor messages are listed in numerical order in Sect. C6.7.1. The second type of message pertaining to both CSAS4 and CSAS6 is identified by CS- followed by a number. These messages are listed in numerical order in Sect. C6.7.2. For additional information concerning a message, simply look up the number in this section.

Warning messages appear when a possible error is encountered. It is the responsibility of the user to verify whether or not the data are correct when a warning message is encountered. The functional modules activated by CSAS6 and related sequences will be executed even though a warning message has been generated.

When an error is recognized, an error message is written and an error flag is set so the functional modules will not be activated. The code stops immediately if the error is too severe to allow continuation of input. However, it will continue to read and check the data if it is able. When the data reading is completed, execution is terminated if an error flag was set when the data were being processed. If the error flag has not been set, execution continues. The STOP codes associated with the severe error messages are listed in Sect. C6.7.3.

### C6.7.1 MATERIAL INFORMATION PROCESSOR MESSAGES

The following messages originate in the Material Information Processor, that portion of the analytical sequence that reads the parameter data, standard compositions specification data, and geometry data necessary for preparing the required information for the cross-section processing codes:

MP-1 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 300 is executed when this message is printed.

MP-2 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 310 is executed when this message is printed.

MP-3 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine KNIGHT indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 320 is executed when this message is printed.

MP-4 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine SETUPA indicates that the allocated computer storage will not hold the directories and isotope information tables. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 330 is executed when this message is printed.

MP-5 ***ERROR***SYSTEM GEOMETRY NAME INCORRECT. INSTEAD OF (_____), THE USER SHOULD HAVE ENTERED LATTICECELL, MULTIREGION, OR INFHOMMEDIUM AS THE SECOND ITEM ON THE PARAMETER CARD. CHECK SPELLING.

This self-explanatory message is from subroutine SETUPB. The user has misspelled the word defining the type of calculation to be performed, or the parameter data preceding it were entered incorrectly. See Sect. C6.4.3 for assistance in correctly specifying the parameter data.

MP-6 ***ERROR*** _____ NOT FOUND IN STANDARD COMPOSITION LIBRARY. MAKE SURE THE COMPOSITION NAME USED IS IN THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message from subroutine STDCMP indicates that a standard composition name is misspelled, the standard composition specification data were out of order, or the "END" has been omitted from the preceding standard composition specification. Check the standard composition specification data carefully. See Sect. C6.4.4 for assistance with standard composition specification cards. See Sect. M8 for the materials available in the Standard Composition Library.

MP-7 ***ERROR*** THE USER ENTERED A VOLUME FRACTION (VF) OF 0.0 FOR ONE OF THE STANDARD COMPOSITIONS (_____). THIS INDICATES THAT HE INTENDS TO ENTER A NUMBER DENSITY FOR THE STANDARD COMPOSITION. THAT, HOWEVER, CAN ONLY BE DONE WHEN THE STANDARD COMPOSITION NAME REPRESENTS A SINGLE NUCLIDE (LIKE H, O, PB, B-10, U-238, ETC.). CHECK THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message is from subroutine STDCMP. Check the Standard Composition Library, Sect. M8, for the available standard composition names. See Sect. C6.4.4 for the correct method of specifying standard composition specification data.

MP-8 ***ERROR*** CELL TYPE NAME (_____) IS INCORRECT. CHECK THE SPELLING. THE ALLOWED NAMES ARE: SQUAREPITCH, TRIANGPITCH, SPHSQUAREP, SPHTRIANGP, SYMMSLABCELL, ASYMSLABCELL, ASQUAREPITCH, ATRIANGPITCH, ASPHSQUAREP, ASPHTRIANGP, ASQP, ATRP, ASSP, ASTP.

This self-explanatory message is from subroutine LATCEL. The data are misspelled or out of order. Some data items may be missing, or extra ones may have been entered. Errors in the standard composition specification data can cause this message to be printed. See Sect. C6.4.6 for additional assistance.

MP-9 ***ERROR*** THE NUMBER OF ZONES (IZM=_____) IS INCORRECT FOR ASYMSLABCELL GEOMETRY. FOR AN ASYMMETRIC LATTICECELL CALCULATION, ONE SHOULD HAVE IZM=3, 5, OR 7, DEPENDING ON THE MATERIALS PRESENT.

This self-explanatory message is from subroutine LATCEL. See Sect. C6.4.6 for an explanation of the ASYMSLABCELL geometry data. A code error is responsible for this message.

MP-10 ***ERROR*** INSUFFICIENT DATA SUPPLIED ON THE GEOMETRY DESCRIPTION CARD.

This message from subroutine LATCEL can be caused by having an error in the standard composition specification data, thereby causing the reading routines to get out of phase. If no error messages precede this message, it may indicate that a code error exists. The number of material zones (IZM) has been destroyed or altered in some manner.

MP-11 ***ERROR*** MIXTURE _____ IN THE CSAS GEOMETRY DESCRIPTION WAS NOT SPECIFIED IN THE STANDARD COMPOSITION DATA.

This message from subroutine LATCEL indicates that a mixture number used in the geometry description was not defined in the standard composition specification data. Check the input data for consistency. If the specified mixture number is zero and if the problem is a LATTICECELL problem using mixture zero for the clad or moderator, it will be necessary to specify a low-density mixture in the standard composition data and use it in the cell specification.

MP-12 ***ERROR*** THE GAP WAS SPECIFIED AS MIXTURE _____ ON THE LATTICECELL. IT SHOULD BE ZERO OR A VALID MIXTURE NUMBER.

This message from subroutine LATCEL indicates that the mixture number used for the GAP was not defined in the standard composition specification data. Check the input data for consistency.

MP-13 **WARNING** NUMBER OF ENTRIES ON THE GEOMETRY DESCRIPTION CARD EXCEEDS THE NUMBER EXPECTED. THIS MESSAGE MAY BE GENERATED IF THE USER FORGOT TO ENTER THE END ON THE GEOMETRY DESCRIPTION CARD.

This message from subroutine LATCEL may result from a code error or from omitting the word "END" from the geometry description card.

MP-14 ***ERROR*** THE COORDINATE SYSTEM NAME (_____) IS INCORRECT. FOR THIS MULTIREGION CALCULATION, THE USER SHOULD HAVE ENTERED SLAB, CYLINDRICAL, SPHERICAL, BUCKLEDSLAB OR BUCKLEDCYL ON THE GEOMETRY DESCRIPTION CARD. CHECK SPELLING.

This self-explanatory message is from subroutine MULTRG.

MP-15 ***ERROR*** ALPHANUMERIC DESCRIPTION OF THE RIGHT-HAND BOUNDARY CONDITION (____) IS INCORRECT. ONLY VACUUM, REFLECTED, PERIODIC, AND WHITE BOUNDARY CONDITIONS ARE ALLOWED. CHECK SPELLING. CHECK GEOMETRY DESCRIPTION CARD INPUT INSTRUCTIONS FOR ADDITIONAL ASSISTANCE.

This self-explanatory message is from subroutine MULTRG. See Sect. C6.4.6 for additional assistance.

MP-16 ***ERROR*** ALPHANUMERIC DESCRIPTION OF THE LEFT-HAND BOUNDARY CONDITION (____) IS INCORRECT. ONLY VACUUM, REFLECTED, PERIODIC, AND WHITE BOUNDARY CONDITIONS ARE ALLOWED IN SLAB GEOMETRY. IN THE CASE OF CYLINDRICAL OR SPHERICAL GEOMETRY, ONLY THE REFLECTED BOUNDARY CONDITION IS ALLOWED ON THE LEFT. CHECK SPELLING. CHECK GEOMETRY DESCRIPTION CARD INPUT INSTRUCTIONS FOR ADDITIONAL INSTRUCTIONS.

This self-explanatory message is from subroutine MULTRG. See Sect. C6.4.6 for additional assistance.

MP-17 ***ERROR*** MIXTURE _____ FROM THE MULTIREGION DATA WAS NOT SPECIFIED IN THE STANDARD COMPOSITION DATA.

This message is from subroutine MULTRG. It indicates that a mixture number specified in the multiregion data was not defined in the standard composition specification data. Check the input data for consistency.

MP-18 ***ERROR*** USER MADE AT LEAST ____ DATA ERRORS. PROBLEM WILL NOT BE RUN. HOPEFULLY, EACH OF THESE ERRORS WILL HAVE GENERATED ITS OWN SELF-EXPLANATORY ERROR MESSAGE. IF NOT, CHECK DATA CAREFULLY BEFORE RESUBMITTING.

This self-explanatory message is from subroutine SETUPB. Check through the problem printout and correct all errors that have occurred.

MP-19 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message is from subroutine SETUPB. It indicates that the allocated computer storage will not hold the data that must be loaded to calculate the information that must be passed to BONAMI. The first number is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that is required to hold these data. The third number is the minimum additional words of storage required to hold these data. Subsequent portions of the program may require even more storage. Increase the region size in the "go step" of the job control language and resubmit the problem. A STOP 390 is executed when this message is printed.

MP-20 ***ERROR*** MIXTURE NUMBER _____ ON A STANDARD COMPOSITION CARD IS LESS THAN 1.

This message is from subroutine SETUPB. A mixture number specified on a standard composition specification card must always be greater than zero. This message is triggered if a MP-6 message is printed. Fixing the MP-6 error should eliminate this message. If the printed mixture number is a positive number, a code error is responsible for the message.

MP-21 ***WARNING*** STANDARD COMPOSITION CARD NUMBER _____ IMPLIES THAT MIXTURE NUMBER____ HAS A TEMPERATURE OF ____ DEGREES KELVIN, WHEREAS STANDARD COMPOSITION SPECIFICATION CARD _____ IMPLIES THAT MIXTURE NUMBER____ HAS A TEMPERATURE OF ____ DEGREES KELVIN. (ONE OF THESE MAY HAVE BEEN THE VALUE SPECIFIED BY DEFAULT.) THE CODE WILL ASSUME THE HIGHER OF THE TWO AND PROCEED. IF THIS IS NOT SATISFACTORY, THE USER SHOULD ENTER THE CORRECT TEMPERATURE ON EACH OF THE STANDARD COMPOSITION SPECIFICATION CARDS INDICATED.

This self-explanatory message is from subroutine SETUPB.

MP-22 **WARNING** STANDARD COMPOSITION SPECIFICATION CARD(S) MISSING FOR MIXTURE NUMBER ____.

This message from SETUPB is printed to remind the user that the mixture numbers specified in the standard composition specification data are not contiguous. This may result in wasted storage space, but is not necessarily indicative of an error.

MP-23 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MTABLE indicates that the allocated computer storage will not hold the nuclide ID numbers that will be processed by subroutine IDENTS. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 340 is executed when this message is printed.

MP-24 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 350 is executed when this message is printed.

MP-25 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 360 is executed when this message is printed.

MP-26 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine MIXTAB indicates that the allocated computer storage will not hold the three mixing table arrays that will be processed by subroutine ARRAY3. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage needed to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 370 is executed when this message is printed.

MP-27 ***ERROR*** THE NUMBER OF MIXTURES FOUND IN THE MIXING TABLE IS INCONSISTENT WITH THE INPUT DATA. THE NUMBER FOUND IS _____, BUT THE INPUT DATA INDICATED _____.

This message from subroutine REORDR indicates that an error was found in the mixing table during operations to reorder and load it in permanent storage. The number of mixtures found in the mixing table should

be the same as the number indicated by the input data. A code error is the likely cause of this message. A STOP 380 is executed when this message is printed.

MP-28 ***ERROR*** NUCLIDE ID ____ SPECIFIED IN THE INPUT DATA WAS NOT FOUND IN THE STANDARD COMPOSITION LIBRARY.

This message is from subroutine IDENTS or subroutine SLNS. A STOP 395 is printed with this message. Check the standard compositions specification data, Sect. C6.4.4, and ensure that the "ZA" ID numbers were entered correctly. This message is also printed if an arbitrary material specifies a value other than 0 or 1 for the MULTIPLE ISOTOPE INDICATOR (IVIS in Sect. C6.4.4).

MP-29 ***ERROR*** AN ARGUMENT OF ____ IS INVALID FOR AN ARCSIN OR ARCCOS.

This message from subroutine ARCHK is indicative of a code error. ARCHK is called from subroutine DAN to ensure that proper arguments are supplied to the ARCSIN and ARCCOS functions in determining cylinder Dancoff factors. This error can be caused by specifying a center-to-center spacing that is smaller than is physically possible.

MP-30 ***ERROR*** THE MATRIX WAS SINGULAR WHEN THE NUMBER FRACTION OF AN ISOTOPE IN AN ELEMENT WAS BEING DETERMINED. MAKE SURE THE SPECIFIED ISOTOPE DISTRIBUTION (WEIGHT PERCENT) SUMS TO 100.0.

This message from subroutine NOFRAC is caused by an isotope distribution that does not sum to 100% for an element. Entry A7 of Sects. C6.4.4 or M7.4.4 illustrates the correct procedure for specifying the weight percent of the various isotopes in an element.

MP-31 ***ERROR*** THE CALCULATED NUMBER FRACTIONS DO NOT REPRODUCE THE WEIGHT PERCENTS ENTERED AS INPUT DATA OR OBTAINED FROM THE STANDARD COMPOSITION LIBRARY.

WTFRAC=nnnnnnnnnn NOFRAC=mmmmmmmmmm MASS=ℓℓℓℓℓℓℓℓℓℓ

This message is printed from subroutine NOFRAC. WTFRAC, NOFRAC, and MASS are printed for each isotope in the standard composition. WTFRAC is the specified input weight percent for the isotope converted to weight fraction instead of weight percent. NOFRAC is the adjusted atom percent converted to atom fraction for the isotope corresponding to a total weight percent of 100% for the isotopes in the standard composition. MASS is the atomic weight of the isotope. The adjusted isotopic weight percents are not printed in MP-31 but can be calculated according to:

$$ADJ_i = \frac{NOFRAC_i \times MASS_i}{\sum_i NOFRAC_i \times MASS_i} \times 100, \qquad \text{(C6.7.1)}$$

where

$ADJ_i$ is the adjusted weight percent of isotope i,

$NOFRAC_i$ is the number fraction of isotope i printed in MP-31,

$MASS_i$ is the atomic weight of isotope i printed in MP-31.

Message MP-31 is printed if the adjusted weight percent of an isotope differs by 0.4% or more from the isotopic weight percent specified for that particular isotope in the standard composition data. An error flag is set, and the problem will not execute. If the isotopic weight percents specified in the input data do not sum to 100%, the weight percent of the first isotope whose specified weight percent is greater than 0.1% is adjusted to force the total weight percent to sum to 100%. If this adjustment results in a calculated weight percent for that isotope that is less than 0.4% different from that specified in the input data, the problem is run using the adjusted data. If message MP-31 is printed, correct the isotopic weight percents specified in the standard composition input data to sum to 100 or specify the isotope with the largest weight percent first, and resubmit the problem.

MP-32 ***ERROR*** END COMP WAS EXPECTED, BUT _____ WAS READ INSTEAD.

This message from subroutine SETUPA is printed if the data are out of order or the keywords END COMP or END COMPOSITIONS are omitted after the last standard composition specification. An END followed by blanks is also allowed. Check the data and the spelling of the END COMP. See Sect. C6.4.4.

MP-33 ***ERROR*** END OF FILE WAS ENCOUNTERED WHILE READING MULTIREGION ZONE DESCRIPTION. VERIFY THAT THE END ZONE CARD WAS ENTERED.

This self-explanatory message from subroutine SETUPA is printed if the END ZONE is omitted following the MULTIREGION zone description data. See Sect. C6.4.7.

MP-34 ***ERROR*** END ZONE WAS EXPECTED, BUT ____ WAS READ INSTEAD.

This message from subroutine SETUPA is printed if data is out of order or the keywords END ZONE are omitted at the end of the MULTIREGION zone description data. Check the data and the spelling of the END ZONE. See Sect. C6.4.7.

MP-35 ***ERROR*** END OF FILE WAS ENCOUNTERED WHEN THE GEOMETRY DESCRIPTION CARD WAS EXPECTED.

This self-explanatory message from subroutine SETUPA is printed if data are omitted or out of order. See Sect. C6.4.

MP-36 ***ERROR*** ILLEGAL SOLUTION NAME _____. ALLOWED NAMES INCLUDE SOLNUO2F2, SOLNUO2(NO3)2 AND SOLNPU(NO3)4.

This self-explanatory message from subroutines SLNS and SLNAME is printed if a solution name is misspelled or data are out of order. See Sect. C6.4.4 for assistance in specifying solution names correctly.

MP-37 ***ERROR*** ____ IS AN INVALID NAME WHEN READING MORE DATA.

This message is printed from subroutine MOREDT when extra parameter data is to be entered as explained in Sect. C6.4.8. The data are out of order, or a parameter name is misspelled.

MP-38 ***ERROR*** INVALID GEOMETRY WORD _____. ONLY INFHOMMEDIUM, MULTIREGION, OR LATTICECELL ARE ALLOWED.

This self-explanatory message is from subroutine SETUPA. Correct the input data and resubmit the problem. Check the spelling of the geometry word, ensure that the data are in order, and resubmit the problem.

MP-39 ***WARNING*** AN END DATA WAS READ BEFORE IT WAS EXPECTED. AN END WAS PROBABLY OMITTED FROM A STANDARD COMPOSITION.

This message from subroutine SETUPA indicates that an "END" was omitted from a standard composition specification or that the END for a standard composition specification was followed by COMP. The first four characters following the END on a standard composition specification card cannot be COMP. The words END COMP terminate the standard composition specification data. Check the standard composition specification data carefully.

MP-40 ***ERROR*** NUCLIDE ID ____ IN MIXTURE ____ DOES NOT OCCUR IN THE STANDARD COMPOSITION LIBRARY.

This self-explanatory message is from subroutine REORDR. A STOP 385 is executed in conjunction with this message.

MP-41 ***ERROR*** A NUCLIDE ID OF _____ IS INVALID FOR _____ IN THE FOLLOWING STANDARD COMPOSITION SPECIFICATION.

This message from subroutine STDCMP indicates that a nuclide ID is not compatible with a standard composition name. For example: UO2 3 1 300 94240 50 94238 50 END would cause this error because plutonium is invalid for a uranium standard composition specification. If this message is printed for a solution, the specific gravity of the solution may have been entered without the required keyword SPG=.

MP-42 ***ERROR*** AN END WAS ENCOUNTERED BEFORE MORE DATA WAS COMPLETED.

This message from MOREDT indicates incomplete input for MORE DATA. See Sect. C6.4.8 or M7.4.8 for assistance.

MP-43 ***WARNING*** BAL=_____ IS AN INVALID BALANCE TABLE SPECIFICATION. ALL BALANCE TABLES WILL BE PRINTED.

This message from subroutine MOREDT indicates that the value entered following BAL= is incorrect. Available options are NONE, FINE, and ALL. If any other value is entered, this message will be printed.

MP-44 *** ERROR *** NUCLIDE _____ IS NOT A MULTIPLE ISOTOPE NUCLIDE ***ERROR***.

This message from subroutine STDCMP indicates that a value greater than zero was entered for the multiple isotope indicator (IVIS) in an arbitrary material and that the first ID NUMBER (NCZA) is not a multiple isotope nuclide (see Sect. C6.4.4). When this message is printed, an error flag is set to prevent execution and IVIS is set to zero to allow the code to continue checking the remaining input data. If an arbitrary material was not specified, this message indicates that the Standard Composition Library is in error for the standard composition printed immediately following this message.

MP-45 ***** ERROR ***** _____ IS NOT A VALID CALCULATION TYPE. USE INFHOMMEDIUM, LATTICECELL, OR MULTIREGION. EXECUTION IS TERMINATED.

This message from subroutine CSPARM indicates that the TYPE OF CALCULATION (see item 3 of Sect. C6.4.3) was spelled incorrectly, entered out of order, or was improperly specified.

MP-46 *** ERROR *** THE FUELOD CANNOT BE GREATER THAN THE PITCH.

This self-explanatory message from subroutine LATCEL results from an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem (see Sect. C6.4.6). The value of PITCH must be at least as large as the value of FUELOD.

MP-47 *** ERROR *** THE CLADOD CANNOT BE LESS THAN THE FUELOD, NOR GREATER THAN THE PITCH.

This message from subroutine LATCEL indicates an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem. See Sect. C6.4.5 and verify the values of PITCH, FUELOD, and CLADOD.

MP-48 *** ERROR *** THE GAPOD CANNOT BE LESS THAN THE FUELOD, NOR GREATER THAN THE CLADOD.

This message from subroutine LATCEL indicates an error in the UNIT CELL SPECIFICATION for a LATTICECELL problem. See Sect. C6.4.6 and verify the values of FUELOD, CLADOD, and CLADID (note that CLADID is the GAPOD and is equal to FUELOD + twice the gap thickness).

MP-49 *** ERROR *** THE ABOVE RADIUS MUST BE GREATER THAN THE PREVIOUS RADIUS.

This message from subroutine MULTRG indicates an error in the GEOMETRY SPECIFICATION for a MULTIREGION problem. See Sect. C6.4.7 and check to ensure that each value of RZ is greater than the previous value of RZ.

MP-50 *** WARNING *** IN MORE DATA, RESONANCE INFORMATION (RES=) WAS SUPPLIED FOR MIXTURE ____. IT WILL NOT BE USED BECAUSE THAT MIXTURE OCCURS IN THE CELL DESCRIPTION.

This message from subroutine MOREDT is printed if resonance data (RES=) are entered for a mixture in MORE DATA (Sect. C6.4.8) and are also used in the cell description (Sects. C6.4.6 and C6.4.7).

MP-51 *** ERROR *** IN MORE DATA _____ IS AN INVALID GEOMETRY TYPE FOR USE IN SPECIFYING RESONANCE DATA.
CYLINDER, SPHERE, OR SLAB MUST BE ENTERED.

This message from subroutine MOREDT is printed if resonance data (RES=) are entered in MORE DATA (Sect. C6.4.8) and an incorrect shape is specified. The correct specification for a sphere of mixture 3 having a radius of 1.5 cm is: RES=3 SPHERE 1.5

MP-52 *** WARNING *** IN MORE DATA, DANCOFF INFORMATION (DAN( )=) WAS SUPPLIED FOR MIXTURE _____. THIS VALUE WILL BE USED INSTEAD OF CALCULATING THE DANCOFF FROM THE CELL DESCRIPTION.

This message from subroutine MOREDT is printed to alert the user that the Dancoff information calculated from the unit cell (Sects. C6.4.6 and C6.4.7) specification is being overridden by the value that was entered in MORE DATA (Sect. C6.4.8).

