A GRAPHICAL KERNEL SYSTEM (GKS) VERSION OF COMPUTER PROGRAM MODPATH-PLOT FOR DISPLAYING PATH LINES GENERATED FROM THE U.S. GEOLOGICAL SURVEY THREE-DIMENSIONAL GROUND-WATER FLOW MODEL

by David W. Pollock

U.S. GEOLOGICAL SURVEY

Open File Report 89-622

Reston, Virginia
1990

DEPARTMENT OF THE INTERIOR

MANUEL LUJAN, JR., Secretary

U. S. GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information write to:

David W. Pollock
U. S. Geological Survey
411 National Center
Reston, VA 22092
(703) 648-5007

This report can be purchased from:

U. S. Geological Survey
Books and Open-File Report Section
Federal Center, Box 25425
Denver, CO 80225
(303)236-7476

Preface

This report describes a new version of the particle tracking graphics program MODPATH-PLOT that uses the Graphical Kernel System (GKS) graphics standard. The user is requested to notify the originating office of any errors found in this report or in the computer programs. Updates may occasionally be made to both the report and the computer program. Users who wish to be added to the mailing list to receive updates, if any, may send a request to the following address:

David W. Pollock
U.S. Geological Survey
Office of Ground Water
411 National Center
Reston, VA 22092

The FORTRAN source code for the computer program is available at cost of processing from:

U.S. Geological Survey
WATSTORE Program Office
437 National Center
Reston, VA 22092
Telephone: (703) 648-5686

Contents

A GRAPHICAL KERNEL SYSTEM (GKS) VERSION OF COMPUTER PROGRAM MODPATH-PLOT FOR DISPLAYING PATH LINES GENERATED FROM THE U.S. GEOLOGICAL SURVEY MODULAR THREE-DIMENSIONAL GROUND-WATER FLOW MODEL

by

David W. Pollock

Abstract

The computer program MODPATH-PLOT graphically displays results generated by the three-dimensional particle tracking program MODPATH. A new version of the computer program MODPATH-PLOT has been developed that uses the Graphical Kernel System (GKS) to generate graphical output. The Graphical Kernel System is a set of standardized graphics functions adopted by the American National Standards Institute (ANSI). GKS functions are widely available and can be obtained from many sources. The use of GKS in MODPATH-PLOT improves the program's portability between computers and makes the program accessible to a much broader cross section of ground-water hydrologists. The computer programs MODPATH and MODPATH-PLOT are described in U.S. Geological Survey Open File Report 89-381, titled "Documentation of computer programs to compute and display path lines using results from the U.S. Geological Survey modular three-dimensional finite-difference ground-water flow model" by David W. Pollock.

Introduction

A particle tracking post-processing package (Pollock, 1989) was developed to cc . three-dimensional path lines from output generated from steady-sta[illegible] using the U.S. Geological Survey modular three-dimensional finite-difference ground-water flow model (McDonald and Harbaugh, 1988). The particle tracking post-processing package consists of two computer programs: (1) MODPATH, which calculates path lines, and (2) MODPATH-PLOT, which presents results graphically. The graphics program,

MODPATH-PLOT, was originally developed for use with the DISSPLA[1] graphics subroutine library (Computer Associates, 1981). A new version of the graphics program has been developed that uses GKS graphics subroutines. The Graphical Kernel System is a set of standardized graphics functions adopted by the American National Standards Institute (American National Standards Institute, 1985). GKS software is available from a variety of software vendors and for most computers. As a result, the program is now available to a broader group of ground water hydrologists, especially those using small personal computers.

The primary reason for converting to a GKS-based program was to make the computer program as independent as possible of specific graphics software packages. Most changes to MODPATH-PLOT simply involved a one-to-one replacement of calls to DISSPLA subroutines with calls to standard GKS subroutines performing the identical function. However, some of the changes designed to improve the program's "portability" resulted in minor changes in interactive input and data files. This report describes only those aspects of the program that differ from the original version of MODPATH-PLOT. For a complete description of the computer program, the reader is referred to the documentation provided in the original report (Pollock, 1989).

Changes to Input and Output

Graphical devices

In the original version of MODPATH-PLOT, the choices for graphical output device were explicitly specified in the program source code. Adding or deleting graphical devices to the menu displayed on the computer monitor required that the FORTRAN source code be modified. The GKS version of MODPATH-PLOT was changed to allow the program to accommodate a wide variety of graphical output devices without the need to

[1] The use of trade and firm names in this report is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

modify the FORTRAN source code. The GKS version of MODPATH-PLOT requires an additional data file that contains information about the graphical output devices that are available. The new data file must be named "DEVICE.DAT". The information in DEVICE.DAT is used to generate a customized menu from which the user can select a graphical output device. The file is of format-free structure and contains one line of data for each graphical output device that will appear in the menu, up to a maximum of 10 output devices. Each line contains the following information in the order shown: (1) graphical device type, (2) output medium, and (3) character string prompt to be displayed in the menu. These items are defined as:

(1) graphical device type --	an integer corresponding to a graphical device supported by the GKS software package. Device type numbers for a given device are specific to the GKS package that is used.
(2) output medium --	an integer equal to 0 or 1. If the graphical device is a hard copy device (printer or plotter), this parameter should be set equal to 1. If the graphical device is a computer display monitor, it should be set equal to 0.
(3) character string prompt --	a character string **placed in single quotes**. The string will appear next to one of the choices in the menu.

If DEVICE.DAT contains:

```
1     0     'COMPUTER CONSOLE'
6     1     'PRINTER'
7     1     'PLOTTER'
```

the following menu would be displayed on the monitor:

```
ENTER T.    PE OF GRAPHICS OUTPUT DEVICE:
    1 = COMPUTER CONSOLE
    2 = PRINTER
    3 = PLOTTER
```

Entering the number "2" would select item 2 from the menu and cause graphical device type 6 to be opened by the GKS package. Because the program requires a graphical output device of some kind, the file DEVICE.DAT must contain information for at least one

device. If it does not, a message to that effect is written to the monitor and program execution is terminated. If DEVICE.DAT contains information for only one device, that device is nominated automatically and no prompt is written to the screen. DEVICE.DAT is opened to FORTRAN unit 98. The unit number can be changed in the main program, if necessary.

Plot scaling

The original version of MODPATH-PLOT determined a plot scale that would result in the largest possible plot for the device. The user was then allowed to change the plot scale interactively. The GKS version of MODPATH-PLOT handles plot scaling somewhat differently. When using hard copy output devices such as plotters and printers a user may want to specify the scale to be used in order to produce a plot of a certain size. On video display monitors the goal generally is to make the plot as large as possible. If the chosen device is a video display monitor, the GKS version of MODPATH-PLOT determines a scale that will produce the largest possible plot for that device, and the user is not given an opportunity to adjust the scale of the plot interactively. In that case, no information is printed to the screen concerning plot scaling. If the chosen device is a hard copy device, the program computes the plot scale and then allows the user to adjust the scale interactively, just as in the original version of the program. The program uses the information supplied in DEVICE.DAT to determine if the graphical device is a display monitor or a hard copy unit.

Log file

The GKS version of the program produces an additional output file named "GKS.LOG". This file is required by GKS to log error messages. Information about the the type of graphical device being used is recorded in this file. It also contains any error messages generated by GKS during execution. If desired, the source code can be modified to write additional information to GKS.LOG. If GKS.LOG does not already exist, a file named GKS.LOG is generated automatically by MODPATH-PLOT. The file is opened on

FORTRAN unit number 99. The unit number can be changed in the main program , if necessary.

Program Portability

The GKS standard allows software developers some latitude in implementing graphics subroutines. That fact, combined with the variability in hardware and operating system requirements of individual computers, means that some customizing of the source code is usually necessary. In developing the GKS version of MODPATH-PLOT, an effort was made to anticipate the types of changes that might be necessary. The program was designed so that those changes could be implemented as easily as possible.