MP-54 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the volume fraction data from subroutine VERIFY. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-55 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the mixing table data required for subroutine BONAMI. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-56 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the mixture number, outer radii, temperature, sigma-H, and zone identifiers for each BONAMI zone. The first

number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-57 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the BONAMI control parameters. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-58 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the cross-section data required by subroutines EPSIG and XSMESH. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-59 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the data required by subroutine XSDRN. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-60 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutine C4DATA indicates that the allocated computer storage will not hold the data required by subroutine ICE. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem.

MP-61 *** ERROR *** THE CROSS-SECTION LIBRARY NAME ENTERED IN THE MATERIAL INFORMATION PROCESSOR DATA FOR THIS CSAS PROBLEM IS INCORRECT. SEE TABLE M8.2.2 OF THE SCALE MANUAL.

This message from subroutine C4DATA indicates that an invalid CROSS-SECTION LIBRARY NAME (see item 2 of Sect. M7.4.3) was entered. The name was misspelled, out of order, or improperly specified.

MP-62 ***ERROR*** A MIXTURE NUMBER IS OUT OF RANGE. ENTRY _____ IN THE MIXING TABLE IS _____. IT SHOULD FALL BETWEEN 1 AND _____, THE LARGEST NUMBER USED IN THE STANDARD COMPOSITION DATA.

This message is from subroutine EPSIG. Because the largest mixture number is determined by the code as the input data are read, this message may be the result of a code error introduced when changes were made to the code.

MP-63 ***ERROR*** A POINTER INTO THE NUCLIDE DATA IS OUT OF RANGE. ENTRY ___ IN THE MIXING TABLE IS ____. IT SHOULD FALL BETWEEN 1 AND ____, THE NUMBER OF NUCLIDES IN THE STANDARD COMPOSITION LIBRARY.

This message is from subroutine EPSIG. It may result from a code error introduced when changes were made to the code.

MP-64 TAPE COPY NEEDS _____ WORDS, BUT ONLY HAS _____ WORDS.

This message from subroutine SHORTX indicates that the allocated computer storage is insufficient to create the short cross-section library utilized by the Materials Information Processor. Increase the region size for the "go step" and resubmit the problem. A STOP 200 is executed when this message is printed.

MP-65 ***** ERROR ***** THE FOLLOWING NUCLIDES WERE NOT FOUND ON THE MASTER LIBRARY:

This message from subroutine SHORTX indicates that the user specified a standard composition that contains a nuclide that is not available on the specified master library. For example, nuclide ID 25304, which is stainless-steel-weighted manganese, is not available on any of the libraries but is listed in the Standard Composition Library. 25055 could be used instead.

MP-66 ***ERROR*** THE MIX. NO. (MIXZ=_____) ASSIGNED TO ZONE IZ=_____ WAS OUTSIDE THE RANGE 0 TO MXX=_____.

This message from subroutine VERIFY indicates that the mixture number is out of range. A code error is the likely cause of this error.

MP-67 **WARNING** MIXTURE ____ WAS ASSIGNED TO ZONES ____ AND ____. THIS MAY BE CORRECT IF THE MIXTURE HAS NO RESONANCE NUCLIDES AND HAS THERMAL SCATTERING DATA AT ONLY ONE TEMPERATURE. USUALLY, UNIQUE MIXTURE NUMBERS SHOULD BE USED IN EACH ZONE, EVEN THOUGH THE MIXTURE SPECIFICATIONS MAY BE IDENTICAL. THIS ALLOWS BONAMI AND NITAWL TO PRODUCE SEPARATE CROSS SECTIONS BASED ON THE APPROPRIATE NUMBER OF EXTERNAL MODERATORS AND THE ACTUAL MEAN CHORD LENGTH FOR EACH SPECIFIED ZONE AS WELL AS THE ACTUAL TEMPERATURE FOR THAT ZONE.

This self-explanatory message is from subroutine VERIFY. The user should verify that the multiple use of the stated mixture number was intentional and correct.

MP-68 ERRONEOUS MIXING TABLE LENGTH FOR XSDRN.

This message from subroutine XSDRN is indicative of a code error. The computed mixing table length, MS, passed to subroutine XSDRN, does not match the counted length in subroutine XSDRN.

MP-69 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

This message from subroutine XSMESH indicates that the allocated computer storage will not hold the interval boundaries and zone number for each interval. The first number printed is the amount of storage, in words, needed to hold this data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage required to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. Specification (by the control module) of too many mesh intervals in a given zone is the most frequent cause of failure. In MULTIREGION calculations where one has a large zone containing a strong absorber (such as a tank of uranyl-fluoride solution), it may be necessary to reduce the number of mesh intervals by increasing the size factor (SZE) in the optional parameter data (Sect. C6.4.8). Making this factor large enough (SZF=10, 15, 20, 30, etc.) will generally allow any problem to run in the space allocated. Once the problem runs, one should check the number density of all strong absorbers to see if they are realistic. Occasionally a user will accidentally specify an unrealistic concentration of a strong absorber in one or more zones. Such a mistake dramatically increases the number of mesh intervals the code thinks it needs.

MP-70 ***ERROR*** MORE THAN ONE ZONE WAS ENCOUNTERED FOR THIS INFINITE HOMOGENEOUS MEDIA PROBLEM.

This self-explanatory message is from subroutine XSMESH. Correct the input data and resubmit the problem.

MP-71 ***WARNING*** THE TEMPERATURE FOR STANDARD COMPOSITION _____ WAS SPECIFIED AS ____ DEGREE CELSIUS.
DENSITY DATA AT 25 DEGREES CELSIUS WILL BE USED FOR THIS SOLUTION.

This message from subroutine SLNS is printed if SOLNUO2F2 or SOLNPU(NO3)4 were specified in the standard composition data with a temperature outside the range of 15 to 30°C. The Standard Composition Library does not contain data for these solutions outside that temperature range.

MP-72 *** ERROR *** KEYWORD SPG= TO SPECIFY THE SPECIFIC GRAVITY OF THE SOLUTION WAS EXPECTED. KEYWORD _____ WAS READ INSTEAD.

This message from subroutine STDCMP indicates an error in the specification of solution data in the standard composition specification. If the specific gravity of a solution is to be entered, it is the fourth entry after the solution name. Check the solution input data carefully.

MP-73 ***ERROR*** EXTERNAL MODERATOR INDEX IS INCORRECT IN ZONE _____ OF THE MULTIREGION DATA. ENTER NOEXTERMOD, ONE EXTERMOD, OR TWOEXTERMOD INSTEAD OF (_____). CHECK SPELLING.

This self-explanatory message is printed from subroutine MULTRG and subroutine RESDA. The user has misspelled the word defining the external moderator index of the specified zone or the MULTIREGION data are out of order. See Sect. M7.4.4.7.

MP-74 *** ERROR *** FIRST MODERATOR HAS NEGATIVE THICKNESS

This message from subroutine DANCOF indicates that the dimensions specified for an ASYMSLABCELL are inconsistent. The array pitch (PITCH) is smaller than the sum of the second moderator thickness (TKMOD2) and *MAX* where *MAX* is the outside diameter of the clad (CLADOD) if entered; otherwise, *MAX* is the outside diameter of the fuel (FUELOD). Correct the data and resubmit the problem. See Sect. M7.4.6 and Fig. M7.5.6.2 for assistance.

MP-75 *** ERROR *** KEYWORD DEN= OR SPG= TO SPECIFY THE DENSITY OF THE BASIC STANDARD COMPOSITION WAS EXPECTED. KEYWORD _____ WAS READ INSTEAD.

This message from subroutine STDCMP indicates an error in the specification of the density in the basic standard composition data. If the density of the material is to be entered, it is the second entry after the standard composition name. The density is optional data. If the density is entered, it is multiplied by the volume fraction to obtain the density actually used for this composition.

MP-76 *** ERROR *** THE LOWER BOUND OF XSDRN INTERVAL _____ IS _____WHICH MUST BE LESS THAN THE UPPER BOUND OF _____.

This message from subroutine XSMESH indicates that the specified XSDRN mesh interval is incorrect. Each mesh interval must have a finite positive thickness. This message can be caused by an error in the unit cell data. Either a zero-thick region was specified or the unit cell dimensions were incorrectly specified. Carefully check the unit cell dimensions. If a LATTICECELL annular cell is specified, verify that the FUELOD is at least as large as TKMOP2+2(CLADOD-FUELOD).

MP-77 A NEGATIVE DIMENSION WAS ENCOUNTERED IN THE UNIT CELL DATA. CHECK AND CORRECT THE GEOMETRY DIMENSION.

This message from subroutine LATCEL is printed if a negative number is read for a cell dimension. Check the printed problem geometry and correct the input data (Sect. M7.4.6).

MP-78 THE FUEL THICKNESS IS NEGATIVE. THE ANNULAR CELL DIMENSIONS FOR TKMOD2, CLADOD, AND FUELOD ARE INCONSISTENT.

This message from subroutine LATCEL indicates an error in the description of a cylindrical or spherical annular cell. The cell description must provide a finite fuel thickness. The ID of the fuel is calculated using the diameter of the second moderator, TKMOD2, and the OD of the clad, CLADOD. If the calculated ID of the fuel is greater than or equal to the specified OD of the fuel, there is no fuel in the cell and this message is printed. The thickness of the gap and clad is ½(CLADOD–FUELOD), so the ID of the fuel is TKMOD2+CLADOD–FUELOD. See Sect. M7.4.6 for data input instructions.

MP-79 *** ERROR *** THE ORIGIN MUST BE 0.0 FOR SPHERICAL OR CYLINDRICAL GEOMETRY.

This message from subroutine MULTRG is printed if the origin, ORGN of Table M7.4.8, Sect. M.4.7, is nonzero for a cylinder or sphere. Either the MULTIREGION unit cell data are out of order or a nonzero number has been entered for ORGN. Correct the data and resubmit the problem.

MP-80 *** ERROR *** NOT ENOUGH STORAGE IN CLAPSE – FIRST NUMBER IS THE AMOUNT NEEDED, SECOND IS THE AMOUNT AVAILABLE.

PERTINENT CONSTANTS

This message from subroutine CLAPSE indicates that the allocated computer storage is insufficient to allow collapsing the thermal cross sections to one group for shielding calculations. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated. A STOP 201 is executed when this message is printed.

MP-81 *** ERROR *** DANCOFF FACTOR MUST BE BETWEEN 0.0 AND 1.0.

This message from subroutine MOREDT is printed if DAN(n)= is specified in MORE DATA with a value greater than 1.0 or less than 0.0. Dancoff factors must lie between 0 and 1. The problem will not be run if message MP-81 is printed. Correct the data and resubmit the problem. See Sect. M7.4.7 for assistance.

MP-82 THE FIRST MODERATOR THICKNESS IS NEGATIVE. THE ASYMMETRIC SLAB CELL DIMENSIONS FOR TKMOD2, CLADOD, AND PITCH ARE INCONSISTENT

This message from subroutine LATCEL indicates an error in the description of an asymmetric slab cell. The thickness of the first moderator is not specified in the input data and is calculated as PITCH–TKMOD2/2–CLADOD. If the thickness of the first moderator is negative, this message is printed.

MP-83 *** ERROR *** AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE MATERIAL INFORMATION PROCESSOR DATA ENTERED FOR THIS PROBLEM.

This message from subroutine C4DATA is printed if the subroutine library routine LRDERR returns a value of "TRUE," indicating that a reading error was encountered in the Material Information Processor input data. Locate the unnumbered message stating "***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****." Correct the data and resubmit the problem.

MP-84 *** WARNING *** FOR MIXTURE ____ THE DANCOFF FACTOR OF ____ IS OUTSIDE THE RANGE OF 0 TO 1. IT WILL BE RESET TO THE CLOSER BOUND.

This message from subroutine RESDA is printed when the DANCOFF factor is less than $10^{-4}$ outside the allowed range of 0 to 1. This is usually due to roundoff in calculating the Dancoff factor.

MP-85 *** ERROR *** FOR MIXTURE ____ THE DANCOFF FACTOR OF ____ IS OUTSIDE THE RANGE OF 0 TO 1. IT WILL BE RESET TO THE CLOSER BOUND. THE PROBLEM WILL NOT BE EXECUTED.

This message from subroutine RESDA is printed when the Dancoff factor is more than $10^{-4}$ outside the allowed range of 0 to 1. Check the input data to be sure that the specified Dancoff factor is between 0 and 1. See Sect. M7.4.8 or C6.4.8 for input specifications for the Dancoff factor. If the Dancoff factor was not specified in the optional parameter data, the message is due to a code error.

MP-86 ***ERROR*** THE FUEL DENSITY FOR SOLUTION ____ IN MIXTURE ____ SPECIFIES A FUEL DENSITY OF ____ G/L, WHICH EXCEEDS THE FUEL DENSITY OF A HYDRATED CRYSTAL, AND WILL LEAD TO ERRONEOUS DENSITIES WHEN USED AS A SOLUTION

This message from subroutine SLNS is printed only for solution standard compositions (those that begin with SOLN) that are incorrectly specified. Each solution standard composition is composed of (1) a fuel-salt, fs, (2) water, $\mathbf{H_2O}$, and (3) an acid. A fully hydrated fuel-salt is defined as $fs \cdot nH_2O$ where **n** defines the number of water molecules associated with the fuel-salt crystal (water of hydration). The values of **n** are:

| **n** | Solution | Hydrated Salt |
|---|---|---|
| 2 | $UO_2F_2$ | $UO_2F_2 \cdot 2H_2O$ |
| 6 | $UO_2(NO_3)_2$ | $UO_2(NO_3)_2 \cdot 6H_2O$ |
| 5 | $Pu(NO_3)_4$ | $Pu(NO_3)_4 \cdot 5H_2O$ |

If the fuel density is so large that there is not enough water in the solution to form fully hydrated crystals of fuel-salt, this message will be printed. The solution name, mixture number, and specified fuel density are printed in the error message. The problem will not run if this message is printed. Correct the input data for this solution standard composition in the specified mixture and resubmit the problem.

MP-87 ***WARNING*** THE FUEL DENSITY FOR SOLUTION ____ IN MIXTURE ____SPECIFIES A FUEL DENSITY OF ___ G/L. THIS MAY BE GREATER THAN THE SATURATED SOLUTION FUEL DENSITY, AND MAY GIVE INCORRECT DENSITIES AS A SOLUTION. CHECK THE DENSITIES CAREFULLY.

This message from subroutine SLNS is printed because the fuel density for the specified solution standard composition may be high enough to allow precipitation of the fuel-salt. Each solution standard composition is composed of (1) a fuel-salt (**fs**), (2) water ($\mathbf{H_2O}$), and (3) an acid. The water/fuel-salt ratio, **n**, is the ratio of the moles of water to moles of fuel-salt. The minimum threshold values of **n** used to trigger this message are:

| n | Solution |
|---|---|
| 8 | $UO_2F_2$ |
| 17 | $UO_2(NO_3)_2$ |
| 17 | $Pu(NO_3)_4$ |

The solution name, mixture number, and specified fuel density are printed in the error message. The problem will run and it is the responsibility of the user to verify that the resulting solution is correct.

MP-88 ***ERROR*** THE LIBRARY SPECIFIED IS A PRE-NITAWL-II LIBRARY

This error message from subroutine SETUPA indicates the specified library was created to be used with a version of NITAWL prior to NITAWL-II. See Table M4.1.1 for a list of acceptable cross-section libraries. If the library is correctly specified, its format is not consistent with the NITAWL-II format.

MP-89 ***ERROR*** THE MATERIAL SPECIFIED IN THE _____TH STANDARD COMPOSITION SPECIFICATION DOES NOT CONTAIN ANY MULTIPLE ISOTOPE NUCLIDES. ISOTOPIC SPECIFICATIONS ARE NOT ALLOWED.

This error message from subroutine SCDATA indicates that the nth standard composition specification does not contain multiple isotopic nuclides but isotopic data (NCZA and ATPM) were included. It could possibly indicate a missing END statement. When determining the nth specification, arbm and basic standard composition specifications count as one each and solution specifications count as three each (fuel salt, water, and an acid).

MP-90 ***ERROR*** TOO MANY ISOTOPES WERE SPECIFIED FOR THE _____TH ELEMENT OF THE _____TH STANDARD COMPOSITION SPECIFICATION.

This error message from subroutine SCDATA indicates the nth element of the mth standard composition specification contains more isotopes than are allowed. Reduce the number of isotopes used to describe the material. When determining the mth standard composition specification, arbitrary materials and basic standard composition specifications count as one each and solution specifications count as three each (fuel salt, water, and an acid).

MP-91 ***ERROR*** THE USER SPECIFIED A NEGATIVE VOLUME FRACTION FOR _____.

This self-explanatory error message from subroutine STDCMP indicates a negative density multiplier or volume fraction (VF). It was entered for the standard composition named in the message.

MP-92 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0 MIXTURE NUMBER FOR ___.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero mixture number (MX) was entered for the standard composition named in the message.

MP-93 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0 DENSITY FOR _____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero density or specific gravity (SPGR or ROTH) was specified for the standard composition named in the message.

MP-94 ***ERROR*** THE USER SPECIFIED A NEGATIVE OR 0.0 FUEL DENSITY FOR SOLUTION ____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero fuel density (FD) was specified for the solution standard composition named in the message.

MP-95 ***ERROR*** THE USER SPECIFIED A NEGATIVE ACID MOLARITY FOR SOLUTION ____.

This self-explanatory error message from subroutine STDCMP indicates that a negative or zero acid molarity (AML) was specified for the solution standard composition named in the message.

### C6.7.2 MESSAGES FROM CRITICALITY SAFETY SEQUENCE NO. 6

The following messages originate in the part of CSAS6 that reads, checks, and prepares data for KENO-VI. The same set of error messages are also used for CSAS4 that reads, checks, and prepares data for KENO V.a and MODIFY. CSAS6 is not capable of performing searches at this time. An error message referring to a SEARCH routine, from a CSAS26 or CSAS26X problem, indicates a code error.

CS-16 ***WARNING*** READ FLAG NOT FOUND. ASSUME KENO V PARAMETER DATA FOLLOWS.

This message from subroutine CPARAM indicates that the word READ is not the first word of KENO-VI data following the KENO-VI title. If parameter data is to be entered, the code expects the words READ PARAMETERS to precede the parameter input data. If the word READ is not the first word, the code assumes the data are parameter input data.

CS-21 A UNIT NUMBER WAS ENTERED FOR THE CROSS-SECTION LIBRARY. (LIB= IN PARAMETER DATA.) THE DEFAULT VALUE SHOULD BE USED IN ORDER TO UTILIZE THE CROSS SECTIONS GENERATED BY CSAS. MAKE CERTAIN THE CORRECT CROSS-SECTION LIBRARY IS BEING USED.

This message is from subroutine CPARAM. It indicates that a value has been entered for the cross-section library in the KENO-VI parameter data. The cross-section library created by the analytical sequence should be used. MAKE CERTAIN THAT THE CORRECT CROSS SECTIONS ARE BEING USED.

CS-50 *** ERROR *** SEARCH COMMAND NUMBER ____ IS UNABLE TO PERFORM A PITCH SEARCH BECAUSE THE DIMENSIONS OF REGION _____ OF UNIT _____ ARE NOT EXPLICITLY DEFINED.

This message from subroutine PCHSRH indicates that the specified search command is not valid for the specified region. An ARRAY or CORE region cannot be altered; nor can a REPLICATE or REFLECTOR region immediately following an ARRAY or CORE region.

CS-55 *** ERRORS WERE ENCOUNTERED IN PROCESSING THE CSAS-KENO5 DATA. EXECUTION IS IMPOSSIBLE. ***

This message from subroutine SASSY is printed if errors were found in the KENO-VI input data for CSAS. If a search is being made, data reading will continue until all the data have been entered or a fatal error terminates the data reading. When the data reading and checking have been completed, the problem will terminate without executing. Check the printout to locate the errors responsible for this message.

CS-62 *** ERROR *** MIXTURE _____ IN THE GEOMETRY WAS NOT CREATED IN THE STANDARD COMPOSITIONS SPECIFICATION DATA.

This message from subroutine MIXCHK indicates that a mixture specified in the KENO-VI geometry was not created in the standard composition data. If mixture 500 (a cell-weighted, homogenized mixture) is to be used in the KENO-VI geometry data, CSAS2X or CSAS4X must be specified to create it.

CS-65 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. C4DATA WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the material input processor and data preprocessor. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 207 is executed when this message is printed.

CS-66 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. SASSY WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the KENO-VI input processor. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 208 is executed when this message is printed.

CS-67 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. RDOPT WAS NOT CALLED.

This message from the main program, CSAS, is printed if the space requested in the job control language is not enough to hold the program. There is not enough space available to load the processor for the search data. Increase the amount of space requested in the job control language and resubmit the problem. A STOP 209 is executed when this message is printed.

CS-68 *** ERROR *** AN INPUT DATA ERROR HAS BEEN ENCOUNTERED IN THE ____ DATA ENTERED FOR THIS PROBLEM.

This message from the main program, CSAS, is printed if the subroutine library routine LRDERR returns a value of "TRUE," indicating that a reading error has been encountered in the "KENO PARAMETER" data or the CSAS "SEARCH" data. The appropriate data type is printed in the message. Locate the unnumbered message stating "***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****." Correct the data and resubmit the problem.

CS-69 ***ERROR*** MIXTURE ____ IS AN INAPPROPRIATE MIXTURE NUMBER FOR USE IN THE KENO GEOMETRY DATA BECAUSE IT IS A COMPONENT OF THE CELL-WEIGHTED MIXTURE CREATED BY XSDRNPM.

This message from subroutine CMXCHK indicates that a mixture that is a component of a cell-weighted mixture has been used in the KENO-VI geometry data.

CS-70 ***** ERROR ***** SEARCH OR OPTIMIZATION DATA MUST BE ENTERED FOR CSAS6, CSAS6X, AND CSAS3. NO SEARCH DATA WAS ENTERED.

This message from subroutine RDOPT is self-explanatory. If the user does not desire to run a search, another sequence such as CSAS25, CSAS2X, or CSAS1X should be chosen.

CS-71 *** ERROR *** _____ IS NOT A VALID SEARCH TYPE.

This message is from subroutine RDOPT. The allowed search types include OPTIMIZATION, DIMENSION, and CONCENTRATION. The first four characters of the search data after the words READ SEARCH must be OPTI, OPT, OPTM, DIM, DIME, DMSN, CON, or CONC. The data may be misspelled or out of order.

CS-72 *** ERROR *** THE SEARCH TYPE IS INVALID. I=_____.

This message is from subroutine RDOPT. The numerical index, I, should be 1 for an optimum pitch search, 2 for a dimension search, and 3 for a concentration search. If it is none of these, the search type has been incorrectly specified or a code error has been introduced.

CS-73 ***** AN END OF FILE WAS ENCOUNTERED BEFORE ALL THE SEARCH DATA WAS READ.

This self-explanatory message is from subroutine RDOPT. Check the input data to be sure nothing was omitted or misspelled.

CS-74 ***** AN END SEARCH FLAG WAS READ BEFORE ALL THE SEARCH DATA WAS READ.

This self-explanatory message is from subroutine RDOPT. Check the input data for omissions and correct order.

CS-75 *** ERROR *** READ SEARCH FLAG WAS NOT FOUND. _____ WAS READ INSTEAD.

This self-explanatory message is from subroutine RDOPT. READ SEARCH was expected but was not found. Check the input data for omissions and correct order.

CS-76 *** ERROR *** END SEARCH FLAG WAS NOT FOUND. _____ WAS READ INSTEAD.

This self-explanatory message from subroutine RDOPT indicates that an end of file was encountered when looking for READ SEARCH. Check the input data for omissions, correct order, and spelling.

CS-77 *** ERROR *** AN END OF FILE WAS FOUND WHEN THE READ SEARCH FLAG WAS EXPECTED.

This self-explanatory message is from subroutine RDOPT. Check the input data for omissions, correct order, and spelling.

CS-78 *** ERROR *** _____ IS NOT A VALID OPTIMIZATION SEARCH TYPE.

This message is from subroutines FIXTYP and OPTYPE. It indicates that an invalid search type was read. The valid search names include OPTIMUM PITCH, OPTIMUM CONCENTRATION, and OPTIMUM DIMENSION. Either the data was entered improperly or a code error has been introduced. A STOP 215 is executed when this message is printed.

CS-79 *** ERROR *** THE SEARCH TYPE IS INVALID. I=_____.

This message from subroutines FIXTYP and OPTYPE indicates that the search type was misspelled or a code error has been introduced. A STOP 216 is executed when this message is printed.

CS-80 *** ERROR *** SEARCH DATA HAS BEEN DESTROYED. I=____ ICMND=_____ IPNUM= ____ II=____ IGEOM=____

This message from subroutines CNCSRH or DIMSRH indicates that the search data cannot be interpreted. This is probably due to a code error.

CS-81 THE SEARCH WAS PREMATURELY TERMINATED BECAUSE ONLY ____ GENERATIONS WERE COMPLETED IN THE LAST KENO PASS. AT LEAST ____GENERATIONS (NSK= , IN THE KENO PARAMETER DATA) ARE REQUIRED TO SUSTAIN A SEARCH.

This self-explanatory message is from subroutine LODATA. Alter the data and/or the job control language to specify enough time and I/Os to allow completion of a sufficient number of generations in each pass. A STOP 212 is executed when this message is printed.

CS-82 *** AN ERROR WAS ENCOUNTERED IN ONE OF THE FUNCTIONAL MODULES.

This message from MODIFY, the main program of the search package, indicates that an error was encountered during execution of BONAMI, NITAWL, XSDRN, or KENO-VI. Check the printout to locate and correct the error.

CS-83 ***** NO FEASIBLE SOLUTION WAS FOUND IN THE SEARCH PACKAGE. EXECUTION IS TERMINATED. *****

This message from MODIFY, the main program of the search package, is self-explanatory. The code was unable to reach a feasible solution for the requested type of search.

CS-84 **** ERROR **** POINTER INDEX IS OUT OF RANGE. MODIFIED KENO V DATA CANNOT BE WRITTEN. NDX=____.

This message from subroutine RERITE is probably due to a code error. A STOP 213 is executed when this message is printed.

CS-85 *** ERROR *** ALL OF THE ROOTS FOR K=_____ LIE OUTSIDE THE PARAMETER CONSTRAINTS.

This self-explanatory message from subroutine LODATA indicates that a solution does not exist for the specified desired value of k-effective.

CS-86 FIRST TWO NUMBERS ARE WORDS OF STORAGE NEEDED AND ALLOCATED. THIRD IS REQUIRED ADDITIONAL WORDS.

PERTINENT CONSTANTS

This message from subroutines FIXTYP and OPTYP indicates that the allocated computer storage will not hold the information required for a search. The first number printed is the amount of storage, in words, needed to hold these data. The second number is the amount of computer storage that was allocated, and the third number is the minimum additional words of storage necessary to hold the data to this point. Subsequent portions of the program may require even more storage. Increase the region size for the "go step" in the job control language and resubmit the problem. A STOP 211 is executed in conjunction with this message.

CS-87 *** ERROR *** INSUFFICIENT SPACE ALLOCATION. LODATA WAS NOT CALLED.

This message from subroutine MODIFY indicates that the allocated computer storage is not large enough to load the search data. The message is accompanied by a STOP 220. Increase the requested computer storage in the job control language.

CS-88 *** NOT ENOUGH STORAGE TO LOAD THE KENO V MIXING TABLE.

This message from subroutine MXTBLE is printed if the allocated computer storage will not accommodate the KENO-VI mixing table data. Increase the requested computer storage in the job control language.

CS-89 *** ERROR *** _____ IS AN INVALID DIMENSION SEARCH COMMAND.

This message from subroutine DMSN indicates that the dimension search data are out of order, a search command is spelled incorrectly, or the search data are specified incorrectly.

CS-90 *** ERROR *** _____ IS AN INVALID SEARCH PARAMETER.

This message printed from subroutines FIXCNC, FIXDIM, FIXPCH, OPTCNC, OPTDMN, or OPTPCH indicates that a parameter entered in the search type specification data is not valid (see Sect. C6.4.9.1 and Table C6.4.15). The data could be misspelled or out of order. Omission of the keyword MORE before entering the individual search commands (Sect. C6.4.9.2) can cause this error.

CS-91 *** ERROR *** ____ IS AN INVALID CONCENTRATION SEARCH COMMAND.

This message from subroutine CNCN is caused by a misspelled or illegal search command when attempting to do a concentration search.

CS-92 *** ERROR *** ITY IS NOT PITCH(1), DIMENSION(2), OR CONCENTRATION(3). ITY IS PRINTED BELOW.

PERTINENT CONSTANTS

This message from subroutine LODATA indicates that the search data was incorrect when it was reloaded. This error is probably due to a code error.

CS-93 *** ERROR *** NOPT IS NOT MINIMUM (-1), CRITICAL(0), OR MAXIMUM(1). NOPT IS PRINTED BELOW.

PERTINENT CONSTANTS

This message from subroutine LODATA indicates that the search data were incorrect when they were reloaded. This error is probably due to a code error.

CS-94 *** ERROR *** REG=____ IS AN INVALID SEARCH DATA ENTRY. THE REGION NUMBER MUST BE GREATER THAN ZERO AND NO LARGER THAN THE NUMBER OF REGIONS IN THE UNIT.

This message from subroutine DMSN indicates that the region to be altered is incorrectly specified. For example, if the unit being altered contains five geometry regions, the value specified for REG= can be as small as 1 and as large as 5.

CS-95 *** ERROR *** AN ERROR WAS ENCOUNTERED IN THE SEARCH DATA. THE LAST REGION NUMBER MUST BE AT LEAST AS LARGE AS THE FIRST REGION NUMBER. CHECK THE SEARCH DATA PRINTED BELOW.
keyword UNIT ___ REGIONS ___ TO ___ PARAMETER=____
SEARCH CONSTANTS ARE ____ ____ ____ ____ ____ ____

This message from subroutine DMSN indicates that the search data specified an invalid region number for the final region to be altered. Check the printed data and correct as appropriate.

CS-96 *** ERROR *** AN ERROR WAS ENCOUNTERED IN THE SEARCH DATA. THE REGION NUMBERS MUST BE GREATER THAN ZERO AND NO LARGER THAN THE NUMBER OF REGIONS IN THE UNIT. CHECK THE SEARCH DATA PRINTED BELOW.
keyword UNIT ____ REGIONS ____ TO ____ PARAMETER=____
SEARCH CONSTANTS ARE ____ ____ ____ ____ ____ ____

This message from subroutine DMSN indicates that one of the specified region numbers is incorrect. Check the printed data and correct as appropriate.

CS-97 *** ERROR *** NO VALID SEARCH COMMANDS WERE FOUND IN THE DATA.

This message is accompanied by a STOP 235 and is printed from subroutines PCHSRH, CNCSRH, and DIMSRH. A common cause of this error is the omission of the MORE command before the individual search commands are entered. See Sects. C6.4.10.1 and C6.4.10.2.

CS-99 THIS PROBLEM WILL NOT BE RUN BECAUSE PARM=CHECK WAS ENTERED IN THE ANALYTICAL SEQUENCE SPECIFICATION.

This message from subroutine CSAS indicates that the problem data were read and checked and no errors were found. To execute the problem, remove the PARM=CHECK or PARM=CHK from the analytical sequence indicator data entry.

CS-100 THIS PROBLEM WILL NOT BE RUN BECAUSE ERRORS WERE ENCOUNTERED IN THE INPUT DATA.

This message from subroutine CSAS is self-explanatory. Examine the printout to locate the error or errors in the input data. Correct them and resubmit the problem.

### C6.7.3. STOP CODES

The STOP codes that are utilized in CSAS4 and CSAS6 are listed in tabular form, indicating the subroutine where they occur and the associated error message. A STOP is executed when a fatal error is recognized. Locate the associated message number to determine the appropriate corrective measures. A traceback may be generated whenever subroutine STOP is called to print a message. If no traceback is indicated in the STOP CODE table, a STOP is printed at the completion of the associated message.

| Stop No. | Subroutine | Traceback | Associated Message |
|---|---|---|---|
| 20 | | | See Section C6.7.5 |
| 200 | SHORTX | No | MP-64 |
| 201 | CLAPSE | No | MP-80 |
| 203 | MXTBLE | No | CS-88 |
| 207 | CSAS | Yes | CS-65 |
| 208 | CSAS | Yes | CS-66 |
| 209 | CSAS | Yes | CS-67 |
| 211 | OPTYP | No | CS-86 |
| | FIXTYP | No | CS-86 |
| 212 | LODATA | No | CS-81 |
| 213 | RERITE | No | CS-84 |
| 214 | CSAS | Yes | CS-68 |
| 215 | OPTYP | No | CS-78 |
| | FIXTYP | No | CS-78 |
| 216 | OPTYP | No | CS-79 |
| | FIXTYP | No | CS-79 |
| 220 | MODIFY | No | CS-87 |
| 225 | LODATA | Yes | CS-92 |
| 230 | LODATA | Yes | CS-93 |
| 235 | DIMSRH | Yes | CS-97 |
| | OPTSRH | Yes | CS-97 |
| | PCHSRH | Yes | CS-97 |
| 300 | KNIGHT | Yes | MP-1 |
| 310 | KNIGHT | Yes | MP-2 |
| 320 | KNIGHT | Yes | MP-3 |
| 330 | SETUPA | Yes | MP-4 |
| 340 | MTABLE | Yes | MP-23 |
| 350 | MIXTAB | Yes | MP-24 |
| 360 | MIXTAB | Yes | MP-25 |
| 370 | MIXTAB | Yes | MP-26 |
| 380 | REORDR | No | MP-27 |
| 385 | REORDR | No | MP-40 |
| 390 | SETUPB | Yes | MP-19 |
| 395 | IDENTS | No | MP-28 |
| | SLNS | No | MP-28 |
| 396 | CSPARM | No | MP-45 |

### C6.7.4 MESSAGES ASSOCIATED WITH STOP 20 IN CSAS

The following messages from Subroutine Library routines will result in a STOP 20 in the material information processor and CSAS:

LMP001 DA ERROR – INVALID UNIT NUMBER. THE LOGICAL UNIT NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that an invalid unit number was specified as a direct-access device. In CSAS this message is indicative of a code error.

LMP002 DA ERROR – FORTRAN USING THIS UNIT. THE LOGICAL UNIT NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that the specified unit number is open as a sequential dataset rather than a direct-access dataset. In CSAS, this error may be caused by entering a direct-access unit number for FRD= or FWR= in MORE DATA.

LMP003 DA ERROR – DCB NOT OPEN. THE LOGICAL UNIT NUMBER IS _____.

This message from the Subroutine Library direct-access routines indicates that the program attempted to read or write on a direct-access device but that the data control block was not open. In CSAS this message is indicative of a code error.

LMP004 DA ERROR – UNABLE TO OPEN DCB. THE LOGICAL UNIT NUMBER IS _____.

This message from the Subroutine Library direct-access routines indicates that the program is unable to open the data control block for the direct-access device. This message indicates that the job control language did not include proper specification of the above-named unit.

LMP005 DA ERROR – RELATIVE BLOCK NOT IN DATA SET. RELATIVE BLOCK NUMBER IS ____.

This message from the Subroutine Library direct-access routines indicates that the number of direct-access blocks is too small for the problem. Increase the number of direct-access blocks in MORE DATA by entering the parameter DAB=nnn where nnn is larger. For example, if the relative block number is 201, try increasing the number of direct-access blocks to 300 or more. For CSAS25, CSAS2X, CSAS26, CSAS26X, CSAS4, and CSAS4X, it may also be necessary to set NB8=nnn in the KENO parameter data.

LMP006 DA ERROR – INVALID BLOCK LENGTH. THE BLOCK LENGTH IS _____.

This message from the Subroutine Library direct-access routines indicates that the length of the direct-access blocks is invalid. A valid block length must be positive. In CSAS this message is indicative of a code error.

LMP007 DA ERROR – DCB ALREADY OPEN. THE LOGICAL UNIT NUMBER IS _____.

This message from the Subroutine Library direct-access routines indicates that the data control block for the above-named unit was previously opened and not closed.

LMP008 DA ERROR – PERMANENT I/O ERROR.

This message from the Subroutine Library direct-access routines indicates that a permanent I/O error has occurred.

### C6.7.5 UNNUMBERED MESSAGES

Occasionally the code may terminate without printing a message that can be located via the message number. Some of these messages are:

***** END OF FILE READ ON UNIT n *****

This message is from subroutine ENFILE in the Subroutine Library. Unit n is the device from which data is being read. This message can be caused by lack of a double blank between the END for the last standard composition and the END COMP. It can also be caused by omitting the END COMP or misspelling COMP in the END COMP entry. Other causes include incomplete or missing data. A traceback accompanies this message. Correct the data and resubmit the problem.

***** ERROR IN INPUT. CARD IMAGE PRINTED ON NEXT LINE *****
card image is printed here
ON THE ABOVE CARD, CHARACTER NUMBER m (IMAGE=c) IS NOT VALID.

This message is from the Subroutine Library free-form reading routine DREAD. The character number, m, and the character image, c, printed in the message are recognized as being invalid for the type of data being read. For example, character data were encountered when numerical data were expected. This message can be caused by missing or incomplete data and by omitting double blanks after a labeled or unlabeled END. A traceback accompanies this message. Correct the data and resubmit the problem.

## C6.A ALPHABETICAL INDEX OF SUBROUTINES

This section provides a convenient alphabetical index of the subroutines and functions and common blocks used in CSAS.

Table C6.A.1 provides an alphabetical listing of the subroutines and functions that comprise CSAS6. The first column lists the subroutine name, the second column lists the subroutine that calls it, and the third column contains a list of the subroutines it calls. KENO VI subroutines utilized by CSAS6 are contained in brackets <>; Material Information Processor subroutines used by CSAS are contained in braces {}, and FORTRAN-supplied library routines are contained in parentheses ().

Table C6.A.2 provides an alphabetical listing of the subroutines and functions in the control module MODIFY. MODIFY allows the user to perform searches using KENO V.a.

Table C6.A.3 provides an alphabetical index of the common blocks used in CSAS.

Table C6.A.4 provides an alphabetical index of the common blocks used in MODIFY.