Provision for short integers

Some GKS packages (especially those designed for small computers) require that integers passed to the GKS subroutines be short (2 byte) integers rather than standard long (4 byte) integers. However, in MODPATH-PLOT, some integers must be long integers. To accommodate both requirements, those integers in MODPATH-PLOT that must be long integers are explicitly declared using the INTEGER*4 statement. That allows the program to be compiled using short integer as the default, if necessary. When the program is compiled with long integer as the default, all integers are treated as long integers. The use of explicit INTEGER*4 declarations is not standard FORTRAN 77 (ANSI, 1978), however nearly all compilers support INTEGER*4 as an extension to FORTRAN 77.

If MODPATH-PLOT is compiled using the short integer default option, the maximum number of particles cannot exceed 16,000 (even though the master array may be dimensioned large enough to accommodate more than 16,000 particles). To allow for this possibility, an integer variable "LIMIT" is defined in the main program. If "LIMIT = 1", the maximum number of particles is not allowed to exceed 16,000. If "LIMIT = 0", the maximum number of particles is allowed to be as large as can be accommodated by the size

of the master array. If the GKS package requires short integers, LIMIT must be set to 1. Otherwise, LIMIT should be set to 0.

Terminating graphics mode

GKS packages differ in the way the graphics system is closed at the end of the program. Some GKS packages clear the monitor immediately, causing the plot to be erased from the monitor as soon as it is complete. Other GKS packages automatically wait for some input at the keyboard from the user (usually a carriage return) before closing the graphics system. In order to accommodate both types of packages, an integer variable "NWAIT" is set in the main program. When "NWAIT = 1", the program is forced to wait for the user to enter a carriage return before calling the GKS subroutine that closes the graphics system. That allows the plot to remain undisturbed on the monitor for as long as desired. When "NWAIT = 0" the program does not pause before calling the GKS close routines. If the GKS package erases the monitor or overwrites the plot with unwanted text without waiting for a response from the user, NWAIT should be set equal to 1. If the GKS package automatically waits for a response before erasing the monitor, NWAIT should be set to 0. A trial run of the program usually is necessary to determine which setting is appropriate for a given combination of GKS package and computer.

Program testing

The GKS version has been tested with two GKS software packages on two types of computers:

1. PRIME computer using ISSCO-GKS (Computer Associates, 1985).
2. IBM-compatible personal computers using GSS*GKS (Graphic Software Systems, 1985).

ISSCO-GKS on the PRIME computer requires that LIMIT = 0 and NWAIT = 0. GSS*GKS for IBM-compatible personal computers requires that LIMIT = 1 and that NWAIT = 1. The program must be compiled using short integer as the default condition when GSS*GKS is used.

These results are presented only to illustrate the types of changes that may be necessary when customizing MODPATH-PLOT for a specific GKS package. Other GKS packages and other computers may require different combinations of settings or, possibly, additional changes to the program. The version of the code that accompanies this report sets NWAIT=1 and LIMIT=1. These flags will need to be changed for certain combinations of GKS packages and computers.

Summary

The GKS version of MODPATH-PLOT can be used with a wide variety of GKS software packages on a range of computers. This report only describes changes in the GKS version of MODPATH-PLOT. Readers are referred to the original documentation for a complete description of the theory and the use of computer programs MODPATH and MODPATH-PLOT. Although the program was designed to be independent of hardware and the choice of GKS software, some minor changes are usually required to customize the program to a specific GKS package on a specific computer.

References

American National Standards Institute, 1978, Programming Language FORTRAN, ANSI, X3.9-1978, chs. 1-18.

American National Standards Institute, 1985, Computer Graphics -- Graphical Kernel System (GKS) functional description, ANSI X3.124-1985,106 p.

Computer Associates, 1981, DISSPLA, (Display Integrated Software System and Plotting LAnguage) User's Manual.

Computer Associates, 1985, ISSCO-GKS command reference guide, 60 p.

Graphic Software Systems, 1985, GSS*GKS kernel system -- programmer's guide, Graphic Software Systems, Inc., 25117 SW Parkway, Wilsonville, Oregon, 97070.

McDonald, M.G. and A.W. Harbaugh, 1988, A modular three-dimensional finite-difference ground-water flow model, U.S. Geological Survey Techniques of Water Resources Investigations, Book 6, chapter A1, 586 p.