Table C6.A.1 Alphabetical index of CSAS routines

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {ADAN} | {DAN} | (ACOS)<br>(ASIN)<br>(COS)<br>(SIN)<br>(SQRT)<br>{GAUSS}<br>{GDAN} |
| <ADJUST> | <HOLCHK> | MOVE |
| (ACOS) | {ADAN}<br>{DAN}<br>{FINNER}<br>{GDAN}<br>{OUTER}<br>{SPHDAN} | |
| (ASIN) | {ADAN}<br>{GDAN}<br>{OUTER} | |
| (COS) | {ADAN}<br>{DAN}<br>{FINNER}<br>{GDAN}<br>{OUTER} | |
| (EXP) | {DANCOF}<br>{DANDAD}<br>{FINNER} | |
| (SIN) | {ADAN}<br>{FINNER}<br>{GDAN}<br>{OUTER} | |
| (SQRT) | {ADAN}<br>{BALL}<br>{DAN}<br>{FINNER<br>{GDAN}<br>{OUTER}<br>{POKE} | |
| (ASIN) | <VOLUME> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| (EXP) | {NFLUX} | |
| (SQRT) | {BONDA}<br><CORSIZ><br><CRMAX><br><CRMIN><br>{DANCOF}<br>{EPSIG}<br><HOLHOL><br>{NFLUX}<br><RDPLOT><br><SRMAX><br><SRMIN><br><VOLUME><br><XRMIN><br><XXLIM><br><XXMIN> | |
| {AGAUSS} | {DAN}<br>{SPHDAN} | |
| <ALBRD> | <DIFALB> | <ALBUSE> |
| <ALBUSE> | <ALBRD> | INQUIR<br>IO<br>RITE |
| ALLOWC | {MOREDT} | |
| ALOCAT | CSAS | |
| <ARALBA> | <JOMITY><br>SASSY | <PRTLBA> |
| <ARASIZ> | <CORSIZ> | <LSCAN> |
| <ARAYIN> | <DATAIN> | AREAD<br><BOX><br>CLEAR<br><HLFWRD><br>IREAD<br><RCHRS><br><RDBOX><br>STOP<br>YREAD |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {ARCHK} | {DAN} | |
| AREAD | <ARAYIN> | |
| | CNCN | |
| | CPARAM | |
| | {CSPARM} | |
| | <DATAIN> | |
| | DMSN | |
| | FIXCNC | |
| | FIXDIM | |
| | FIXPCH | |
| | FIXTYP | |
| | <KENOG> | |
| | {LATCEL} | |
| | <MIXIT> | |
| | {MOREDT} | |
| | {MULTRG} | |
| | OPTCNC | |
| | OPTDMN | |
| | OPTPCH | |
| | OPTYP | |
| | <RCHRS> | |
| | <RDBIAS> | |
| | RDOPT | |
| | <RDORGN> | |
| | <RDPLOT> | |
| | <RDREF> | |
| | <RDSTRT> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {STDCMP} | |
| {ARRAY3} | {MIXTAB} | |
| {AVGB} | {BOND} | |
| {AVGY} | {TWODYS} | MGCWRD |
| | | STOP |
| {AVG1} | {CLAPSE} | |
| {AVG2} | {TWODXS} | MGCWRD |
| {AXSDRN} | {C6DATA} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {BALL} | {DANDAD} | (SQRT)<br>{SPHDAN} |
| {BINFIL} | {ICE}<br>{XSDRN} | |
| {BINRCF} | {BONAMI}<br>{ICE}<br>{NITAWL}<br>{XSDRN} | |
| {BINRCX} | {BONAMI}<br>{ICE}<br>{NITAWL}<br>{XSDRN} | |
| BIRITE | CPARAM | |
| {BONAMI} | {C6DATA} | {BINRCF}<br>{BINRCX}<br>{LCHKMX} |
| {BOND} | {CLAPSE} | {AVGB}<br>{FOLD}<br>IO<br>{UNFOLD} |
| {BONDA} | {RESDA} | (SQRT) |
| <BOX> | <ARAYIN> | |
| <BOXC> | <LOADIT> | |
| {CALCMS} | {C6DATA} | {LCHKMX} |
| CELLMX | {C6DATA} | {LCHKMX} |
| {CINIT} | {C6DATA} | CLEAR<br>INQUIR<br>OPENDA<br>RITE |
| {CLAPSE} | {C6DATA} | {AVG1}<br>{BOND}<br>CLEAR<br>COPY |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | {DEAN} |
| | | IO |
| | | IO4 |
| | | STOP |
| | | {TWODXS} |
| | | {TWODYS} |
| CLEAR | <ARAYIN> | |
| | {CINIT} | |
| | {CLAPSE} | |
| | CNCN | |
| | CPARAM | |
| | CSAS | |
| | {C6DATA} | |
| | <DATAIN> | |
| | DMSN | |
| | <GEOMIN> | |
| | <HOLE> | |
| | <HUNTER> | |
| | <LODARA> | |
| | <LODWTS> | |
| | <MESH> | |
| | {NITAWL} | |
| | <PRTPLT> | |
| | <RDALB> | |
| | <RDICE> | |
| | RDOPT | |
| | <RDPLOT> | |
| | <RDSTRT> | |
| | <RELATE> | |
| | {RESDA} | |
| | <RGUSED> | |
| | <RTARA> | |
| | <SAVST6> | |
| | {SETUPB} | |
| | <SORTA> | |
| | {TWODXS} | |
| | {TWODYS} | |
| | <UNTCRS> | |
| | <VOLUME> | |
| | {XSDRN} | |
| CLOSDA | CSAS | |
| | {KNIGHT} | |
| | {SHORTX} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| CMXCHK | SASSY | {LCHKMX} |
| CNCN | FIXTYP | AREAD |
| | OPTYP | CLEAR |
| | | FREAD |
| | | INQUIR |
| | | IREAD |
| | | RITE |
| CNCSRH | FIXTYP | PRTSRH |
| | OPTYP | RITE |
| | | STOP |
| COPY | {CLAPSE} | |
| | {SHORTX} | |
| | {TWODXS} | |
| | {TWODYS} | |
| <CORSIZ> | <JOMITY> | (SQRT) |
| | | <ARASIZ> |
| CPARAM | CSAS | AREAD |
| | | BIRITE |
| | | CLEAR |
| | | FREAD |
| | | IREAD |
| | | LIBEQL |
| | | OPNFIL |
| | | RNDIN |
| | | RNDOUT |
| | | TIMFAC |
| | | ZREAD |
| CREAD | <RDREF> | |
| <CRMAX> | <HOLEXT> | (SQRT) |
| | <JOMCHK> | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |
| | | <VECADD> |
| | | <VECNRM> |
| <CRMIN> | <HOLHOL> | (SQRT) |
| | | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <VECADD> |
| | | <VECNRM> |
| <CRSPRD> | <CRMAX> | |
| | <CRMIN> | |
| | <SRMAX> | |
| | <SRMIN> | |
| CSAS | | ALOCAT |
| | | CLEAR |
| | | CLOSDA |
| | | CPARAM |
| | | {C6DATA} |
| | | GETMS |
| | | INDEX |
| | | IOLEFT |
| | | <IOSDUN> |
| | | OPNFIL |
| | | PARTBL |
| | | RDOPT |
| | | RESETM |
| | | SASSY |
| | | SCANON |
| | | STOP |
| {CSPARM} | {SETUPA} | AREAD |
| | | LCOMPR |
| | | {SMMTBA} |
| {C6DATA} | CSAS | {AXSDRN} |
| | | {BONAMI} |
| | | {CALCMS} |
| | | CELLMX |
| | | {CINIT} |
| | | {CLAPSE} |
| | | CLEAR |
| | | {EPSIG} |
| | | {ICE} |
| | | {KNIGHT} |
| | | {MIPCHK} |
| | | {NITAWL} |
| | | {PRELPS} |
| | | {SHORTX} |
| | | {SMMTBA} |
| | | STOP |
| | | {VERIFY} |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | {XSDRN} |
| | | {XSMESH} |
| {DAN} | {DANDAD} | {ADAN} |
| | | (ACOS) |
| | | (COS) |
| | | (SQRT) |
| | | {AGAUSS} |
| | | {ARCHK} |
| | | ERRSET |
| {DANCOF} | {RESDA} | (EXP) |
| | | (SQRT) |
| | | {DANDAD} |
| | | E3 |
| | | SHELC |
| | | STOP |
| {DANDAD} | {DANCOF} | (EXP) |
| | | {BALL} |
| | | {DAN} |
| | | {SDAN} |
| <DATAIN> | SASSY | <ARAYIN> |
| | | AREAD |
| | | CLEAR |
| | | <EXTRA> |
| | | <FLDATA> |
| | | <GEOMIN> |
| | | <IDX1D> |
| | | IOWRT |
| | | <MIXIT> |
| | | <RDBIAS> |
| | | <RDPLOT> |
| | | <RDREF> |
| | | <RDRST> |
| | | <RDSTRT> |
| | | <RT> |
| | | <RTARA> |
| | | <SAVST6> |
| | | STOP |
| | | <WRTPLT> |
| {DEAN} | {CLAPSE} | {KAUL} |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <DIFALB> | <FLDATA> | <ALBRD> |
| DIMSRH | FIXTYP<br>OPTYP | PRTSRH<br>RITE<br>STOP |
| DMSN | FIXTYP<br>OPTYP | AREAD<br>CLEAR<br>FREAD<br>INQUIR<br>IREAD<br>RITE |
| <DOTPRD> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN><br><XRMIN> | |
| DTASET | {SHORTX} | |
| {EPSIG} | {C6DATA} | (SQRT) |
| ERRSET | {DAN} | |
| <EXTRA> | <DATAIN> | |
| E3 | {DANCOF}<br>{SDAN} | |
| {FAX} | {STDCMP} | FREAD<br>IREAD |
| {FAXPRT} | {STDCMP} | |
| <FINDBX> | <LOCATE> | |
| {FINNER} | {OUTER} | (ACOS)<br>(COS)<br>(EXP)<br>(SIN)<br>(SQRT) |
| {FIX} | {STDCMP} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| FIXCNC | FIXTYP | AREAD |
| | | FREAD |
| | | IREAD |
| FIXDIM | FIXTYP | AREAD |
| | | FREAD |
| | | IREAD |
| FIXPCH | FIXTYP | AREAD |
| | | FREAD |
| | | IREAD |
| | | REED |
| FIXTYP | RDOPT | AREAD |
| | | CNCN |
| | | CNCSRH |
| | | DIMSRH |
| | | DMSN |
| | | FIXCNC |
| | | FIXDIM |
| | | FIXPCH |
| | | PCHSRH |
| | | REED |
| | | STOP |
| <FLDATA> | <DATAIN> | <DIFALB> |
| | | REED |
| | | <RGUSED> |
| | | RITE |
| | | <SORTA> |
| | | <SORTR> |
| | | STOP |
| | | <WATES> |
| {FOLD} | {BOND} | |
| {FOX} | {STDCMP} | |
| FREAD | CNCN | |
| | CPARAM | |
| | DMSN | |
| | {FAX} | |
| | FIXCNC | |
| | FIXDIM | |
| | FIXPCH | |
| | {LATCEL} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <MIXIT><br>{MOREDT}<br>{MULTRG}<br>OPTCNC<br>OPTDMN<br>OPTPCH<br><RDBIAS><br><RDORGN><br><RDPLOT><br><RDSTRT><br>{SETUPA}<br>{STDCMP}<br><XXIN> | |
| {GAUSS} | {ADAN}<br>{OUTER} | |
| {GDAN} | {ADAN} | (ACOS)<br>(ASIN)<br>(COS)<br>(SIN)<br>(SQRT)<br>KI3 |
| <GEOMIN> | <DATAIN> | CLEAR<br><KENOG><br><READGM><br>STOP |
| GETMS | CSAS | |
| GETPTR | <RCHRS> | |
| {GFLUX} | {PRELPS} | |
| <GOCURS> | <VOLUME> | <HUNTER> |
| <GTVOLS> | <VOLUME> | |
| <HLFWRD> | <ARAYIN> | |
| {HOGGIE} | {TWODYS} | STOP |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <HOLCHK> | <JOMCHK> | <ADJUST><br><HOLEXT><br><HOLHOL> |
| <HOLE> | <LODARA> | CLEAR<br>STOP |
| <HOLEXT> | <HOLCHK> | <CRMAX><br><SRMAX><br><XXLIM> |
| <HOLHOL> | <HOLCHK> | (SQRT)<br><CRMIN><br><SRMIN><br><XRMIN><br><XXMIN> |
| <HUNTER> | <GOCURS><br><VOLUME> | CLEAR<br>MOVE |
| {ICE} | {C6DATA} | {BINFIL}<br>{BINRCF}<br>{BINRCX}<br>{VOLFRC} |
| {IDENTS} | {MTABLE} | {ISCAN}<br>{NOFRAC} |
| <IDX1D> | <DATAIN> | IREAD |
| INDEX | CSAS<br><RTADJ> | |
| INQUIR | <ALBUSE><br>{CINIT}<br>CNCN<br>DMSN<br><MIXIT><br>MXTBLE<br><PRTPLT><br><RDALB><br><RDARA><br><RDICE><br>RDOPT<br><RDRST><br><RDWTS> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <RT> | |
| | <RTARA> | |
| | <SAVST6> | |
| | <WAITIN> | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTICE> | |
| | <WRTPLT> | |
| | <WRTRST> | |
| IO | <ALBUSE> | |
| | {BOND} | |
| | {CLAPSE} | |
| | <RDALB> | |
| | <RDARA> | |
| | <RDBIAS> | |
| | <RDICE> | |
| | <RDPLOT> | |
| | <RDRST> | |
| | <RDSTRT> | |
| | <RDWTS> | |
| | <SAVST6> | |
| | {TWODXS} | |
| | {TWODYS} | |
| | <WAITIN> | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTARA> | |
| | <WRTICE> | |
| | <WRTPLT> | |
| | <WRTRST> | |
| | <WRTWTS> | |
| IOLEFT | CSAS | |
| | <IOSDUN> | |
| <IOSDUN> | CSAS | IOLEFT |
| | <JOMITY> | |
| | SASSY | |
| IOWRT | <DATAIN> | |
| | {SHORTX} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| IO4 | {CLAPSE} | |
| | {TWODXS} | |
| | {TWODYS} | |
| IREAD | <ARAYIN> | |
| | CNCN | |
| | CPARAM | |
| | DMSN | |
| | {FAX} | |
| | FIXCNC | |
| | FIXDIM | |
| | FIXPCH | |
| | <IDX1D> | |
| | <KENOG> | |
| | {LATCEL} | |
| | <MIXIT> | |
| | {MOREDT} | |
| | {MULTRG} | |
| | OPTCNC | |
| | OPTDMN | |
| | OPTPCH | |
| | <RDBIAS> | |
| | <RDBOX> | |
| | <RDPLOT> | |
| | <RDSTRT> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {STDCMP} | |
| | {XSMESH} | |
| | <XXIN> | |
| {ISCAN} | {IDENTS} | |
| | {NITAWL} | |
| | {REORDR} | |
| | {SLNS} | |
| | {STDCMP} | |
| <JOMCHK> | <JOMITY> | <CRMAX> |
| | | <HOLCHK> |
| | | <SRMAX> |
| | | <XXLIM> |
| <JOMITY> | SASSY | <ARALBA> |
| | | <CORSIZ> |
| | | <IOSDUN> |
| | | <JOMCHK> |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <LOADIT> |
| | | <PRTJOM> |
| | | <VOLUME> |
| {KAUL} | {DEAN} | |
| <KENOG> | <GEOMIN> | AREAD |
| | | IREAD |
| | | <RCHRS> |
| | | <RDORGN> |
| | | <XXIN> |
| | | XXINA |
| KI3 | {GDAN} | |
| {KNIGHT} | {C6DATA} | CLOSDA |
| | | MESAGE |
| | | OPENDA |
| | | REED |
| | | {SETUPA} |
| | | STOP |
| {LATCEL} | {SETUPB} | AREAD |
| | | FREAD |
| | | IREAD |
| | | LCOMPR |
| | | {MIXERR} |
| | | SCANOF |
| | | SCANON |
| {LCHKMX} | {BONAMI} | |
| | {CALCMS} | |
| | CELLMX | |
| | CMXCHK | |
| | MIXCHK | |
| | {XSDRN} | |
| LCOMPR | {CSPARM} | |
| | {LATCEL} | |
| | {MULTRG} | |
| | {RESDA} | |
| | {SETUPA} | |
| | {SETUPB} | |
| | {SMMTBA} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| LEN | <RCHRS> | |
| LIBEQL | CPARAM | |
| <LOADIT> | <JOMITY> | <BOXC><br><LODALB><br><LODARA><br><PRTARA><br>REED |
| <LOCATE> | <MESH> | <FINDBX><br><LOCBOX><br>MOVE<br><POSIT> |
| <LOCBOX> | <LOCATE><br><LODARA><br><SORTA> | |
| <LODALB> | <LOADIT> | REED |
| <LODARA> | <LOADIT> | CLEAR<br><HOLE><br><LOCBOX><br>REED |
| <LODRGC> | <RGUSED> | |
| <LODWTS> | SASSY | CLEAR<br><PRTWTS><br>RD<br>REED |
| {LOGID} | {SHORTX} | |
| LREAD | <MIXIT><br>{MULTRG}<br>{SETUPA}<br>{STDCMP} | |
| <LSCAN> | <ARASIZ><br><RGUSED> | |
| MESAGE | {KNIGHT} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <MESH> | <PRINT> | CLEAR<br><LOCATE> |
| MGCWRD | {AVGY}<br>{AVG2} | |
| {MIPCHK} | {C6DATA} | WOT8 |
| MIXCHK | MXTBLE | {LCHKMX} |
| {MIXERR} | {LATCEL}<br>{MOREDT}<br>{MULTRG} | |
| <MIXIT> | <DATAIN> | AREAD<br>FREAD<br>INQUIR<br>IREAD<br>LREAD<br><RDMIXT><br>RITE<br>STOP |
| {MIXTAB} | {SETUPB} | {ARRAY3}<br>{MODIFY}<br>{MTABLE}<br>{REORDR}<br>STOP |
| {MODIFY} | {MIXTAB} | |
| {MOREDT} | {SETUPB} | ALLOWC<br>AREAD<br>FREAD<br>IREAD<br>{MIXERR}<br>RESETC<br>SCANOF<br>SCANON |
| MOVE | <ADJUST><br><HUNTER><br><LOCATE><br>SASSY<br><SAVST6><br>{TWODYS} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | <VOLUME><br><WATES> | |
| {MTABLE} | {MIXTAB} | {IDENTS}<br>STOP |
| {MULTRG} | {SETUPB} | AREAD<br>FREAD<br>IREAD<br>LCOMPR<br>LREAD<br>{MIXERR}<br>SCANOF<br>SCANON |
| MXTBLE | SASSY | INQUIR<br>MIXCHK<br>RITE<br>STOP<br>TABLE |
| {NFLUX} | {PRELPS} | (EXP)<br>(SQRT) |
| {NITAWL} | {C6DATA} | {BINRCF}<br>{BINRCX}<br>CLEAR<br>{ISCAN} |
| {NOFRAC} | {IDENTS} | {WTINV} |
| OPENDA | {CINIT}<br>{KNIGHT}<br>{SHORTX} | |
| OPNFIL | CPARAM<br>CSAS<br><RDSTRT><br>{SHORTX} | |
| OPTCNC | OPTYP | AREAD<br>FREAD<br>IREAD |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| OPTDMN | OPTYP | AREAD<br>FREAD<br>IREAD |
| OPTPCH | OPTYP | AREAD<br>FREAD<br>IREAD<br>REED |
| OPTYP | RDOPT | AREAD<br>CNCN<br>CNCSRH<br>DIMSRH<br>DMSN<br>OPTCNC<br>OPTDMN<br>OPTPCH<br>PCHSRH<br>REED<br>STOP |
| {OUTER} | {SPHDAN} | (ACOS)<br>(ASIN)<br>(COS)<br>(SIN)<br>(SQRT)<br>{FINNER}<br>{GAUSS} |
| PARTBL | CSAS | RNDOUT |
| {PATCH} | {TWODXS}<br>{TWODYS} | |
| PCHSRH | FIXTYP<br>OPTYP | PRTSRH<br>RITE<br>STOP<br><XXLIM> |
| <POINT> | SASSY | |
| {POKE} | {SETUPB} | (SQRT) |
| <POSIT> | <LOCATE> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| PRATBL | SASSY | <RTADJ> |
| {PRELPS} | {C6DATA} | {GFLUX}<br>{NFLUX} |
| <PRINT> | <PRTPLT> | <MESH> |
| <PRTARA> | <LOADIT><br>SASSY | |
| <PRTJOM> | <JOMITY><br>SASSY | |
| <PRTLBA> | <ARALBA> | |
| <PRTPLT> | SASSY | CLEAR<br>INQUIR<br><PRINT><br>REED<br><RELATE><br><UNTCRS> |
| PRTSRH | CNCSRH<br>DIMSRH<br>PCHSRH | |
| <PRTWTS> | <LODWTS> | |
| <RCHRS> | <ARAYIN><br><KENOG><br><RDPLOT> | AREAD<br>GETPTR<br>LEN<br>RCRDLN<br>RSTPTR<br>STOP |
| RCRDLN | <RCHRS> | |
| RD | <LODWTS> | |
| <RDALB> | <RDRST> | CLEAR<br>INQUIR<br>IO<br>RITE |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <RDARA> | <RDRST> | INQUIR<br>IO<br>RITE |
| <RDBIAS> | <DATAIN> | AREAD<br>FREAD<br>IO<br>IREAD<br><WAITIN> |
| <RDBOX> | <ARAYIN> | IREAD |
| <RDICE> | <RDRST> | CLEAR<br>INQUIR<br>IO<br>RITE |
| <RDMIXT> | <MIXIT> | |
| RDOPT | CSAS | AREAD<br>CLEAR<br>FIXTYP<br>INQUIR<br>OPTYP<br>RITE |
| <RDORGN> | <KENOG> | AREAD<br>FREAD |
| <RDPLOT> | <DATAIN> | (SQRT)<br>AREAD<br>CLEAR<br>FREAD<br>IO<br>IREAD<br><RCHRS> |
| <RDREF> | <DATAIN> | AREAD<br>CREAD |
| <RDRST> | <DATAIN> | INQUIR<br>IO<br><RDALB><br><RDARA><br><RDICE> |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | <RDWTS> |
| | | RITE |
| <RDSTRT> | <DATAIN> | AREAD |
| | | CLEAR |
| | | FREAD |
| | | IO |
| | | IREAD |
| | | OPNFIL |
| <RDWTS> | <RDRST> | INQUIR |
| | | IO |
| | | RITE |
| <READGM> | <GEOMIN> | |
| REED | FIXPCH | |
| | FIXTYP | |
| | <FLDATA> | |
| | {KNIGHT} | |
| | <LOADIT> | |
| | <LODALB> | |
| | <LODARA> | |
| | <LODWTS> | |
| | OPTPCH | |
| | OPTYP | |
| | <PRTPLT> | |
| | {SHORTX} | |
| | <SORTA> | |
| | {STDCMP} | |
| | <WATES> | |
| | <WRTALB> | |
| | <WRTARA> | |
| | <WRTICE> | |
| | <WRTRST> | |
| | <WRTWTS> | |
| <RELATE> | <PRTPLT> | CLEAR |
| {REORDR} | {MIXTAB} | {ISCAN} |
| {RESDA} | {SETUPB} | {BONDA} |
| | | CLEAR |
| | | {DANCOF} |
| | | LCOMPR |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| RESETC | {MOREDT} | |
| RESETM | CSAS | |
| <RGUSED> | <FLDATA> | CLEAR<br><LODRGC><br><LSCAN> |
| RITE | <ALBUSE><br>{CINIT}<br>CNCN<br>CNCSRH<br>DIMSRH<br>DMSN<br><FLDATA><br><MIXIT><br>MXTBLE<br>PCHSRH<br><RDALB><br><RDARA><br><RDICE><br>RDOPT<br><RDRST><br><RDWTS><br><RT><br><RTARA><br><SAVST6><br><WAITIN><br><WATES><br><WRTPLT> | |
| RNDIN | CPARAM | |
| RNDOUT | CPARAM<br>PARTBL<br><WRTRST> | |
| RSTPTR | <RCHRS> | |
| <RT> | <DATAIN> | INQUIR<br>RITE |
| <RTADJ> | PRATBL | INDEX |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <RTARA> | <DATAIN> | CLEAR<br>INQUIR<br>RITE |
| SASSY | CSAS | <ARALBA><br>CMXCHK<br><DATAIN><br><IOSDUN><br><JOMITY><br><LODWTS><br>MOVE<br>MXTBLE<br><POINT><br>PRATBL<br><PRTARA><br><PRTJOM><br><PRTPLT><br>STOP<br><WRTRST> |
| <SAVST6> | <DATAIN> | CLEAR<br>INQUIR<br>IO<br>MOVE<br>RITE<br>STOP |
| SCANOF | {LATCEL}<br>{MOREDT}<br>{MULTRG}<br>{STDCMP} | |
| SCANON | CSAS<br>{LATCEL}<br>{MOREDT}<br>{MULTRG}<br>{SETUPA} | |
| {SDAN} | {DANDAD} | E3 |
| {SETUPA} | {KNIGHT} | AREAD<br>{CSPARM}<br>FREAD<br>IREAD<br>LCOMPR<br>LREAD |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | | SCANON<br>{SETUPB}<br>STOP |
| {SETUPB} | {SETUPA} | AREAD<br>CLEAR<br>IREAD<br>{LATCEL}<br>LCOMPR<br>{MIXTAB}<br>{MOREDT}<br>{MULTRG}<br>{POKE}<br>{RESDA}<br>{STDCMP}<br>STOP |
| SHELC | {DANCOF} | |
| {SHORTX} | {C6DATA} | CLOSDA<br>COPY<br>DTASET<br>IOWRT<br>{LOGID}<br>OPENDA<br>OPNFIL<br>REED |
| {SLNAME} | {STDCMP} | |
| {SLNS} | {STDCMP} | {ISCAN} |
| {SMMTBA} | {CSPARM}<br>{C6DATA} | LCOMPR |
| <SORTA> | <FLDATA> | CLEAR<br><HOLE><br><LOCBOX><br>REED<br>STOP |
| <SORTR> | <FLDATA> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| {SPHDAN} | {BALL} | (ACOS) |
| | | {AGAUSS} |
| | | {OUTER} |
| <SRMAX> | <HOLEXT> | (SQRT) |
| | <JOMCHK> | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |
| | | <VECADD> |
| | | <VECNRM> |
| <SRMIN> | <HOLHOL> | (SQRT) |
| | | <CRSPRD> |
| | | <DOTPRD> |
| | | STOP |
| | | <VECADD> |
| | | <VECNRM> |
| {STDCMP} | {SETUPB} | AREAD |
| | | {FAX} |
| | | {FAXPRT} |
| | | {FIX} |
| | | {FOX} |
| | | FREAD |
| | | IREAD |
| | | {ISCAN} |
| | | LREAD |
| | | REED |
| | | SCANOF |
| | | {SLNAME} |
| | | {SLNS} |
| STOP | <ARAYIN> | |
| | {AVGY} | |
| | {CLAPSE} | |
| | CNCSRH | |
| | <CRMAX> | |
| | <CRMIN> | |
| | CSAS | |
| | {C6DATA} | |
| | {DANCOF} | |
| | <DATAIN> | |
| | DIMSRH | |
| | FIXTYP | |
| | <FLDATA> | |
| | <GEOMIN> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| | {HOGGIE} | |
| | <HOLE> | |
| | {KNIGHT} | |
| | <MIXIT> | |
| | {MIXTAB} | |
| | {MTABLE} | |
| | MXTBLE | |
| | OPTYP | |
| | PCHSRH | |
| | <RCHRS> | |
| | SASSY | |
| | <SAVST6> | |
| | {SETUPA} | |
| | {SETUPB} | |
| | <SORTA> | |
| | <SRMAX> | |
| | <SRMIN> | |
| | {TWODYS} | |
| | <WATES> | |
| | {XSDRN} | |
| | {XSMESH} | |
| TABLE | MXTBLE | |
| TIMFAC | CPARAM | |
| {TWODXS} | {CLAPSE} | {AVG2} |
| | | CLEAR |
| | | COPY |
| | | IO |
| | | IO4 |
| | | {PATCH} |
| {TWODYS} | {CLAPSE} | {AVGY} |
| | | CLEAR |
| | | COPY |
| | | {HOGGIE} |
| | | IO |
| | | IO4 |
| | | MOVE |
| | | {PATCH} |
| | | STOP |
| {UNFOLD} | {BOND} | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <UNTCRS> | <PRTPLT> | CLEAR |
| <VECADD> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN><br><XRMIN> | |
| <VECNRM> | <CRMAX><br><CRMIN><br><SRMAX><br><SRMIN> | |
| {VERIFY} | {C6DATA} | {VOLJB} |
| {VOLFRC} | {ICE} | |
| {VOLJB} | {VERIFY} | |
| <VOLUME> | <JOMITY> | (ASIN)<br>(SQRT)<br>CLEAR<br><GOCURS><br><GTVOLS><br><HUNTER> |
| MOVE | | |
| <WAITIN> | <RDBIAS> | INQUIR<br>IO<br>RITE |
| <WATES> | <FLDATA> | INQUIR<br>IO<br>MOVE<br>REED<br>RITE<br>STOP |
| WOT8 | {MIPCHK} | |
| <WRTALB> | <WRTRST> | INQUIR<br>IO<br>REED |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <WRTARA> | <WRTRST> | IO<br>REED |
| <WRTICE> | <WRTRST> | INQUIR<br>IO<br>REED |
| <WRTPLT> | <DATAIN> | INQUIR<br>IO<br>RITE |
| <WRTRST> | SASSY | INQUIR<br>IO<br>REED<br>RNDOUT<br><WRTALB><br><WRTARA><br><WRTICE><br><WRTWTS> |
| <WRTWTS> | <WRTRST> | IO<br>REED |
| {WTINV} | {NOFRAC} | |
| <XRMIN> | <HOLHOL> | (SQRT)<br><DOTPRD><br><VECADD> |
| {XSDRN} | {C6DATA} | {BINFIL}<br>{BINRCF}<br>{BINRCX}<br>CLEAR<br>{LCHKMX}<br>STOP |
| {XSMESH} | {C6DATA} | IREAD<br>STOP |
| <XXIN> | <KENOG> | FREAD<br>IREAD |
| XXINA | <KENOG> | |