Pollock, David W., 1989, Documentation of computer programs to compute and display pathlines using results from the U.S. Geological Survey modular three-dimensional finite-difference ground-water flow model, U.S. Geological Survey Open File Report 89-381, 188 p.

Appendix - Listing of Computer Code

IMPORTANT -- This version of the code sets NWAIT=1 and LIMIT=1. These flags will need to be reset for some combinations of GKS software and computers.

```
C---GKS Version 1.0    March 1, 1990                                    # A00020
C*********************************************************************# A00030
C                     MODPATH-PLOT (GKS version)                      # A00040
C                   programmed by David W. Pollock                    # A00050
C                                                                       # A00060
C  Plotting program for results from the particle tracking program      # A00070
C  MODPATH. This version of MODPATH-PLOT uses the Graphical Kernel      # A00080
C  System (GKS) graphics subroutines. It is a modification of the       # A00090
C  original version documented in Open File Report 89-381 that used    # A00100
C  the DISSPLA graphics subroutine library. Supplemental documentation  # A00110
C  for this GKS version is available in Open File Report 89-622.        # A00120
C*********************************************************************# A00130
      INTEGER*4 LCIBOU,LCXMX,LCXMN,LCDX,LCYMX,LCYMN,LCDY,LCZBOT,LCZTOP, # A00140
     2LCDZ,LCZMX,LCZMN,LCHEAD,LCLAYC,LCNCON,LCDZCB,LCIBUF,              # A00150
     3LCIUN,LCIBYZ,LCIBXZ,LCIZN,LENA                                    # A00160
C---------------------------------------------------------------------# A00170
C--- REDIMENSION MASTER ARRAY BY CHANGING THE NEXT TWO STATEMENTS ------# A00180
C--- MAKE SURE THAT LENA IS SET EQUAL TO THE LENGTH OF THE A ARRAY -----# A00190
      COMMON A(44000)                                                   # A00200
      LENA=44000                                                        # A00210
C---------------------------------------------------------------------# A00220
C  SET UNIT NUMBERS                                                     # A00230
C                                                                       # A00240
C  IO = UNIT NUMBER OF FILE CONTAINING FILE NAMES AND UNIT NUMBERS      # A00250
      IO=101                                                            # A00260
C  I2 = UNIT NUMBER FOR THE "PATHLINE" FILE                             # A00270
      I2=102                                                            # A00280
C  I3 = UNIT NUMBER FOR THE "ENDPOINT" FILE                             # A00290
      I3=103                                                            # A00300
C  I4 = UNIT NUMBER FOR THE "TIMESERS" FILE                             # A00310
      I4=104                                                            # A00320
C  I7 = UNIT NUMBER FOR THE "SUMMARY.PLT" FILE                          # A00330
      I7=107                                                            # A00340
C  IUGKS = UNIT NUMBER FOR THE GKS LOG FILE "GKS.LOG"                   # A00350
      IUGKS = 99                                                        # A00360
C  IUDEV = UNIT NUMBER FOR THE GKS DEVICE DATA FILE "DEVICE.DAT"        # A00370
      IUDEV = 98                                                        # A00380
C---------------------------------------------------------------------# A00390
C                                                                       # A00400
C  SET INTEGER FLAG "NWAIT"                                             # A00410
C    IF NWAIT=1 MODPATH-PLOT WAITS FOR THE USER TO ENTER <RETURN>       # A00420
C               BEFORE CALLING THE GKS ROUTINE TO CLOSE THE GRAPHICS    # A00430
C               SYSTEM.                                                 # A00440
C    IF NWAIT=0 MODPATH-PLOT CALLS THE GKS ROUTINES TO CLOSE THE        # A00450
C               GRAPHICS SYSTEM AS SOON AS THE PLOT IS FINISHED.        # A00460
C                                                                       # A00470
C  A TRIAL RUN IS USUALLY REQUIRED TO DETERMINE THE APPROPRIATE SETTING # A00480
C  FOR SPECIFIC COMPUTERS.                                              # A00490
C                                                                       # A00500
      NWAIT=1                                                           # A00510
C                                                                       # A00520
C  SET INTEGER FLAG "LIMIT". IF "LIMIT=1", MAXIMUM NUMBER OF PARTICLES  # A00530
C  IS NOT ALLOWED TO EXCEED 16000. IF "LIMIT=0", MAXIMUM NUMBER OF      # A00540
C  PARTICLES IS LIMITED ONLY BY AVAILABLE SPACE IN THE MASTER ARRAY.    # A00550
C                                                                       # A00560
      LIMIT=1                                                           # A00570
```

```
C                                                                       # A00580
      WRITE(*,*)                                                        # A00590
     1' ------------------------------------------------------------'   # A00600
      WRITE(*,*)                                                        # A00610
     1' MODPATH-PLOT -- GKS Graphics Version 1.0 -- March 1, 1990    '  # A00620
      WRITE(*,*)                                                        # A00630
     1' ------------------------------------------------------------'   # A00640
C                                                                       # A00650
C  OPEN STANDARD INPUT AND OUTPUT FILES                                 # A00660
C                                                                       # A00670
      CALL FILES (I0,I1,I7)                                             # A00680
C                                                                       # A00690
C  WRITE VERSION NUMBER AND DATE TO FILE SUMMARY.PLT                    # A00700
C                                                                       # A00710
      WRITE(I7,*) ' '                                                   # A00720
      WRITE(I7,*)                                                       # A00730
     1'MODPATH-PLOT -- GKS Graphics Version 1.0 -- March 1, 1990    '   # A00740
      WRITE(I7,*) ' '                                                   # A00750
C                                                                       # A00760
C  ALLOCATE SPACE FOR ARRAYS                                            # A00770
C                                                                       # A00780
      CALL SPACE(LENA,NCOL,NROW,NLAY,NCBL,IGRID,NZDIM,NUNIT,            # A00790
     1LCIBOU,LCXMX,LCXMN,LCDX,LCYMX,LCYMN,LCDY,LCZBOT,LCZTOP,           # A00800
     2LCDZ,LCZMX,LCZMN,LCHEAD,LCLAYC,LCNCON,LCDZCB,LCIBUF,              # A00810
     3LCIUN,LCIBYZ,LCIBXZ,LCIZN,NPART,I1,I7,LIMIT)                      # A00820
C                                                                       # A00830
      NPART2=2*NPART                                                    # A00840
C                                                                       # A00850
C  TRANSFER CONTROL TO MAIN SUBROUTINE                                  # A00860
C                                                                       # A00870
      CALL DRIVER (A(LCIBOU),A(LCXMX),A(LCXMN),A(LCDX),A(LCYMX),        # A00880
     1A(LCYMN),A(LCDY),A(LCZBOT),A(LCZTOP),A(LCDZ),A(LCZMX),            # A00890
     2A(LCZMN),A(LCHEAD),A(LCLAYC),A(LCNCON),A(LCDZCB),                 # A00900
     3A(LCIBUF),A(LCIUN),A(LCIBYZ),A(LCIBXZ),NCOL,NROW,NLAY,NZDIM,NUNIT,# A00910
     4NCBL,IGRID,I1,I2,I3,I4,I7,NPART,NPART2,A(LCIZN),NWAIT,IUGKS,      # A00920
     5IUDEV)                                                            # A00930
C                                                                       # A00940
      STOP                                                              # A00950
      END                                                               # A00960
C-----END OF ROUTINE----------------------------------------------------# A00970
```

```
C                                                                       # B00010
C---GKS Version 1.0    March 1, 1990                                    # B00020
C***********************************************************************# B00030
C                                                                       # B00040
C                            DRIVER                                     # B00050
C                                                                       # B00060
C  THIS ROUTINE CONTROLS THE OVERALL SEQUENCE OF OPERATIONS AND         # B00070
C  HANDLES MOST OF THE INTERACTIVE I/O                                  # B00080
C                                                                       # B00090
C***********************************************************************# B00100
C                                                                       # B00110