Table C6.A.1 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| <XXLIM> | <HOLEXT><br><JOMCHK><br>PCHSRH | (SQRT) |
| <XXMIN> | <HOLHOL> | (SQRT) |
| YREAD | <ARAYIN> | |
| ZREAD | CPARAM | |

Table C6.A.2 Alphabetical index of MODIFY routines

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| (ACOS) | FIXEDK<br>RCUBIC | |
| ALOCAT | MODIFY | |
| ALTER | LODATA | CLEAR<br>XXLIM |
| AVGKEF | LODATA | (SQRT) |
| CLEAR | ALTER | |
| CLOSDA | MODIFY | |
| (COS) | FIXEDK<br>RCUBIC | |
| CUBFIT | FIXEDK<br>MAXMIN<br>OPTMIZ | DECOMP<br>SOLVE |
| DECOMP | CUBFIT | |
| DENSTY | LODATA | |
| (DSQRT) | OPTMIZ | |
| FIXEDK | OPTMIZ | (ACOS)<br>(COS)<br>(SQRT)<br>CUBFIT |
| GETMS | MODIFY | |
| INQUIR | WRTALB | |
| IO | RERITE<br>WRTALB<br>WRTARA<br>WRTWTS | |
| IOLEFT | MODIFY | |
| ISCAN | MAXMIN | |

Table C6.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| LISTQA | MODIFY | |
| LODATA | LODCOM | ALTER<br>AVGKEF<br>DENSTY<br>MOVE<br>OPTMIZ<br>PRTJOM<br>PRTMIX<br>PRTSRH<br>REED<br>RERITE<br>RESTRT<br>RITE<br>STOP |
| LODCOM | MODIFY | LODATA<br>MOVE |
| MAXMIN | OPTMIZ | (SQRT)<br>CUBFIT<br>ISCAN |
| MODBLK | | |
| MODIFY | | ALOCAT<br>CLOSDA<br>GETMS<br>IOLEFT<br>LISTQA<br>LODCOM<br>OPENDA<br>OPNFIL<br>REED<br>RESETM<br>RITE<br>STOP |
| MOVE | LODATA<br>LODCOM | |
| OPENDA | MODIFY | |
| OPNFIL | MODIFY | |

Table C6.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| OPTMIZ | LODATA | (SQRT)<br>CUBFIT<br>DSQRT<br>FIXEDK<br>MAXMIN<br>RCUBIC |
| PRTJOM | LODATA | |
| PRTMIX | LODATA | |
| PRTSRH | LODATA | |
| RCUBIC | OPTMIZ | (ACOS)<br>(COS)<br>(SQRT) |
| REED | LODATA<br>MODIFY<br>RERITE<br>WRTALB<br>WRTARA<br>WRTWTS | |
| RERITE | LODATA | IO<br>REED<br>WRTALB<br>WRTARA<br>WRTWTS |
| RESETM | MODIFY | |
| RESTRT | LODATA | |
| RITE | LODATA<br>MODIFY | |
| SOLVE | CUBFIT | |
| (SQRT) | AVGKEF<br>FIXEDK<br>MAXMIN<br>OPTMIZ<br>RCUBIC<br>XXLIM | |

Table C6.A.2 (continued)

| SUBROUTINE NAME | CALLING SUBROUTINE | CALLED SUBROUTINE |
|---|---|---|
| STOP | LODATA<br>MODIFY | |
| WRTALB | RERITE | INQUIR<br>IO<br>REED |
| WRTARA | RERITE | IO<br>REED |
| WRTWTS | RERITE | IO<br>REED |
| XXLIM | ALTER | (SQRT) |

Table C6.A.3 Commons used in CSAS6

| COMMON NAME | SUBROUTINE |
|---|---|
| ALBDAT | ALBRD |
| | ALBUSE |
| | CPARAM |
| | DATAIN |
| | DIFALB |
| | FLDATA |
| | LOADIT |
| | PARTBL |
| | POINT |
| | PRATBL |
| | RDALB |
| | RDREF |
| | RDRST |
| | SASSY |
| | WRTALB |
| | WRTRST |
| ALBNAM | ALBRD |
| | ALBUSE |
| | CPARAM |
| | DATAIN |
| | DIFALB |
| | FLDATA |
| | LOADIT |
| | PARTBL |
| | POINT |
| | PRATBL |
| | RDALB |
| | RDREF |
| | RDRST |
| | SASSY |
| | WRTALB |
| | WRTRST |
| BLKLNC | CINIT |
| | CPARAM |
| | MOREDT |
| | PARTBL |
| | SETUPB |
| | STDCMP |
| CCDATA | CSAS |
| | C6DATA |
| | SASSY |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| CDATA | CSAS |
| | C6DATA |
| | SASSY |
| CELMIX | CELLMX |
| | CMXCHK |
| CLAD | DANDAD |
| | GDAN |
| | SDAN |
| COMPOZ | KNIGHT |
| | MTABLE |
| CONMOD | KNIGHT |
| CUNIT | ARCHK |
| | CSAS |
| | CSPARM |
| | FAXPRT |
| | IDENTS |
| | KNIGHT |
| | LATCEL |
| | MIXTAB |
| | MOREDT |
| | MTABLE |
| | MULTRG |
| | NOFRAC |
| | POKE |
| | REORDR |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | SLNAME |
| | SLNS |
| | STDCMP |
| DEAL | C6DATA |
| | DANCOF |
| | LATCEL |
| | MULTRG |
| | POKE |
| | RESDA |
| | SETUPB |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| DIMEN | ARASIZ |
| | ARAYIN |
| | BOX |
| | BOXC |
| | CNCN |
| | CORSIZ |
| | CPARAM |
| | C6DATA |
| | DATAIN |
| | DMSN |
| | FINDBX |
| | FIXTYP |
| | FLDATA |
| | GEOMIN |
| | HOLCHK |
| | JOMCHK |
| | JOMITY |
| | KENOG |
| | LOCATE |
| | LOCBOX |
| | LODARA |
| | MESH |
| | MIXIT |
| | MXTBLE |
| | OPTYP |
| | PARTBL |
| | POINT |
| | POSIT |
| | PRATBL |
| | PRINT |
| | PRTJOM |
| | PRTPLT |
| | RDBIAS |
| | RDBOX |
| | RDICE |
| | RDREF |
| | RDRST |
| | RDSTRT |
| | READGM |
| | RGUSED |
| | RTARA |
| | SASSY |
| | SORTA |
| | SORTR |
| | VOLUME |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | WRTICE |
| | WRTRST |
| DRTACS | ALBUSE |
| | CINIT |
| | CNCN |
| | CNCSRH |
| | CSAS |
| | DATAIN |
| | DIMSRH |
| | DMSN |
| | FIXPCH |
| | FIXTYP |
| | FLDATA |
| | LOADIT |
| | LODARA |
| | LODWTS |
| | MIXIT |
| | MXTBLE |
| | OPTPCH |
| | OPTYP |
| | PCHSRH |
| | PRTPLT |
| | RDALB |
| | RDARA |
| | RDICE |
| | RDOPT |
| | RDRST |
| | RDWTS |
| | RT |
| | RTARA |
| | SASSY |
| | SAVST6 |
| | SHORTX |
| | SORTA |
| | WAITIN |
| | WATES |
| | WRTALB |
| | WRTICE |
| | WRTPLT |
| | WRTRST |
| | WRTWTS |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| ERRFLG | ARAYIN |
| | PRTLBA |
| | SASSY |
| FLG | ARCHK |
| | CNCSRH |
| | CSAS |
| | C6DATA |
| | DANCOF |
| | DIMSRH |
| | FAX |
| | FIXCNC |
| | FIXDIM |
| | FIXPCH |
| | FIXTYP |
| | LATCEL |
| | MOREDT |
| | MULTRG |
| | OPTCNC |
| | OPTDMN |
| | OPTPCH |
| | OPTYP |
| | PCHSRH |
| | RDOPT |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | SLNS |
| | STDCMP |
| | VERIFY |
| | XSMESH |
| GIN | ADAN |
| | BALL |
| | DAN |
| | DANDAD |
| | FINNER |
| | GDAN |
| | OUTER |
| | SPHDAN |
| ICECOM | ICE |
| | MOREDT |
| | SETUPB |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| IOS | CSAS |
| | SASSY |
| LIBA | C6DATA |
| | EPSIG |
| | FIX |
| | FOX |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MULTRG |
| | RESDA |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | STDCMP |
| LIBC | C6DATA |
| | EPSIG |
| | FIX |
| | FOX |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MULTRG |
| | RESDA |
| | SETUPA |
| | SETUPB |
| | SHORTX |
| | STDCMP |
| LOGIC | ALBRD |
| | ARASIZ |
| | ARAYIN |
| | CORSIZ |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | FLDATA |
| | IDX1D |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | JOMCHK |
| | JOMITY |
| | KENOG |
| | LOCATE |
| | MESH |
| | PARTBL |
| | POINT |
| | POSIT |
| | PRTJOM |
| | PRTPLT |
| | RDBIAS |
| | RDBOX |
| | RDREF |
| | RDRST |
| | READGM |
| | RTARA |
| | SASSY |
| | SORTA |
| | SORTR |
| | VOLUME |
| | WATES |
| | WRTRST |
| LOWBND | CORSIZ |
| | GOCURS |
| | HUNTER |
| | JOMCHK |
| | LOCATE |
| | LODARA |
| | MESH |
| | POINT |
| | PRINT |
| | PRTJOM |
| | PRTPLT |
| | VOLUME |
| LPNT | CINIT |
| | RDOPT |
| | SASSY |
| MATRX | FLDATA |
| | PRATBL |
| | SORTA |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| NUTRON | FINDBX |
| | FLDATA |
| | LOCATE |
| | MESH |
| PARM | C6DATA |
| | MOREDT |
| | SETUPA |
| | SETUPB |
| PASS | C6DATA |
| | DANCOF |
| | KNIGHT |
| | LATCEL |
| | MIPCHK |
| | MIXTAB |
| | MODIFY |
| | MOREDT |
| | MULTRG |
| | POKE |
| | RESDA |
| | SASSY |
| | SETUPA |
| | SETUPB |
| | VERIFY |
| | XSDRN |
| PASS2 | C6DATA |
| | KNIGHT |
| | MIXTAB |
| | MOREDT |
| | SETUPA |
| | SETUPB |
| | XSDRN |
| PICT | MESH |
| | PRINT |
| | PRTPLT |
| | RDPLOT |
| | RELATE |
| PICTTL | MESH |
| | PRINT |
| | PRTPLT |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | RDPLOT |
| | RELATE |
| POINTA | C6DATA |
| | SETUPB |
| POINTB | C6DATA |
| | SETUPA |
| POINTC | C6DATA |
| | SETUPA |
| POINTR | DATAIN |
| | JOMITY |
| | LOADIT |
| | POINT |
| | SASSY |
| REC1 | CNCN |
| | CSAS |
| | DMSN |
| | FIXPCH |
| | FIXTYP |
| | OPTPCH |
| | OPTYP |
| | PRTSRH |
| | RDOPT |
| | SASSY |
| RUNTYP | BIRITE |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | PARTBL |
| STDATA | DATAIN |
| | RDRST |
| | RDSTRT |
| | SASSY |
| | WRTRST |
| TITL | ARASIZ |
| | CORSIZ |
| | CPARAM |
| | DATAIN |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | GEOMIN |
| | HOLCHK |
| | HOLEXT |
| | HOLHOL |
| | JOMCHK |
| | KENOG |
| | PARTBL |
| | POINT |
| | PRATBL |
| | PRTJOM |
| | PRTLBA |
| | PRTPLT |
| | PRTSRH |
| | PRTWTS |
| | RDBOX |
| | RDPLOT |
| | READGM |
| | VOLUME |
| | WRTRST |
| UNIT | ALBRD |
| | ALBUSE |
| | ARASIZ |
| | ARAYIN |
| | BOX |
| | CINIT |
| | CNCN |
| | CNCSRH |
| | CORSIZ |
| | CPARAM |
| | CSAS |
| | DATAIN |
| | DIFALB |
| | DIMSRH |
| | DMSN |
| | FIXCNC |
| | FIXDIM |
| | FIXPCH |
| | FIXTYP |
| | FLDATA |
| | GEOMIN |
| | ICE |
| | IDX1D |
| | JOMCHK |
| | JOMITY |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| | KENOG |
| | LOADIT |
| | LODARA |
| | LODWTS |
| | MIXIT |
| | MXTBLE |
| | OPTCNC |
| | OPTDMN |
| | OPTPCH |
| | OPTYP |
| | PARTBL |
| | PCHSRH |
| | POINT |
| | POSIT |
| | PRINT |
| | PRTARA |
| | PRTJOM |
| | PRTLBA |
| | PRTPLT |
| | PRTSRH |
| | RDALB |
| | RDBIAS |
| | RDBOX |
| | RDICE |
| | RDOPT |
| | RDPLOT |
| | RDREF |
| | RDRST |
| | RDSTRT |
| | RDWTS |
| | READGM |
| | RT |
| | RTARA |
| | SASSY |
| | SAVST6 |
| | SHORTX |
| | SORTA |
| | SORTR |
| | VOLUME |
| | WAITIN |
| | WATES |
| | WRTALB |
| | WRTICE |
| | WRTRST |
| | WRTWTS |

Table C6.A.3 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| UNITS | CSAS |
| | C6DATA |
| | SASSY |
| XSDRNC | MOREDT |
| | SETUPB |
| | XSDRN |

Table C6.A.4 Commons used in MODIFY

| COMMON NAME | SUBROUTINE |
|---|---|
| ALBDAT | LODCOM |
| | RERITE |
| | RESTRT |
| | WRTALB |
| ALBNAM | LODCOM |
| | RERITE |
| | RESTRT |
| | WRTALB |
| CCDATA | MODIFY |
| CDATA | MODIFY |
| CYCLE | LODATA |
| | LODCOM |
| | MODIFY |
| DIMEN | LODATA |
| | LODCOM |
| | PRTJOM |
| | RERITE |
| | RESTRT |
| DRTACS | LODATA |
| | LODCOM |
| | MODIFY |
| | RERITE |
| | RESTRT |
| | WRTALB |
| | WRTWTS |
| LOGIC | PRTJOM |
| | RERITE |
| | RESTRT |
| LOWBND | PRTJOM |
| LPNT | LODATA |
| | MODIFY |
| REC1 | LODATA |
| | LODCOM |
| | MODIFY |
| | PRTSRH |
| | RERITE |

Table C6.A.4 (continued)

| COMMON NAME | SUBROUTINE |
|---|---|
| STDATA | LODCOM |
| | RERITE |
| | RESTRT |
| TITL | LODATA |
| | PRTJOM |
| | PRTMIX |
| | PRTSRH |
| | RESTRT |
| UNIT | LODATA |
| | LODCOM |
| | MODIFY |
| | PRTJOM |
| | PRTSRH |
| | RERITE |
| | WRTALB |
| | WRTWTS |
| UNITS | MODIFY |

## C6.B SAMPLE PROBLEMS

This section contains sample problems to demonstrate some of the options available in CSAS and its associated sequences. A brief problem description and the associated card input data are included for each problem. See Sect. C6.5 for additional examples.

### SAMPLE PROBLEM 1 ALUMINUM 30 DEGREE PIPE ANGLE INTERSECTION

The purpose of this problem is to calculate the k-effective of a system composed of intersecting aluminum pipes, in the shape of a Y, filled with a 5% enriched $UO_2F_2$ solution. The $UO_2F_2$ solution at 299°K contains 907.0 gm/l of uranium, no excess acid, and has a specific gravity of 2.0289 gm/cm$^3$. The assembly is composed of a 212.1 cm long vertical pipe and a second pipe that intersects the vertical pipe 76.7 cm from the outside bottom at an angle of 29.26 degrees with the upper vertical pipe. Both pipes have 13.95 cm inner diameters and 14.11 cm outer diameters. The vertical pipe is open on the top and 1.3 cm thick on the bottom. The Y-leg pipe, in the YZ-plane, is 126.04 cm in length with the sealed end 0.64 cm thick. The assembly is filled with solution to a height 129.5 cm above the outside bottom of the vertical pipe. From the point where the pipes intersect the assembly is surrounded by water 37.0 cm in the ±X directions, 100 cm in the +Y direction, -37 cm in the -Y direction, to the top of the assembly in the +Z direction, and -99.6 cm in the -Z direction.

```
=CSAS26
KENO-VI 30 DEG Y 5%UO2F2 907.0G/L 128.2 SOLN HEIGHT K=1.001 +- .004
27GROUPNDF4 INFHOMMEDIUM
SOLNUO2F2 1 907.0 0.0 SPG=2.0289 1.0 299 92235 5.0 92238 95.0 END
AL      2 1.0 END
H2O     3 1.0 END
END COMP
READ PARAMETERS
 GEN=103 NPG=300
 FLX=YES FDN=YES FAR=YES
 PGM=YES  TBA=1.0 RUN=YES PLT=YES
  END PARAMETERS
READ START NST=6 TFX=0.1 TFY=0.1 TFZ=0.1 LNU=300
END START
READ GEOMETRY
GLOBAL
UNIT 1
COM='30 DEG Y'
CYLINDER 10  13.95  135.4 -75.4
CYLINDER 20  14.11  135.4 -76.7
CYLINDER 30  13.95  125.4   0.0  ROTATE A2=-29.26
CYLINDER 40  14.11  126.04  0.0  ROTATE A2=-29.26
CUBOID  50  2P37.0 100. -37.0 52.8 -75.4
CUBOID  60  2P37.0 100. -37.0 135.4 -99.6
MEDIA  1  1 10  50
MEDIA  2  1 20 -10 -30
MEDIA  1  1 30  50 -10
MEDIA  2  1 40 -30 -20
```

```
MEDIA 0 1 10 -50
MEDIA 0 1 30 -50 -10
MEDIA 3 1 60 -20 -40 -10
BOUNDARY 60
END GEOMETRY
READ PLOT
TTL='Y-Z SLICE AT X=0.0 THROUGH CENTERLINE OF BOTH PIPES'
XUL=0.0 YUL=-39.0 ZUL=137.0
XLR=0.0 YLR=105.0 ZLR=-105.0
VAX=1 WDN=-1
NAX=130 NCH=' *A.' LPI=10 END PLT1
TTL='X-Z SLICE AT Y=0.0 SHOWS ONLY VERTICAL COLUMN'
XUL=-40.0 YUL=0.0 ZUL=137.0
XLR=+40.0 YLR=0.0 ZLR=-105.0
UAX=1 WDN=-1 NAX=130 END PLT2
TTL='X-Y SLICE AT Z=-50.0 VERTICAL COLUMN BELOW JUNCTION'
XUL=-40.0 YUL=105.0 ZUL=-50.0
XLR=+40.0 YLR=-40.0 ZLR=-50.0
UAX=1 VDN=-1 NAX=130 END PLT3
TTL='X-Y SLICE AT Z=0.0 THROUGH JUNCTION'
XUL=-40.0 YUL=105.0 ZUL=0.0
XLR=+40.0 YLR=-40.0 ZLR=0.0
UAX=1 VDN=-1 NAX=130 END PLT4
TTL='X-Y SLICE AT Z=24.0 SLIGHTLY BELOW POINT OF SEPARATION'
XUL=-40.0 YUL=105.0 ZUL=24.0
XLR=+40.0 YLR=-40.0 ZLR=24.0
UAX=1 VDN=-1 NAX=130 END PLT5
TTL='X-Y SLICE AT Z=26.0 SLIGHTLY ABOVE POINT OF SEPARATION'
XUL=-40.0 YUL=105.0 ZUL=26.0
XLR=+40.0 YLR=-40.0 ZLR=26.0
UAX=1 VDN=-1 NAX=130 END PLT6
TTL='X-Y SLICE AT Z=75.0 WELL ABOVE POINT OF SEPARATION'
XUL=-40.0 YUL=105.0 ZUL=75.0
XLR=+40.0 YLR=-40.0 ZLR=75.0
UAX=1 VDN=-1 NAX=130 END PLT7
TTL='Y-Z SLICE AT X=0.0 ENLARGED VIEW 5CM EITHER SIDE OF CENTRAL COLUMN'
XUL=0.0 YUL=9.11 ZUL=65.0
XLR=0.0 YLR=19.11 ZLR=-10.0
VAX=1 WDN=-1 NAX=130 END PLT8
END PLOT
END DATA
END
```