      SUBROUTINE DRIVER (IBOUND,XMX,XMN,DELR,YMX,YMN,DELC,ZBOT,ZTOP,    # B00120
     1DELZ,ZMX,ZMN,HEAD,LAYCON,NCON,DELZCB,IBUFF,IUNIT,IBYZ,IBXZ,       # B00130
     2NCOL,NROW,NLAY,NZDIM,NUNIT,NCBL,IGRID,I1,I2,I3,I4,I7,NPART,       # B00140
     3NPART2,IZPART,NWAIT,IUGKS,IUDEV)                                  # B00150
C                                                                       # B00160
      DIMENSION LAYCON(NLAY),NCON(NLAY),DELR(NCOL),DELC(NROW),DELZ(NLAY)# B00170
     1,DELZCB(NLAY),XMX(NCOL),YMX(NROW),ZTOP(NZDIM),ZBOT(NZDIM),        # B00180
     2HEAD(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY),XMN(NCOL),YMN(NROW),  # B00190
     4IUNIT(NUNIT),IBUFF(NCOL,NROW,NLAY),ZMN(NLAY),                     # B00200
     5ZMX(NLAY),IBYZ(NROW,NLAY),IBXZ(NCOL,NLAY),IZPART(NPART2)          # B00210
C                                                                       # B00220
      DIMENSION ISPLOT(51)                                              # B00230
      DIMENSION XBRDR(5),YBRDR(5)                                       # B00240
C                                                                       # B00250
      CHARACTER*80 FNAME,TITLE                                          # B00260
      CHARACTER*79 MES                                                  # B00270
      DIMENSION LASF(13)                                                # B00280
      DATA LASF/13*1/                                                   # B00290
C                                                                       # B00300
      DO 10 N=1,51                                                      # B00310
10     ISPLOT(N)=0                                                      # B00320
C                                                                       # B00330
C  READ MODPATH DATA SET                                                # B00340
C                                                                       # B00350
      CALL DATIN (LAYCON,NCON,HEAD,XMX,XMN,YMX,YMN,DELR,                # B00360
     1DELC,DELZ,DELZCB,ZTOP,ZBOT,ZMN,ZMX,IBOUND,IUNIT,NCOL,NROW,NLAY,   # B00370
     2NUNIT,NZDIM,IGRID,NCBL,I1,I7)                                     # B00380
C                                                                       # B00390
C                                                                       # B00400
      WRITE (*,*)'ENTER TITLE (80 CHARACTERS OR LESS):'                 # B00410
      READ (*,5000) TITLE                                               # B00420
5000  FORMAT(A)                                                         # B00430
C                                                                       # B00440
      CALL DEVICE (MODEL,KIND,NWAIT,IWAIT,IUDEV,I7)                     # B00450
C                                                                       # B00460
      MES= 'DRAW INTERIOR GRID LINES ?'                                 # B00470
      CALL YESNO (MES,IGL)                                              # B00480
C                                                                       # B00490
      WRITE (*,*)'ENTER THE TYPE OF GRAPH:'                             # B00500
      WRITE (*,*)'    1 = FLOW LINE PLOT'                               # B00510
      WRITE (*,*)'    2 = MAP VIEW OF STARTING LOCATIONS (FORWARD TRACKI# B00520
     1NG)'                                                              # B00530
      WRITE (*,*)'    3 = MAP VIEW OF FINAL LOCATIONS (FORWARD TRACKING)# B00540
     1'                                                                 # B00550
      WRITE (*,*)'    4 = MAP VIEW OF FINAL LOCATIONS (BACKWARD TRACKING# B00560
     1)'                                                                # B00570
      WRITE (*,*)'    5 = TIME SERIES PLOT'                             # B00580
      READ (*,*) ITYPE                                                  # B00590
      IVIEW=1                                                           # B00600
      LAYER=1                                                           # B00610
      IF(ITYPE.EQ.1.OR.ITYPE.EQ.5) THEN                                 # B00620
      WRITE (*,*)'WHAT IS THE ORIENTATION OF THE PLOT ?'                # B00630
      WRITE (*,*)'  1 = MAP VIEW'                                       # B00640
```

```
 WRITE (*,*)'  2 = CROSS SECTION VIEW ALONG A COLUMN'                  # B00650
 WRITE (*,*)'  3 = CROSS SECTION VIEW ALONG A ROW'                     # B00660
 READ (*,*) IVIEW                                                      # B00670