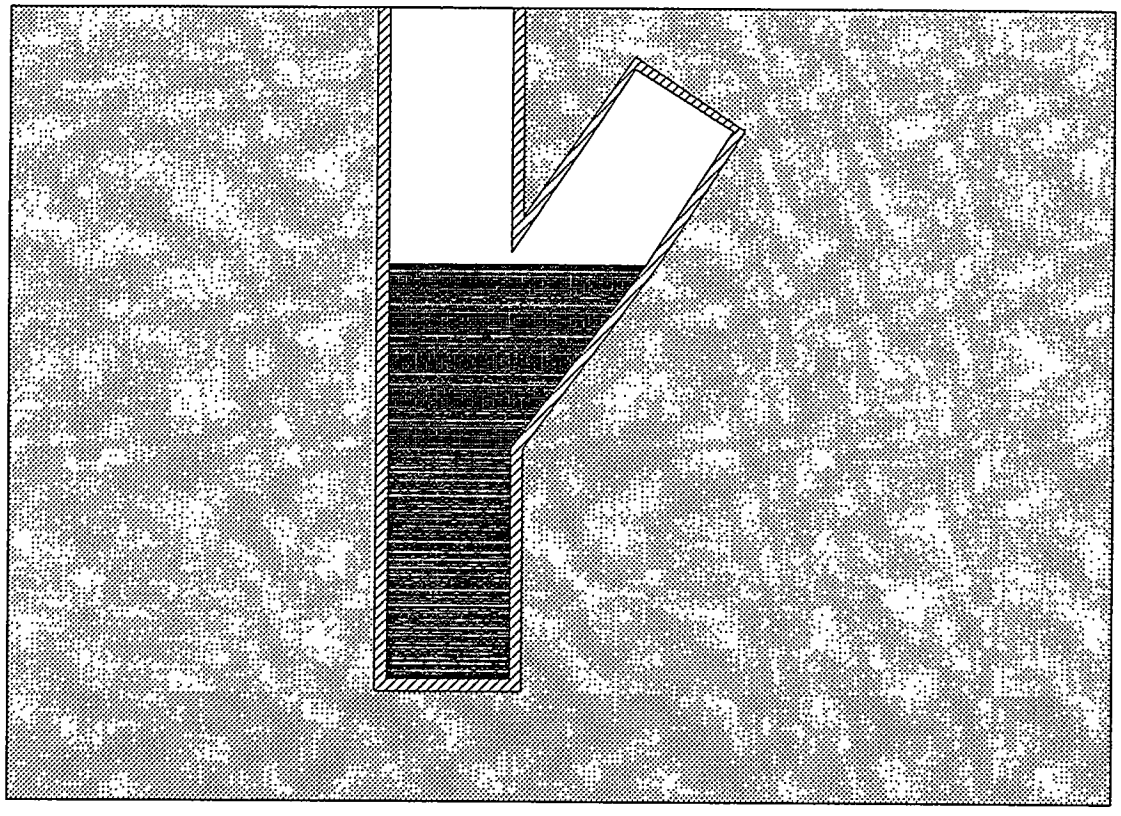


Figure C6.B.1 Critical assembly of $UO_2F_2$ solution in a 30°-Y aluminum pipe

## SAMPLE PROBLEM 2 PLEXIGLASS CROSS

The purpose of this problem is to calculate the k-effective of a system composed of intersecting plexiglass pipes, in the shape of a cross, filled with a 5% enriched $UO_2F_2$ solution. The room temperature $UO_2F_2$ solution contains 896.1 gm/l of uranium, no excess acid, and has a specific gravity of 2.015 gm/cm$^3$. The pipes have a 13.335 cm inner diameter and 16.19 cm outer diameter. The vertical pipe is open on the top and 3.17 cm thick on the bottom. The horizontal pipe ends are 3.17 thick. The vertical pipe is 210.19 cm in length and filled with solution to a height of 117.2 cm. The 2 horizontal legs, positioned in the XZ-plane, intersect the vertical pipe 91.44 cm from the outside bottom at an 89 degree angle with the upper section of the pipe. Each horizontal is 91.44 cm in length and filled with the above specified $UO_2F_2$ solution. A water reflector surrounding the solution filled pipes extends out from the point where the pipes intersect 111.76 cm in the ±X directions, 20.64 cm in the ±Y directions, 29.03 cm in the +Z direction, and -118.428 cm in the -Z direction.

```
=CSAS26
KENOVI 89 DEG CROSS 5% UO2F2 SOLN, PLEXIGLASS PIPES, H2O REFL.
27GROUPNDF4  INFHOMMEDIUM
SOLNUO2F2  1 896.1 0.0 SPG=2.0150 1.0 298 92235 5.0 92238 95.0 END
PLEXIGLASS 3 1.0 END
H2O      2 1.0 END
END COMP
READ PARAM
  GEN=103 NPG=300 RUN=YES PLT=YES
  END PARAM
READ GEOM
GLOBAL UNIT 1
CYLINDER  10  13.335 28.93 -88.27
CYLINDER  20  13.335 121.92 -88.27
CYLINDER  30  16.19 121.92 -91.44
CYLINDER  40  13.335 88.27 0.0 ROTATE A1=90. A2=89.
CYLINDER  50  16.19  91.44 0.0 ROTATE A1=90. A2=89.
CYLINDER  60  13.335 88.27 0.0 ROTATE A1=-90. A2=89.
CYLINDER  70  16.19  91.44 0.0 ROTATE A1=-90. A2=89.
CUBOID    80  2P111.74 2P20.64 29.03 -118.428
CUBOID    90  2P111.74 2P40.64 121.92 -118.428
MEDIA 1 1 10
MEDIA 0 1 20 -10
MEDIA 3 1 30 -10 -20 -50 -70
MEDIA 1 1 40 -10 -20
MEDIA 3 1 50 -40 -10 -20
MEDIA 1 1 60 -10 -20
MEDIA 3 1 70 -60 -10 -20 -50
MEDIA 2 1 80 -10 -20 -30 -40 -50 -60 -70
MEDIA 0 1 90 -20 -30 -80
BOUNDARY  90
END GEOM
READ START NST=6  TFX=0. TFY=0. TFZ=0. LNU=300 END START
READ PLOT
```

```
TTL=' X-Z SLICE AT Y=0.0 '
XUL=-113. YUL=0. ZUL=48. XLR=113. YLR=0. ZLR=-120.
NCH=' 1/*'
UAX=1 WDN=-1 NAX=130 END PLT1
TTL=' Y-Z SLCE AT X=0.0 '
XUL=0. YUL=-42. ZUL=122. XLR=0. YLR=42. ZLR=-120.
VAX=1.0 WDN=-1.0 NAX=130  END PLT2
TTL=' X-Y SLICE AT Z=0.0 '
XUL=-113.0 YUL=42. ZUL=0. XLR=113.0 YLR=-42. ZLR=0.
UAX=1.0 VDN=-1.0 NAX=130  END PLT3
END PLOT
END DATA
END
```

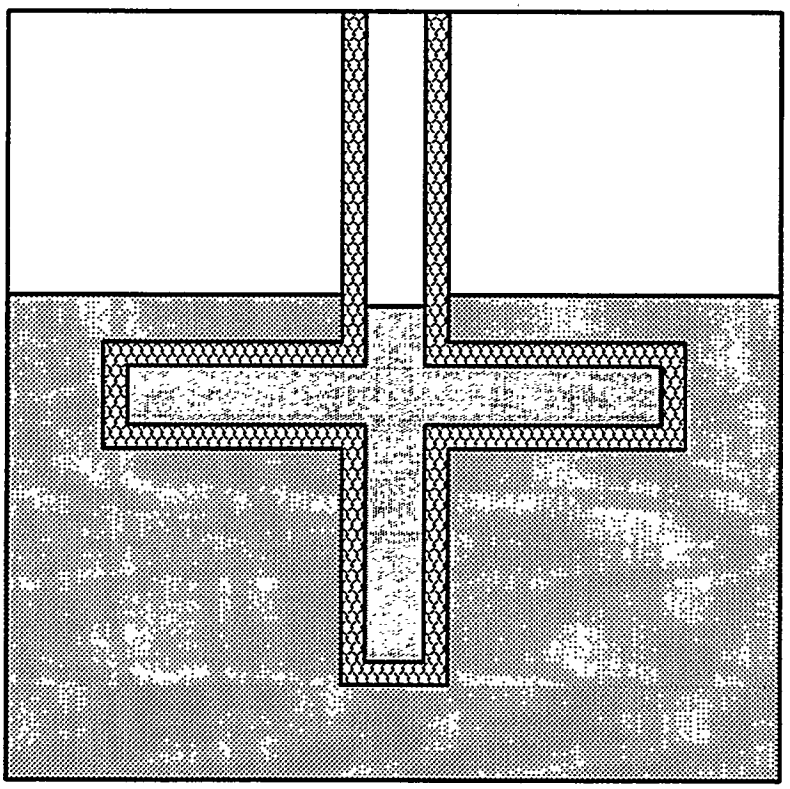


Figure C6.B.2 Critical assembly of $UO_2F_2$ solution in a plexiglass cross

## SAMPLE PROBLEMS 3 - 6 SPHERE MODELS USING CHORDS AND MIRROR ALBEDOS

This set of 4 problems models an assembly consisting of a 93.2 % enriched bare uranium sphere, 8.714 cm in radius, having a density of 18.76 gm/cm$^3$. The first problem models the assembly as a single bare sphere. The second problem models the assembly as a hemisphere with mirror reflection on the flat surface. The third problem models the assembly as a quarter sphere with mirror reflection on the two flat surfaces. The fourth problem models the assembly as an eighth sphere with mirror reflection on the three flat surfaces. This set of problems is designed to illustrate the use of multiple chords in a problem.

```
=CSAS26
BARE 93.2% ENRICHED URANIUM SPHERE
27GROUP INF
URANIUM  1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ PARAM  GEN=103 NPG=1000 END PARAM
READ GEOMETRY
GLOBAL UNIT 1
SPHERE 10  8.714
CUBOID  20  6P8.714
MEDIA  1 1 10
MEDIA  0 1 20 -10
BOUNDARY 20
END GEOMETRY
END DATA
END
```

```
=CSAS26
BARE 93.2% ENRICHED URANIUM SPHERE, HEMISPHERE WITH MIRROR ALBEDO FLAT FACE
27GROUP INF
URANIUM  1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ PARAM  GEN=103 NPG=1000 END PARAM
READ GEOMETRY
GLOBAL UNIT 1
SPHERE 10  8.714  CHORD +X=0.0
CUBOID  20  0.0  8.714 8.714 -8.714 8.714 -8.714
MEDIA  1 1 10
MEDIA  0 1 20 -10
BOUNDARY 20
END GEOMETRY
READ BOUNDS -XB=MIRROR  END BOUNDS
END DATA
END
```

```
=CSAS26
BARE 93.2% ENRICHED URANIUM SPHERE, QUARTER SPHERE WITH MIRROR ALBEDO FLAT
FACES
```

```
27GROUP INF
URANIUM  1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ PARAM  GEN=103 NPG=1000 END PARAM
READ GEOMETRY
GLOBAL UNIT 1
SPHERE 10  8.714  CHORD +X=0.0  CHORD  +Y=0.0
CUBOID  20  8.714 0.0 8.714 0.0 8.714 -8.714
MEDIA  1 1 10
MEDIA  0 1 20 -10
BOUNDARY 20
END GEOMETRY
READ BOUNDS -XY=MIRROR  END BOUNDS
END DATA
END

=CSAS26
BARE 93.2% ENRICHED URANIUM SPHERE, EIGHTH SPHERE WITH MIRROR ALBEDO FLAT
FACES
27GROUP INF
URANIUM  1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ PARAM  GEN=103 NPG=1000 END PARAM
READ GEOMETRY
GLOBAL UNIT 1
SPHERE 10  8.714  CHORD +X=0.0  CHORD  +Y=0.0  CHORD +Z=0.0
CUBOID  20  8.714 0.0 8.714 0.0 8.714 0.0
MEDIA  1 1 10
MEDIA  0 1 20 -10
BOUNDARY 20
END GEOMETRY
READ BOUNDS -FC=MIRROR  END BOUNDS
END DATA
```

**SAMPLE PROBLEM 7  GROTESQUE WITHOUT THE DIAPHRAGM**

The purpose of this problem is to calculate the k-effective of a system composed of eight enriched uranium units placed on a diaphragm, with an irregularly shaped centerpiece positioned in the center hole of the diaphragm. The assembly and centerpiece are shown in Figure C6.B.3. The eight units consist of an approximate parallelepiped with an irregular top, a parallelepiped, and six cylinders of various sizes. The centerpiece, which penetrates the hole in the diaphragm, consists of a cylinder topped by a parallelepiped topped by a hemisphere. The diaphragm is not modeled in this example.

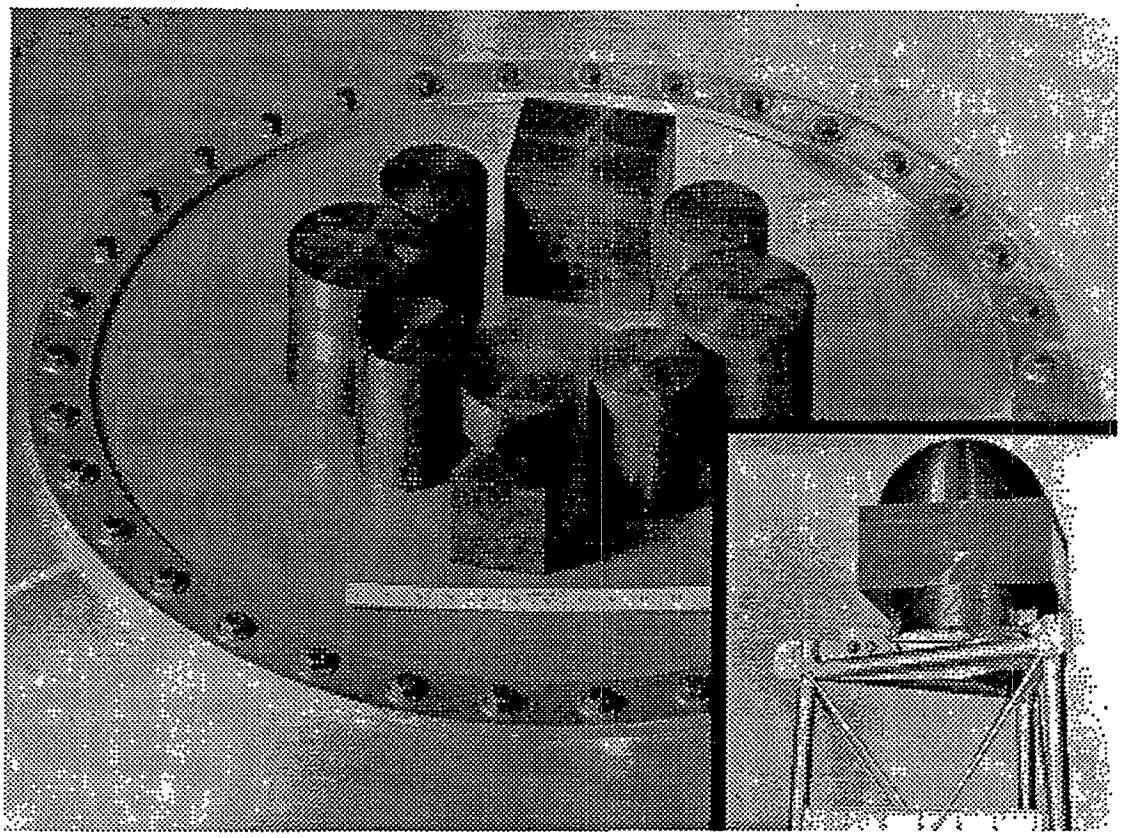


Figure C6.B.3 Grotesque experimental setup

```
=CSAS26
KENO-VI GROTESQUE WITHOUT THE DIAPHRAGM MODELED FROM ORNL/CSD/TM-220
27GROUP INF
URANIUM  1 DEN=18.76 1 293 92235 93.2 92238 5.6 92234 1.0 92236 0.2 END
END COMP
READ PARAM
  GEN=103 NPG=300
  PGM=YES END PARAM
READ GEOM
GLOBAL UNIT 1
'*** ONE THROUGH THREE IS ITEM 1 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
'ONE    TOP PIECE OF ITEM 1
CUBOID 10  2P6.3515 1.2685 -3.8115 13.377 13.058
  ORIGIN Y=-17.464 Z=0.15 ROTATE A2=-1.35
'TWO    MIDDLE PIECE OF ITEM 1
CUBOID 20  2P6.3515 6.3515 -3.8115 13.058 11.155
  ORIGIN Y=-17.464 Z=0.15 ROTATE A2=-1.35
'THREE  BOTTOM PIECE OF ITEM 1
```

```
CUBOID 30  4P6.3515 11.155 0.
 ORIGIN Y=-17.464 Z=0.15 ROTATE A2=-1.35
'*** FOUR IS ITEM 2 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 40 4.555 12.918 0.      ORIGIN X=-12.176 Y=-9.343 Z=0.111
 ROTATE A1=-52.5 A2=-1.400
'*** FIVE IS ITEM 3 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 50 5.761 13.475 0.      ORIGIN X=-16.333 Y=1.681 Z=0.174
 ROTATE A1=83.5 A2=+1.173
'*** SIX IS ITEM 4 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 60 4.5525 12.969 0.   ORIGIN X=-9.539 Y=11.168 Z=0.156
 ROTATE A1=40.5 A2=+1.970
'*** SEVEN AND EIGHT ARE ITEM 5 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
'SEVEN
CUBOID  70  2P3.81 8.13 -4.573 8.91 0.
 ORIGIN Y=15.698 Z=0.290 ROTATE A2=+2.58
'EIGHT
CYLINDER 80 4.573 13.229 0.91   ORIGIN Y=15.698 Z=0.290 ROTATE A2=+2.58
'*** NINE IS ITEM 6 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 90 4.5545 12.974 0.   ORIGIN X=9.854 Y=10.964 Z=0.134
 ROTATE A1=-42.0 A2=+1.680
'*** TEN IS ITEM 7 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 100 5.7495 13.475 0.   ORIGIN X=16.388 Y=1.434 Z=0.140
 ROTATE A1=-86.0 A2=+1.400
'*** ELEVEN IS ITEM 8 IN DRAWING 84-10649 ORNL/CSD/TM-220 ***
CYLINDER 110 4.5565 12.954 0.   ORIGIN X=12.029 Y=-9.398 Z=0.087
 ROTATE A1=38.0 A2=-1.100
'*12 THROUGH 14 IS THE CENTERPIECE IN DRAWING 84-10649 ORNL/CSD/TM-220
'TWELVE
CYLINDER 120 5.757 2.690 0.     ORIGIN X=-0.593 Y=-0.593 Z=-1.755
'THIRTEEN
CUBOID 130 4P6.35 5.718 0.      ORIGIN Z=0.937
'FOURTEEN
SPHERE   140 6.082 CHORD +Z=0.   ORIGIN X=-0.268 Y=0.268 Z=6.655
'*** FIFTEEN IS THE SYSTEM BOUNDARY ***
'FIFTEEN
CUBOID   150 4P25.0 15.0 -2.0
MEDIA 1 1 +10
MEDIA 1 1 +20 -10
MEDIA 1 1 +30 -20
MEDIA 1 1 +40
MEDIA 1 1 +50
MEDIA 1 1 +60
MEDIA 1 1 +70 -80
MEDIA 1 1 +80
MEDIA 1 1 +90
MEDIA 1 1 +100
MEDIA 1 1 +110
```

```
MEDIA 1 1 +120 -130
MEDIA 1 1 +130
MEDIA 1 1 +140 -130
MEDIA 0 1 150 -10 -20 -30 -40 -50 -60 -70 -80 -90 -100
      -110 -120 -130 -140
BOUNDARY 150
END GEOM
READ PLOT
TTL='GROTESQUE X-Y SLICE AT Z=0.5'
PIC=MIX LPI=6.0
NCH=' 16789ABCDEFGH'
XUL=-25.5 YUL=25.5 ZUL=0.5 XLR=25.5 YLR=-25.5 ZLR=.5
UAX=1 VDN=-1  NAX=130 END
TTL='GROTESQUE X-Y SLICE AT Z=2.0'
XUL=-25.5 YUL=25.5 ZUL=2 XLR=25.5 YLR=-25.5 ZLR=2 END
TTL='GROTESQUE X-Y SLICE AT Z=9.5'
XUL=-25.5 YUL=25.5 ZUL=9.5 XLR=25.5 YLR=-25.5 ZLR=9.5 END
TTL='GROTESQUE Y-Z SLICE AT X=-0.593'
XUL=-.593 YUL=-25.5 ZUL=15.5 XLR=-.593 YLR=25.5 ZLR=-3.5
UAX=0 VAX=1  VDN=0 WDN=-1 NAX=130  END
TTL='GROTESQUE X-Z SLICE AT Y=0.0'
XUL=-25.5 YUL=0.0  ZUL=15.5 XLR=25.5 YLR=0.0 ZLR=-3.5
UAX=1 VAX=0 WAX=0 UDN=0 VDN=0 WDN=-1 NAX=130  END
TTL='GROTESQUE X-Z SLICE AT Y=12.125'
XUL=-25.5 YUL=12.125  ZUL=15.5 XLR=25.5 YLR=12.125 ZLR=-3.5
UAX=1 VAX=0 WAX=0 UDN=0 VDN=0 WDN=-1 NAX=130  END
TTL='GROTESQUE X-Z SLICE AT Y=-12.000'
XUL=-25.5 YUL=-12.000  ZUL=15.5 XLR=25.5 YLR=-12.000 ZLR=-3.5
UAX=1 VAX=0 WAX=0 UDN=0 VDN=0 WDN=-1 NAX=130  END
END PLOT
END DATA
END
```

## C6.B.1 REFERENCES

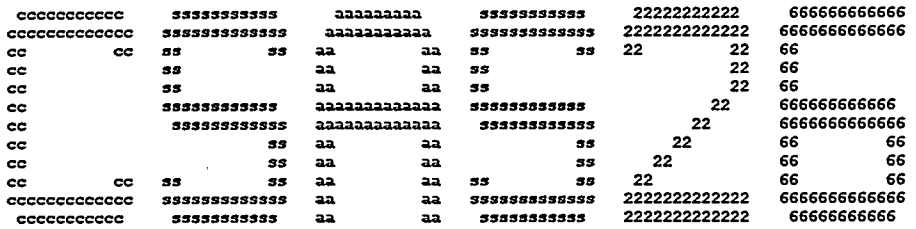

1. J. T. Thomas, "Critical Three-Dimensional Arrays of U(93.2)-Metal Cylinders," *Nucl. Sci. Eng.* **52**, 350 (November 1973).

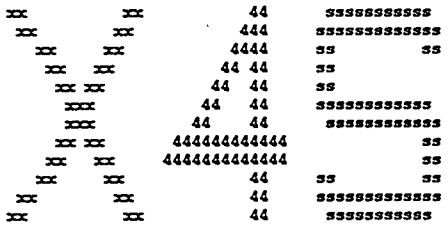
2. J. T. Thomas, *Critical Three-Dimensional Arrays of Neutron-Interacting Units, Part II*, ORNL/TM-868, Union Carbide Corp., Nucl. Div., Oak Ridge Natl. Lab. (July 1964).

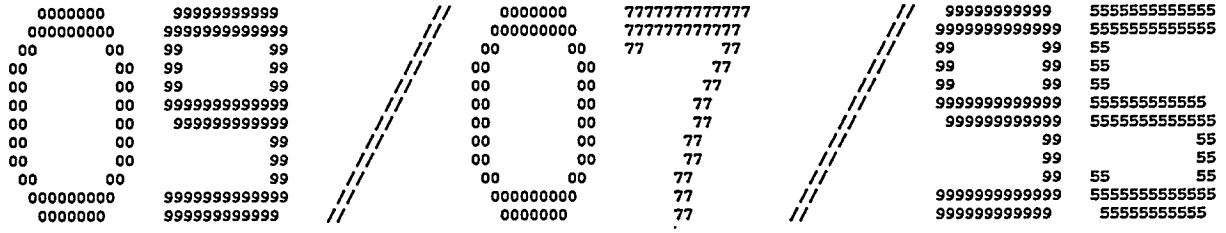
## C6.C SAMPLE OUTPUT

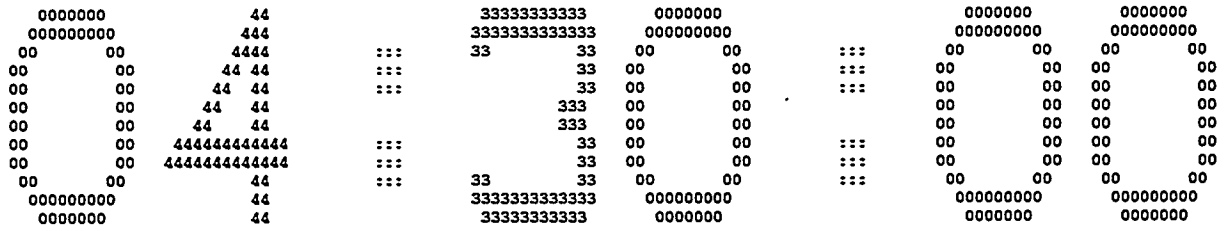
This section contains the computer printout generated by the control module for sample problems 1, 2, 3, and 7 (Y-Pipe, Cross, Sphere, Grotesque). Only the output from CSAS6 is included, output from the functional modules is not included. See Section C6.6 for a generic explanation of the control module output.

The k-effectives resulting from these calculations do not appear in the control module output, These problems describe critical experiments, so the KENO-VI output should yield a k-effective of 1.0 within statistical accuracy. The k-effectives may vary because of differences in the random number sequences on different computers.









```
 sssssssssss     ccccccccccc      aaaaaaaaa      ll              eeeeeeeeeeeee
ssssssssssssss  ccccccccccccc    aaaaaaaaaaa     ll              eeeeeeeeeeeee
ss          ss  cc           cc  aa          aa  ll              ee
ss              cc               aa          aa  ll              ee
ss              cc               aa          aa  ll              ee
ssssssssssss    cc               aaaaaaaaaaaaa  ll              eeeeeeeee
 ssssssssssss   cc               aaaaaaaaaaaaa  ll              eeeeeeeee
          ss    cc               aa          aa  ll              ee
          ss    cc               aa          aa  ll              ee
ss        ss    cc           cc  aa          aa  ll              ee
ssssssssssss    ccccccccccccc    aa          aa  llllllllllllll  eeeeeeeeeeeee
 sssssssssss     ccccccccccc     aa          aa  llllllllllllll  eeeeeeeeeeeee
```

```
**********************************************************************************
**********************************************************************************
**********************************************************************************
*****                                                                        *****
*****                  program verification information                   *****
*****                                                                        *****
*****                 code system:    scale  version:    4.3              *****
*****                                                                        *****
**********************************************************************************
**********************************************************************************
*****                                                                        *****
*****                                                                        *****
*****              program:  csas6                                           *****
*****                                                                        *****
*****         creation date:  08/13/96                                       *****
*****                                                                        *****
*****              library:  /scale4.3/bin                                   *****
*****                                                                        *****
*****                                                                        *****
*****       production code:  csas6                                          *****
*****                                                                        *****
*****              version:  3.18                                            *****
*****                                                                        *****
*****              jobname:  x4s                                             *****
*****                                                                        *****
*****     date of execution:  09/07/95                                       *****
*****                                                                        *****
*****     time of execution:  04:30:00                                       *****
*****                                                                        *****
*****                                                                        *****
**********************************************************************************
**********************************************************************************
**********************************************************************************
```

```
keno-vi  30 deg y  5%uo2f2  907.0g/l  128.2 soln height  k=1.001 +- .004

**** problem parameters ****

lib 27groupndf4  library
mxx            3 mixtures
msc            3 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           1 fuel solutions

**** problem composition description ****

sc  solnuo2f2     standard composition
mx             1 mixture no.
fd      907.0000 solution fuel density
aml       0.0000 acid molarity
vf        1.0000 volume fraction
temp       299.0 deg kelvin
spg       2.0289 specified specific gravity
sc  uo2f2         standard composition
           92000      1.00 atom/molecule
                           92235     5.000 wt%
                           92238    95.000 wt%
            8016      2.00 atoms/molecule
            9019      2.00 atoms/molecule
sc  hfacid        standard composition
            1001      1.00 atom/molecule
            9019      1.00 atom/molecule
sc  h2o           standard composition
            1001      2.00 atoms/molecule
            8016      1.00 atom/molecule
end

sc  al            standard composition
mx             2 mixture no.
vf        1.0000 volume fraction
roth      2.7020 theoretical density
nel            1 no. elements
icp            1 0/1 mixture/compound
           13027      1.00 atom/molecule
end

sc  h2o           standard composition
mx             3 mixture no.
vf        1.0000 volume fraction
roth      0.9982 theoretical density
nel            2 no. elements
icp            1 0/1 mixture/compound
            1001      2.00 atoms/molecule
            8016      1.00 atom/molecule
end

**** problem geometry ****

**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

```
keno-vi  30 deg y  5%uo2f2  907.0g/l  128.2 soln height  k=1.001 +- .004

********** data library information **********

 unit                                                    volume
number              data set name                        name        unit function
------              -------------                        ----        -------------

 89     /scale4.3/data/scale.rev11.sclib                             standard composition library

 82     /scale4.3/data/scale.rev04.xn27                              cross section library

 11     ft11f001                                                     short cross section library

 90     ft90f001                                                     input data direct access


                          standard composition library data
                          ---------------------------------

          unit number  :  89

          dataset name :  /scale4.3/data/scale.rev11.sclib

          library title:  scale-4 standard composition library
                          637 standard compositions, 490 nuclides
                          90 elements with variable isotopic distrbutions.

          creation date:   2/ 6/96


                           cross section library data
                           --------------------------

          unit number  :  82

          dataset name :  /scale4.3/data/scale.rev04.xn27

          library title:  scale 4.2 - 27 group neutron group library
                             based on endf-b version 4 data
                                compiled for nrc     1/27/89
                                   last updated                    08/12/94
                                      l.m.petrie   -   ornl

.....    0 io's were used before reading keno v data    .....

.....    0 io's were used reading the keno-vi parameter data    .....
```

```
                    keno-vi  30 deg y  5%uo2f2  907.0g/l  128.2 soln height  k=1.001 +- .004

                                                geometry description input

  1 cylinder          radius =  13.950     +z=  135.40     -z= -75.400

  2 cylinder          radius =  14.110     +z=  135.40     -z= -76.700

  3 cylinder          radius =  13.950     +z=  125.40     -z=  0.0000
                    angle 1=   0.000     angle 2= -29.260     angle 3=   0.000

  4 cylinder          radius =  14.110     +z=  126.04     -z=  0.0000
                    angle 1=   0.000     angle 2= -29.260     angle 3=   0.000

  5 cuboid               +x =  37.000     -x = -37.000     +y =  100.00     -y = -37.000     +z =  52.800     -z = -75.400

  6 cuboid               +x =  37.000     -x = -37.000     +y =  100.00     -y = -37.000     +z =  135.40     -z = -99.600

sector data associated with media   1 and importance region   1
   10   50

icnt/irgsec(ii),ii=1,icnt    2   10   50

sector data associated with media   2 and importance region   1
   20  -10  -30

icnt/irgsec(ii),ii=1,icnt    3   20  -10  -30

sector data associated with media   1 and importance region   1
   30   50  -10

icnt/irgsec(ii),ii=1,icnt    3   30   50  -10

sector data associated with media   2 and importance region   1
   40  -30  -20

icnt/irgsec(ii),ii=1,icnt    3   40  -30  -20

sector data associated with media   0 and importance region   1
   10  -50

icnt/irgsec(ii),ii=1,icnt    2   10  -50

sector data associated with media   0 and importance region   1
   30  -50  -10

icnt/irgsec(ii),ii=1,icnt    3   30  -50  -10

sector data associated with media   3 and importance region   1
   60  -20  -40  -10

icnt/irgsec(ii),ii=1,icnt    4   60  -20  -40  -10

vector definition array for unit     1
    60

                          *************** data reading completed ***************

                           ........    0 io's were used preparing the keno-vi input data    ........

                           ........    0 io's were used loading the keno-vi data    ........

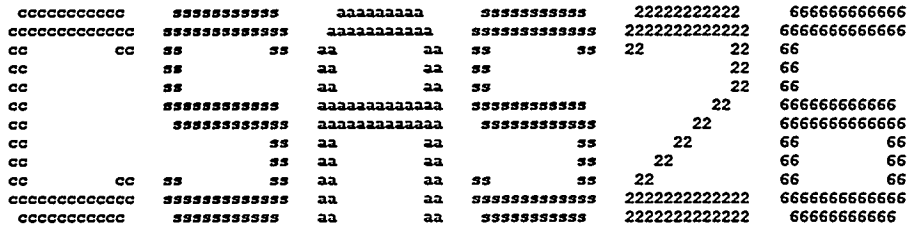
                           ........    0 io's were used loading the data    ........

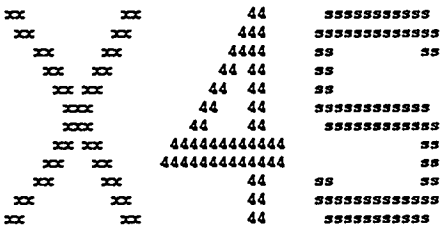
                           ........    0 io's were used checking the keno-vi geometry data    ........
                           *****  restart data has been written on unit 95  *****

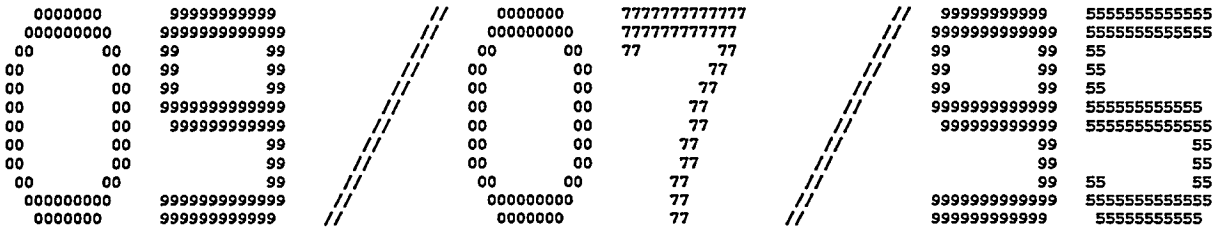
                           ........    0 io's were used writing the keno-vi - csas data    ........

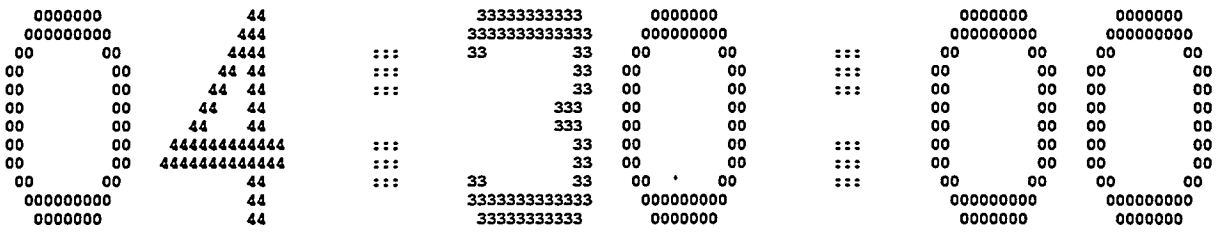
                           ........    0 io's were used processing csas input data    ........

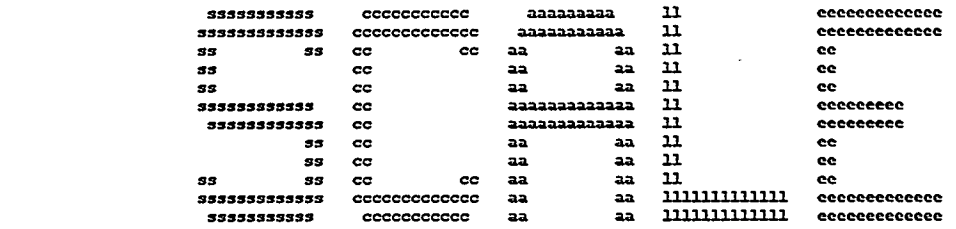
 control module csas26   is complete.
```












```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                 code system:    scale  version:    4.3           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  csas6                                        *****
*****                                                                     *****
*****         creation date:  08/13/96                                    *****
*****                                                                     *****
*****              library:  /scale4.3/bin                                *****
*****                                                                     *****
*****                                                                     *****
*****       production code:  csas6                                       *****
*****                                                                     *****
*****              version:  3.18                                         *****
*****                                                                     *****
*****              jobname:  x4s                                          *****
*****                                                                     *****
*****     date of execution:  09/07/95                                    *****
*****                                                                     *****
*****     time of execution:  04:30:00                                    *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
kenovi 89 deg cross 5% uo2f2 soln, plexiglass pipes, h2o refl.

**** problem parameters ****

lib 27groupndf4  library
mxx            3 mixtures
msc            3 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           1 fuel solutions


**** problem composition description ****

sc  solnuo2f2    standard composition
mx             1 mixture no.
fd      896.1000 solution fuel density
aml       0.0000 acid molarity
vf        1.0000 volume fraction
temp       298.0 deg kelvin
spg       2.0150 specified specific gravity
sc  uo2f2        standard composition
           92000      1.00 atom/molecule
                         92235     5.000 wt%
                         92238    95.000 wt%
            8016      2.00 atoms/molecule
            9019      2.00 atoms/molecule
sc  hfacid       standard composition
            1001      1.00 atom/molecule
            9019      1.00 atom/molecule
sc  h2o          standard composition
            1001      2.00 atoms/molecule
            8016      1.00 atom/molecule
end

sc  plexiglass   standard composition
mx             3 mixture no.
vf        1.0000 volume fraction
roth      1.1800 theoretical density
nel            3 no. elements
icp            1 0/1 mixture/compound
            1001      8.00 atoms/molecule
            6012      5.00 atoms/molecule
            8016      2.00 atoms/molecule
end

sc  h2o          standard composition
mx             2 mixture no.
vf        1.0000 volume fraction
roth      0.9982 theoretical density
nel            2 no. elements
icp            1 0/1 mixture/compound
            1001      2.00 atoms/molecule
            8016      1.00 atom/molecule
end