 IIPRJ=0                                                               # B00680
 IF(IVIEW.EQ.1.AND.NLAY.EQ.1) THEN                                     # B00690
 IIPRJ=1                                                               # B00700
 LAYER=1                                                               # B00710
 IPROJ=0                                                               # B00720
 ELSE IF(IVIEW.EQ.2.AND.NCOL.EQ.1) THEN                                # B00730
 IIPRJ=1                                                               # B00740
 JCOL=1                                                                # B00750
 IPROJ=0                                                               # B00760
 ELSE IF(IVIEW.EQ.3.AND.NROW.EQ.1) THEN                                # B00770
 IIPRJ=1                                                               # B00780
 IROW=1                                                                # B00790
 IPROJ=0                                                               # B00800
 END IF                                                                # B00810
 IF(IIPRJ.EQ.0) THEN                                                   # B00820
 WRITE (*,*)'WHAT DATA SHOULD BE PLOTTED ?'                            # B00830
 WRITE (*,*)'   0 = PLOT ALL DATA BY PROJECTION ONTO THE 2D SLICE'     # B00840
 IF(IVIEW.EQ.1) THEN                                                   # B00850
 WRITE (*,*)'   1 = PLOT ONLY THE DATA WITHIN THE LAYER CORRESPONDI    # B00860
1NG TO THE 2D SLICE'                                                   # B00870
 READ (*,*) IPROJ                                                      # B00880
 LAYER=1                                                               # B00890
 IF(IPROJ.GT.0) THEN                                                   # B00900
 WRITE (*,*)'ALONG WHAT LAYER SHOULD THE 2D SLICE BE TAKEN ?'          # B00910
 READ (*,*) LAYER                                                      # B00920
 END IF                                                                # B00930
 ELSE IF (IVIEW.EQ.2) THEN                                             # B00940
 WRITE (*,*)'   1 = PLOT ONLY THE DATA WITHIN THE COLUMN CORRESPOND    # B00950
1ING TO THE CROSS SECTION'                                             # B00960
 READ (*,*) IPROJ                                                      # B00970
 WRITE (*,*)'ENTER THE COLUMN ALONG WHICH CROSS SECTION IS TAKEN:'     # B00980
 READ (*,*) JCOL                                                       # B00990
 ELSE IF (IVIEW.EQ.3) THEN                                             # B01000
 WRITE (*,*)'   1 = PLOT ONLY THE DATA WITHIN THE ROW CORRESPONDING    # B01010
1 TO THE CROSS SECTION'                                                # B01020
 READ (*,*) IPROJ                                                      # B01030
 WRITE (*,*)'ALONG WHAT ROW SHOULD THE CROSS SECTION BE TAKEN ?'       # B01040
 READ (*,*) IROW                                                       # B01050
 END IF                                                                # B01060
 END IF                                                                # B01070
 END IF                                                                # B01080
 IF(ITYPE.EQ.1) THEN                                                   # B01090
 WRITE (*,*)'WERE THE FLOWLINES GENERATED BY FORWARD OR BACKWARD TR    # B01100
1ACKING ?'                                                             # B01110
 WRITE (*,*)'   0 = FORWARD'                                           # B01120
 WRITE (*,*)'   1 = BACKWARD'                                          # B01130
 READ (*,*) IDIR                                                       # B01140
 IF (IDIR.EQ.0) THEN                                                   # B01150
 MES= 'DO YOU WANT TO SKIP OVER PATH LINES THAT DISCHARGE IN ZONE 1    # B01160
1 ?'                                                                   # B01170
 CALL YESNO (MES,ISKIP)                                                # B01180
 END IF                                                                # B01190
 MES= 'DO YOU WANT TO PLOT POINTS AT SPECIFIED TIME INTERVALS ?'       # B01200
 CALL YESNO (MES,IPTS)                                                 # B01210
 MES= 'DO YOU WANT TO STOP DRAWING PATH LINES AT A SPECIFIED TIME?'    # B01220
 CALL YESNO (MES,IANS)                                                 # B01230