**** problem geometry ****


**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

```
kenovi 89 deg cross 5% uo2f2 soln, plexiglass pipes, h2o refl.

********** data library information **********

 unit                                                    volume
number              data set name                        name       unit function
------              -------------                        ----       -------------

 89     /scale4.3/data/scale.rev11.sclib                           standard composition library

 82     /scale4.3/data/scale.rev04.xn27                            cross section library

 11     ft11f001                                                   short cross section library

 90     ft90f001                                                   input data direct access

                                standard composition library data
                                ---------------------------------

          unit number  :  89

          dataset name :  /scale4.3/data/scale.rev11.sclib

          library title:  scale-4 standard composition library
                          637 standard compositions, 490 nuclides
                          90 elements with variable isotopic distrbutions.

          creation date:   2/ 6/96

                                   cross section library data
                                   --------------------------

          unit number  :  82

          dataset name :  /scale4.3/data/scale.rev04.xn27

          library title:  scale 4.2 - 27 group neutron group library
                            based on endf-b version 4 data
                               compiled for nrc     1/27/89
                                  last updated                                  08/12/94
                                     l.m.petrie   -   ornl

 ........    0 io's were used before reading keno v data    ........

 ........    0 io's were used reading the keno-vi parameter data    ........

 *************** data reading completed ***************

 ........    0 io's were used preparing the keno-vi input data    ........

 ........    0 io's were used loading the keno-vi data    ........

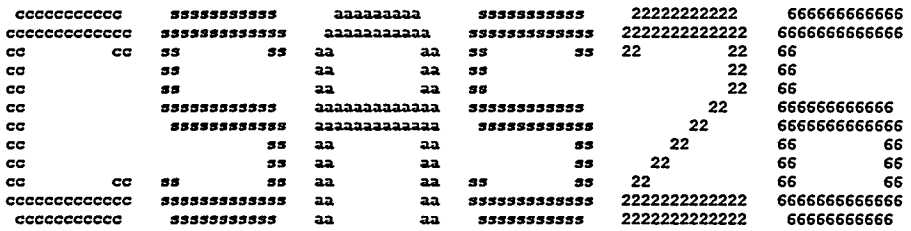
 ........    0 io's were used loading the data    ........

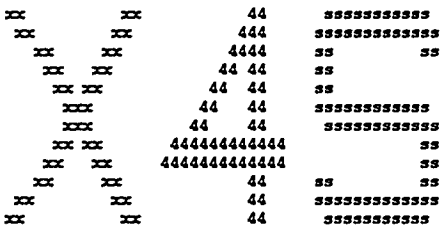
 ........    0 io's were used checking the keno-vi geometry data    ........
 *****  restart data has been written on unit 95  *****

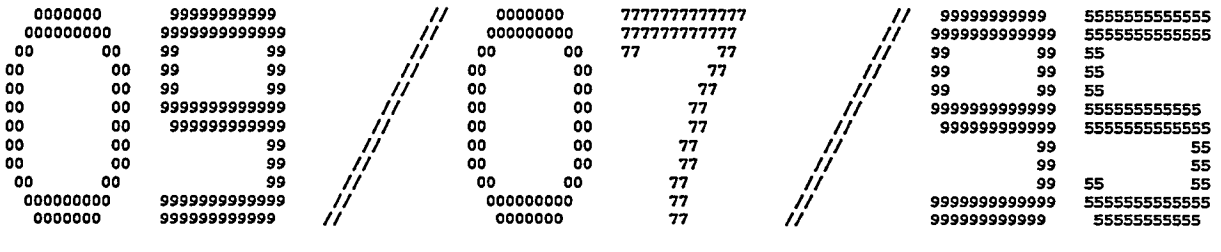
 ........    0 io's were used writing the keno-vi - csas data    ........

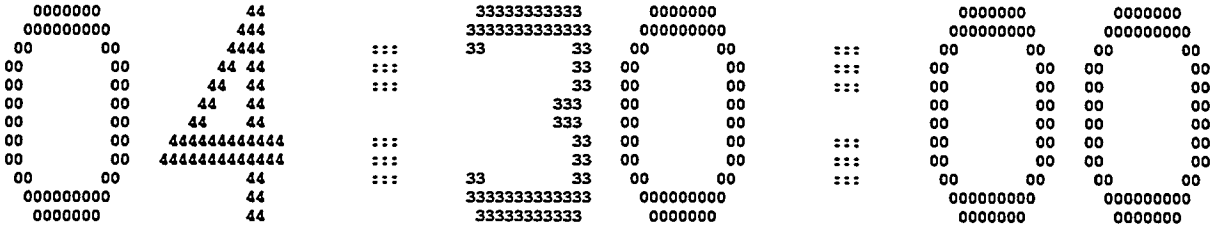
 ........    0 io's were used processing csas input data    ........

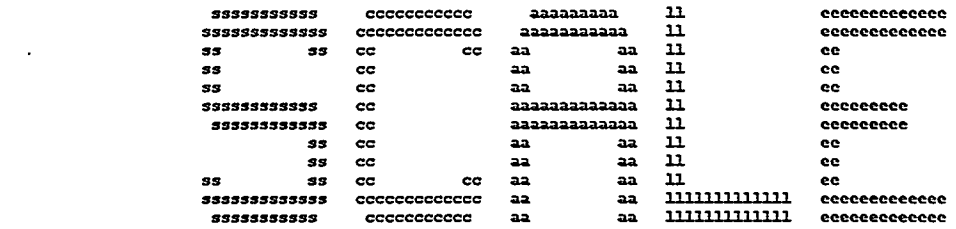
control module csas26   is complete.
```












```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                code system:    scale  version:    4.3           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****              program:  csas6                                        *****
*****                                                                     *****
*****         creation date:  08/13/96                                    *****
*****                                                                     *****
*****              library:  /scale4.3/bin                                *****
*****                                                                     *****
*****                                                                     *****
*****       production code:  csas6                                       *****
*****                                                                     *****
*****              version:  3.18                                         *****
*****                                                                     *****
*****              jobname:  x4s                                          *****
*****                                                                     *****
*****     date of execution:  09/07/95                                    *****
*****                                                                     *****
*****     time of execution:  04:30:00                                    *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
bare 93.2% enriched uranium sphere


**** problem parameters ****

lib 27groupndf4  library
mxx             1 mixtures
msc             1 composition specifications
izm             1 material zones
ge  infhommedium geometry
more            0 0/1 do not read/read optional parameter data
msln            0 fuel solutions


**** problem composition description ****

sc  uranium      standard composition
mx              1 mixture no.
vf         1.0000 volume fraction
roth      18.7600 specified density
nel             1 no. elements
icp             1 0/1 mixture/compound
temp        293.0 deg kelvin
            92000      1.00 atom/molecule
                              92235    93.200 wt%
                              92238     5.600 wt%
                              92234     1.000 wt%
                              92236     0.200 wt%
end


**** problem geometry ****


**** infinite homogeneous medium ****
mfuel           1 mixture no. of the infinite homogeneous medium
```

```
*** bare 93.2% enriched uranium sphere ***

*** ********** data library information ********** ***

***   unit                                          volume                                ***
***  number          data set name                  name      unit function               ***
***  ------          -------------                  ----      -------------               ***
***    89     /scale4.3/data/scale.rev11.sclib                 standard composition library ***
***    82     /scale4.3/data/scale.rev04.xn27                  cross section library        ***
***    11     ft11f001                                         short cross section library  ***
***    90     ft90f001                                         input data direct access     ***

*** standard composition library data ***
***  unit number  :  89 ***
***  dataset name :  /scale4.3/data/scale.rev11.sclib ***
***  library title:  scale-4 standard composition library ***
***                  637 standard compositions, 490 nuclides ***
***                  90 elements with variable isotopic distributions. ***
***  creation date:   2/ 6/96 ***

*** cross section library data ***
***  unit number  :  82 ***
***  dataset name :  /scale4.3/data/scale.rev04.xn27 ***
***  library title:  scale 4.2 - 27 group neutron group library ***
***                    based on endf-b version 4 data ***
***                      compiled for nrc     1/27/89 ***
***                        last updated                  08/12/94 ***
***                          l.m.petrie   -   ornl ***

 ........    0 io's were used before reading keno v data    ........

 ........    0 io's were used reading the keno-vi parameter data    ........

 *************** data reading completed ***************

 ........    0 io's were used preparing the keno-vi input data    ........

 ........    0 io's were used loading the keno-vi data    ........

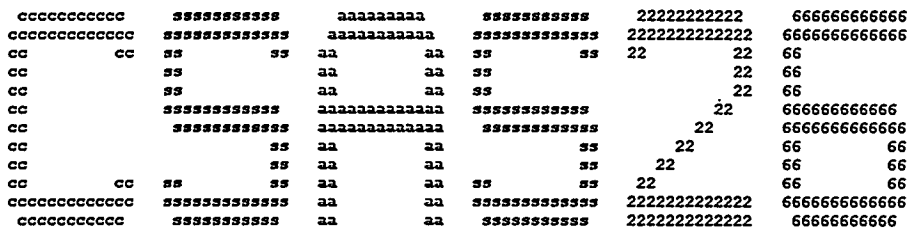
 ........    0 io's were used loading the data    ........

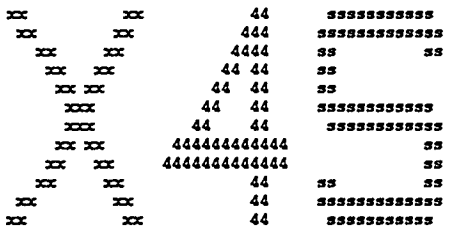
 ........    0 io's were used checking the keno-vi geometry data    ........
 *****  restart data has been written on unit 95  *****

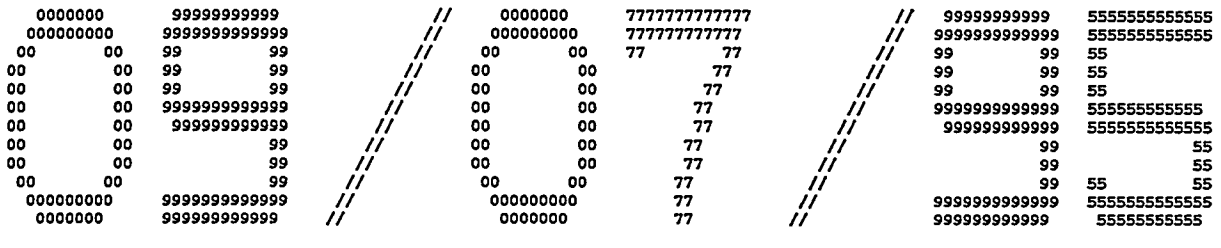
 ........    0 io's were used writing the keno-vi - csas data    ........

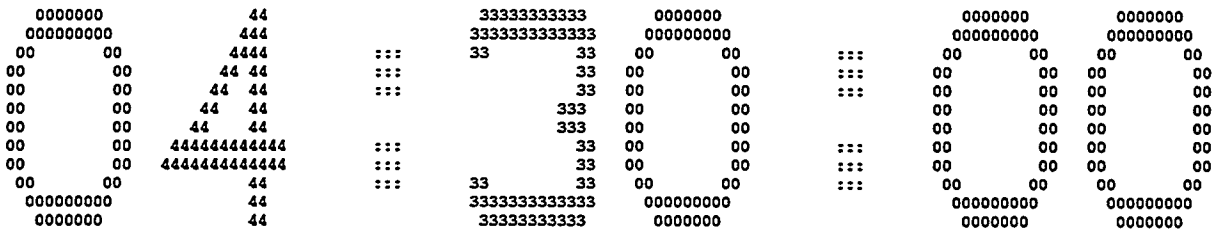
 ........    0 io's were used processing csas input data    ........

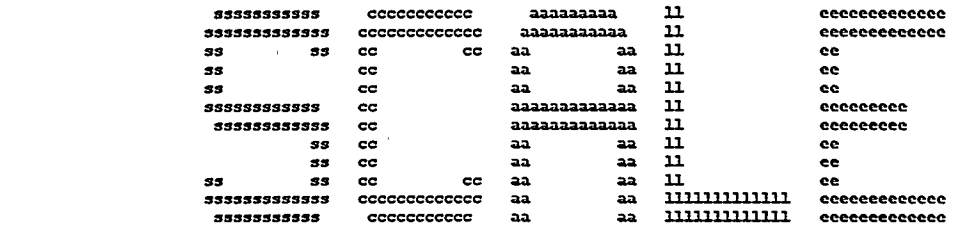
control module csas26   is complete.
```












```
*******************************************************************************
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                  program verification information                *****
*****                                                                     *****
*****                code system:    scale  version:    4.3           *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*****                                                                     *****
*****                                                                     *****
*****             program:  csas6                                         *****
*****                                                                     *****
*****       creation date:  08/13/96                                      *****
*****                                                                     *****
*****             library:  /scale4.3/bin                                 *****
*****                                                                     *****
*****                                                                     *****
*****      production code:  csas6                                        *****
*****                                                                     *****
*****             version:  3.18                                          *****
*****                                                                     *****
*****             jobname:  x4s                                           *****
*****                                                                     *****
*****    date of execution:  09/07/95                                     *****
*****                                                                     *****
*****    time of execution:  04:30:00                                     *****
*****                                                                     *****
*****                                                                     *****
*******************************************************************************
*******************************************************************************
*******************************************************************************
```

```
keno-vi grotesque without the diaphragm modeled from ornl/csd/tm-220


**** problem parameters ****

lib 27groupndf4  library
mxx            1 mixtures
msc            1 composition specifications
izm            1 material zones
ge  infhommedium geometry
more           0 0/1 do not read/read optional parameter data
msln           0 fuel solutions


**** problem composition description ****

sc  uranium      standard composition
mx             1 mixture no.
vf        1.0000 volume fraction
roth     18.7600 specified density
nel            1 no. elements
icp            1 0/1 mixture/compound
temp       293.0 deg kelvin
           92000      1.00 atom/molecule
                           92235    93.200 wt%
                           92238     5.600 wt%
                           92234     1.000 wt%
                           92236     0.200 wt%
end


**** problem geometry ****


**** infinite homogeneous medium ****
mfuel          1 mixture no. of the infinite homogeneous medium
```

```
keno-vi grotesque without the diaphragm modeled from ornl/csd/tm-220

********** data library information **********

 unit                                                  volume
number              data set name                     name        unit function
------              -------------                     ----        -------------

  89     /scale4.3/data/scale.rev11.sclib                        standard composition library

  82     /scale4.3/data/scale.rev04.xn27                         cross section library

  11     ft11f001                                                short cross section library

  90     ft90f001                                                input data direct access


                          standard composition library data
                          ---------------------------------

     unit number  :  89

     dataset name :  /scale4.3/data/scale.rev11.sclib

     library title:  scale-4 standard composition library
                     637 standard compositions, 490 nuclides
                     90 elements with variable isotopic distrbutions.

     creation date:   2/ 6/96


                           cross section library data
                           --------------------------

     unit number  :  82

     dataset name :  /scale4.3/data/scale.rev04.xn27

     library title:  scale 4.2 - 27 group neutron group library
                        based on endf-b version 4 data
                           compiled for nrc     1/27/89
                              last updated                    08/12/94
                                 l.m.petrie   -   ornl


.........    0 io's were used before reading keno v data    .........

.........    0 io's were used reading the keno-vi parameter data    .........
```

```
keno-vi grotesque without the diaphragm modeled from ornl/csd/tm-220

geometry description input

'*** one through three is item 1 in drawing 84-10649 ornl/csd/tm-220 ***

'one    top piece of item 1

 1 cuboid              +x =  6.3515     -x = -6.3515     +y =  1.2685     -y = -3.8115     +z =  13.377     -z =  13.058
                    xorigin=  0.000     yorigin= -17.464     zorigin=  0.150

'two    middle piece of item 1
                    angle 1=  0.000     angle 2= -1.350     angle 3=  0.000

 2 cuboid              +x =  6.3515     -x = -6.3515     +y =  6.3515     -y = -3.8115     +z =  13.058     -z =  11.155
                    xorigin=  0.000     yorigin= -17.464     zorigin=  0.150

'three  bottom piece of item 1
                    angle 1=  0.000     angle 2= -1.350     angle 3=  0.000

 3 cuboid              +x =  6.3515     -x = -6.3515     +y =  6.3515     -y = -6.3515     +z =  11.155     -z =  0.0000
                    xorigin=  0.000     yorigin= -17.464     zorigin=  0.150

'*** four is item 2 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1=  0.000     angle 2= -1.350     angle 3=  0.000

 4 cylinder            radius =  4.5550     +z=  12.918     -z=  0.0000
                    xorigin= -12.176     yorigin=  -9.343     zorigin=  0.111

'*** five is item 3 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1= -52.500     angle 2= -1.400     angle 3=  0.000

 5 cylinder            radius =  5.7610     +z=  13.475     -z=  0.0000
                    xorigin= -16.333     yorigin=  1.681     zorigin=  0.174

'*** six is item 4 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1=  83.500     angle 2=  1.173     angle 3=  0.000

 6 cylinder            radius =  4.5525     +z=  12.969     -z=  0.0000
                    xorigin=  -9.539     yorigin=  11.168     zorigin=  0.156

'*** seven and eight are item 5 in drawing 84-10649 ornl/csd/tm-220 ***

'seven
                    angle 1=  40.500     angle 2=  1.970     angle 3=  0.000

 7 cuboid              +x =  3.8100     -x = -3.8100     +y =  8.1300     -y = -4.5730     +z =  8.9100     -z =  0.0000
                    xorigin=  0.000     yorigin=  15.698     zorigin=  0.290

'eight
                    angle 1=  0.000     angle 2=  2.580     angle 3=  0.000

 8 cylinder            radius =  4.5730     +z=  13.229     -z= 0.91000
                    xorigin=  0.000     yorigin=  15.698     zorigin=  0.290

'*** nine is item 6 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1=  0.000     angle 2=  2.580     angle 3=  0.000

 9 cylinder            radius =  4.5545     +z=  12.974     -z=  0.0000
                    xorigin=  9.854     yorigin=  10.964     zorigin=  0.134
'*** ten is item 7 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1= -42.000     angle 2=  1.680     angle 3=  0.000

10 cylinder            radius =  5.7495     +z=  13.475     -z=  0.0000
                    xorigin=  16.388     yorigin=  1.434     zorigin=  0.140

'*** eleven is item 8 in drawing 84-10649 ornl/csd/tm-220 ***
                    angle 1= -86.000     angle 2=  1.400     angle 3=  0.000

11 cylinder            radius =  4.5565     +z=  12.954     -z=  0.0000
                    xorigin=  12.029     yorigin=  -9.398     zorigin=  0.087

'*12 through 14 is the centerpiece in drawing 84-10649 ornl/csd/tm-220

'twelve
                    angle 1=  38.000     angle 2= -1.100     angle 3=  0.000

12 cylinder            radius =  5.7570     +z=  2.6900     -z=  0.0000

'thirteen
                    xorigin=  -0.593     yorigin=  -0.593     zorigin=  -1.755

13 cuboid             +x =  6.3500     -x = -6.3500     +y =  6.3500     -y = -6.3500     +z =  5.7180     -z =  0.0000

'fourteen
                    xorigin=  0.000     yorigin=  0.000     zorigin=  0.937
```

```
 14 sphere             radius =  6.0820

 14  +z=               0.0000

'*** fifteen is the system boundary ***
'fifteen
                  xorigin=  -0.268     yorigin=   0.268     zorigin=   6.655

 15 cuboid              +x =  25.000     -x = -25.000     +y =  25.000     -y = -25.000     +z =  15.000     -z = -2.0000

sector data associated with media   1 and importance region   1
   10

icnt/irgsec(ii),ii=1,icnt    1   10

sector data associated with media   1 and importance region   1
   20  -10

icnt/irgsec(ii),ii=1,icnt    2   20  -10

sector data associated with media   1 and importance region   1
   30  -20

icnt/irgsec(ii),ii=1,icnt    2   30  -20

sector data associated with media   1 and importance region   1
   40

icnt/irgsec(ii),ii=1,icnt    1   40

sector data associated with media   1 and importance region   1
   50

icnt/irgsec(ii),ii=1,icnt    1   50

sector data associated with media   1 and importance region   1
   60

icnt/irgsec(ii),ii=1,icnt    1   60

sector data associated with media   1 and importance region   1
   70  -80

icnt/irgsec(ii),ii=1,icnt    2   70  -80

sector data associated with media   1 and importance region   1
   80

icnt/irgsec(ii),ii=1,icnt    1   80

sector data associated with media   1 and importance region   1
   90

icnt/irgsec(ii),ii=1,icnt    1   90

sector data associated with media   1 and importance region   1
  100

icnt/irgsec(ii),ii=1,icnt    1  100

sector data associated with media   1 and importance region   1
  110

icnt/irgsec(ii),ii=1,icnt    1  110

sector data associated with media   1 and importance region   1
  120  ***

icnt/irgsec(ii),ii=1,icnt    2  120 -130

sector data associated with media   1 and importance region   1
  130

icnt/irgsec(ii),ii=1,icnt    1  130

sector data associated with media   1 and importance region   1
  140  ***

icnt/irgsec(ii),ii=1,icnt    2  140 -130

sector data associated with media   0 and importance region   1
  150  -10  -20  -30  -40  -50  -60  -70  -80  -90  ***  ***  ***  ***  ***

icnt/irgsec(ii),ii=1,icnt   15  150  -10  -20  -30  -40  -50  -60  -70  -80

icnt/irgsec(ii),ii=1,icnt  -90 -100 -110 -120 -130 -140
```

```
vector definition array for unit     1
   150
                              ********** data reading completed **********
                              ........    0 io's were used preparing the keno-vi input data    ........
                              ........    0 io's were used loading the keno-vi data    ........
                              ........    0 io's were used loading the data    ........
                              ........    0 io's were used checking the keno-vi geometry data    ........
                              *****  restart data has been written on unit 95  *****
                              ........    0 io's were used writing the keno-vi - csas data    ........
                              ........    0 io's were used processing csas input data    ........

 control module csas26   is complete.
```

NRC FORM 335 (2-89) NRCM 1102, 3201, 3202

U.S. NUCLEAR REGULATORY COMMISSION

# BIBLIOGRAPHIC DATA SHEET

*(See instructions on the reverse)*

1. REPORT NUMBER (Assigned by NRC, Add Vol., Supp., Rev., and Addendum Numbers, if any.)

NUREG/CR-0200, Rev. 5
ORNL/NUREG/CSD-2R5
Vol. 1, Part 1

2. TITLE AND SUBTITLE

SCALE: A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluation

Control Modules
C4, C6

3. DATE REPORT PUBLISHED

| MONTH | YEAR |
|---|---|
| March | 1997 |

4. FIN OR GRANT NUMBER

B0009

5. AUTHOR(S)

6. TYPE OF REPORT

7. PERIOD COVERED *(Inclusive Dates)*

8. PERFORMING ORGANIZATION - NAME AND ADDRESS *(If NRC, provide Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address; if contractor, provide name and mailing address.)*

Oak Ridge National Laboratory
Oak Ridge, TN 37831-6370

9. SPONSORING ORGANIZATION - NAME AND ADDRESS *(If NRC, type "Same as above"; if contractor, provide NRC Division, Office or Region, U.S. Nuclear Regulatory Commission, and mailing address.)*

Spent Fuel Project Office
Office of Nuclear Material Safety and Safeguards
U.S. Nuclear Regulatory Commission
Washington, DC 20555-0001

10. SUPPLEMENTARY NOTES

11. ABSTRACT *(200 words or less)*

SCALE-a modular code system for Standardized Computer Analyses Licensing Evaluation- has been developed by Oak Ridge National Laboratory at the request of the U.S. Nuclear Regulatory Commission. The SCALE system utilizes well-established computer codes and methods within standard analysis sequences that (1) allow an input format designed for the occasional user and/or novice, (2) automated the data processing and coupling between modules, and (3) provide accurate and reliable results. System development has been directed at problem-dependent cross-section processing and analysis of criticality safety, shielding, heat transfer, and depletion/decay problems. Since the initial release of SCALE in 1980, the code system has been heavily used for evaluation of nuclear fuel facility and package designs. This revision documents Version 4.3 of the system.

12. KEY WORDS/DESCRIPTORS *(List words or phrases that will assist researchers in locating the report.)*

SCALE, cross sections, criticality safety, shielding, heat transfer, depletion, decay, spent fuel, KENO, ORIGEN, XSDRNPM, MORSE, NITAWL

13. AVAILABILITY STATEMENT

unlimited

14. SECURITY CLASSIFICATION

*(This Page)* unclassified

*(This Report)* unclassified

15. NUMBER OF PAGES

16. PRICE



This form was electronically produced by Elite Federal Forms, Inc.

Instructions for Tab Inserts

| Tab Name | Insert in Vol. No. | Insert at Page No. |
|---|---|---|
| CSAS Introduction/Theory | 1, Part 1 | Title page, Sect. C4 |
| CSAS Program Flow | 1, Part 1 | C4.3.1 |
| CSAS Input | 1, Part 1 | C4.4.1 |
| CSAS Sample Problems | 1, Part 1 | C4.B.1 |
| CSAS6 Introduction/Theory | 1, Part 1 | Title page, Sect. C6 |
| CSAS6 Program Flow | 1, Part 1 | C6.3.1 |
| CSAS6 Input | 1, Part 1 | C6.4.1 |
| CSAS6 Sample Problems | 1, Part 1 | C6.B.1 |
| SAS1 Introduction/Theory | 1, Part 2 | Title page, Sect. S1 |
| SAS1 Program Flow | 1, Part 2 | S1.3.1 |
| SAS1 Input | 1, Part 2 | S1.4.1 |
| SAS1 Sample Problems | 1, Part 2 | S1.5.1 |
| SAS2H Introduction/Theory | 1, Part 2 | Title page, Sect. S2 |
| SAS2H Program Flow | 1, Part 2 | S2.4.1 |
| SAS2H Input | 1, Part 2 | S2.5.1 |
| SAS2H Sample Problems | 1, Part 2 | S2.6.1 |
| SAS3 Introduction/Theory | 1, Part 2 | Title page, Sect. S3 |
| SAS3 Program Flow | 1, Part 2 | S3.3.1 |
| SAS3 Input | 1, Part 2 | S3.5.1 |
| SAS3 Sample Problems | 1, Part 2 | S3.6.1 |
| SAS4 Introduction/Theory | 1, Part 2 | Title page, Sect. S4 |
| SAS4 Program Flow | 1, Part 2 | S4.3.1 |
| SAS4 Input | 1, Part 2 | S4.4.1 |
| SAS4 Sample Problems | 1, Part 2 | S4.5.1 |
| QADS Introduction/Theory | 1, Part 2 | Title page, Sect. S5 |
| QADS Input | 1, Part 2 | S5.3.2 |
| QADS Sample Problems | 1, Part 2 | S5.4.1 |
| HTAS1 Introduction/Theory | 1, Part 2 | Title page, Sect. H1 |
| HTAS1 Program Flow | 1, Part 2 | H1.3.1 |
| HTAS1 Input | 1, Part 2 | H1.4.1 |
| HTAS1 Sample Problems | 1, Part 2 | H1.5.1 |
| BONAMI Introduction/Theory | 2, Part 1 | Title page, Sect. F1 |
| BONAMI Program Flow | 2, Part 1 | F1.4.1 |
| BONAMI Input | 2, Part 1 | F1.5.1 |
| BONAMI Sample Problems | 2, Part 1 | F1.6.1 |
| NITAWL-II Introduction/Theory | 2, Part 1 | Title page, Sect. F2 |
| NITAWL-II Program Flow | 2, Part 1 | F2.6.1 |

| | | |
|---|---|---|
| NITAWL-II Input | 2, Part 1 | F2.7.1 |
| NITAWL-II Sample Problems | 2, Part 1 | F2.8.1 |
| XSDRNPM Introduction/Theory | 2, Part 1 | Title page, Sect. F3 |
| XSDRNPM Program Flow | 2, Part 1 | F3.4.1 |
| XSDRNPM Input | 2, Part 1 | F3.5.1 |
| XSDRNPM Sample Problems | 2, Part 1 | F3.7.1 |
| XSDOSE Introduction/Theory | 2, Part 1 | Title page, Sect. F4 |
| XSDOSE Program Flow | 2, Part 1 | F4.3.1 |
| XSDOSE Input | 2, Part 1 | F4.4.1 |
| XSDOSE Sample Problems | 2, Part 1 | F4.5.1 |
| COUPLE Introduction/Theory | 2, Part 1 | Title page, Sect. F6 |
| COUPLE Program Flow | 2, Part 1 | F6.3.1 |
| COUPLE Input | 2, Part 1 | F6.5.1 |
| COUPLE Sample Problems | 2, Part 1 | F6.6.1 |
| ORIGEN-S Introduction/Theory | 2, Part 1 | Title page, Sect. F7 |
| ORIGEN-S Program Flow | 2, Part 1 | F7.5.1 |
| ORIGEN-S Input | 2, Part 1 | F7.6.1 |
| ORIGEN-S Sample Problems | 2, Part 1 | F7.7.1 |
| ICE Introduction/Theory | 2, Part 1 | Title page, Sect. F8 |
| ICE Program Flow | 2, Part 1 | F8.3.1 |
| ICE Input | 2, Part 1 | F8.4.1 |
| ICE Sample Problems | 2, Part 1 | F8.5.1 |
| MORSE-SGC Introduction/Theory | 2, Part 2 | Title page, Sect. F9 |
| MORSE-SGC Input | 2, Part 2 | F9.A.1 |
| MORSE-SGC Sample Problems | 2, Part 2 | F9.C.1 |
| HEATING Introduction/Theory | 2, Part 2 | Title page, Sect. F10 |
| HEATING Input | 2, Part 2 | F10.4.1 |
| HEATING Sample Problems | 2, Part 2 | F10.C.1 |
| KENO V.a Introduction/Theory | 2, Part 2 | Title page, Sect. F11 |
| KENO V.a Program Flow | 2, Part 2 | F11.3.1 |
| KENO V.a Input | 2, Part 2 | F11.C.1 |
| KENO V.a Sample Problems | 2, Part 2 | F11.D.1 |
| OCULAR Introduction/Theory | 2, Part 3 | Title page, Sect. F16 |
| OCULAR Program Flow | 2, Part 3 | F16.3.1 |
| OCULAR Input | 2, Part 3 | F16.4.1 |
| OCULAR Sample Problems | 2, Part 3 | F16.A.1 |
| KENO VI Introduction/Theory | 2, Part 3 | Title page, Sect. F17 |
| KENO VI Program Flow | 2, Part 3 | F17.3.1 |
| KENO VI Input | 2, Part 3 | F17.C.1 |
| KENO VI Sample Problems | 2, Part 3 | F17.D.1 |

| | | |
|---|---|---|
| M1 SYSTEM DRIVER | 3 | Title page, Sect. M1 |
| M2 SUBROUTINE LIBRARY | 3 | Title page, Sect. M2 |
| M3 FREE-FORM INPUT | 3 | Title page, Sect. M3 |
| M4 CROSS-SECTION LIBRARIES | 3 | Title page, Sect. M4 |
| M5 THERMAL MATL LIBRARIES | 3 | Title page, Sect. M5 |
| M6 ORIGEN-S DATA LIBRARIES | 3 | Title page, Sect. M6 |
| M7 MIPLIB | 3 | Title page, Sect. M7 |
| M8 STANDARD COMP LIBRARY | 3 | Title page, Sect. M8 |
| M9 MARSLIB | 3 | Title page, Sect. M9 |
| M10 FIDO INPUT | 3 | Title page, Sect. M10 |
| M13 PICTURE | 3 | Title page, Sect. M13 |
| M14 COMPOZ | 3 | Title page, Sect. M14 |
| M15 UTILITY MODULES | 3 | Title page, Sect. M15 |
| M16 COMMENT CONVERSION CODE | 3 | Title page, Sect. M16 |