 TMAX= 1.0E+30                                                         # B01240
 IF (IANS.EQ.1) THEN                                                   # B01250
 WRITE (*,*)                                                           # B01260
1'ENTER: THE TIME, AND A FACTOR FOR CONVERTING TO UNITS USED IN THE    # B01270
1 MODEL'                                                               # B01280
```

```
      READ (*,*) TMAX,UCFAC                                              # B01290
      TMAX= 1.00001*TMAX*UCFAC                                           # B01300
      END IF                                                             # B01310
      END IF                                                             # B01320
5010   FORMAT(A)                                                         # B01330
      IPLOT=0                                                            # B01340
      IF(ITYPE.EQ.3.OR.ITYPE.EQ.4) THEN                                  # B01350
      MES= 'DO YOU WANT TO PLOT ONLY THOSE POINTS THAT TERMINATE IN ONE  # B01360
     1SPECIFIC ZONE ?'                                                   # B01370
      CALL YESNO (MES,IANS)                                              # B01380
      IF(IANS.EQ.1) THEN                                                 # B01390
      WRITE (*,*) 'ENTER THE ZONE CODE:'                                 # B01400
      READ (*,*) IPLOT                                                   # B01410
      END IF                                                             # B01420
      END IF                                                             # B01430
      WRITE (*,*)'ENTER NAME OF ENDPOINT FILE (<CR>="ENDPOINT"):'        # B01440
      READ (*,5010) FNAME                                                # B01450
      IF (FNAME.EQ.' ') FNAME='ENDPOINT'                                 # B01460
      IUCF=I3                                                            # B01470
      CALL OPNFIL (I3,FNAME,1,I7,0,1)                                    # B01480
      IF(ITYPE.EQ.1) THEN                                                # B01490
      WRITE (*,*)'ENTER NAME OF PATHLINE FILE (<CR>="PATHLINE"):'        # B01500
      READ (*,5010) FNAME                                                # B01510
      IF (FNAME.EQ.' ') FNAME='PATHLINE'                                 # B01520
      IUL=I2                                                             # B01530
      CALL OPNFIL (I2,FNAME,1,I7,0,1)                                    # B01540
      CALL COUNTP (I3,NPART)                                             # B01550
      END IF                                                             # B01560
      IF(ITYPE.EQ.5) THEN                                                # B01570
      WRITE (*,*)'ENTER NAME OF TIMESERIES FILE (<CR>="TIMESERS"):'      # B01580
      READ (*,5010) FNAME                                                # B01590
      IF (FNAME.EQ.' ') FNAME='TIMESERS'                                 # B01600
      CALL OPNFIL (I4,FNAME,1,I7,0,1)                                    # B01610
      WRITE (*,*)'HOW MANY TIME STEPS DO YOU WANT TO PLOT ?'             # B01620
      WRITE (*,*)'  (YOU MAY PLOT UP TO 50 TIME STEPS)'                  # B01630
      WRITE (*,*)'  (TO PLOT ALL OF THE TIME STEPS, ENTER A NEGATIVE NUM# B01640
     1BER)'                                                              # B01650
      READ (*,*) NPSTPS                                                  # B01660
      IF(NPSTPS.GT.0) THEN                                               # B01670
      WRITE (*,*)'ENTER THE TIME STEP NUMBERS THAT YOU WANT TO PLOT:'    # B01680
      READ (*,*) (ISPLOT(N),N=1,NPSTPS)                                  # B01690
      END IF                                                             # B01700
      END IF                                                             # B01710
C                                                                        # B01720
      MES='DO YOU WANT THE PLOT TO INCLUDE THE ENTIRE GRID ?'            # B01730
      CALL YESNO (MES,IALL)                                              # B01740
      IF(IVIEW.EQ.1) THEN                                                # B01750
      JMIN=1                                                             # B01760
      JMAX=NCOL                                                          # B01770
      IMIN=1                                                             # B01780
      IMAX=NROW                                                          # B01790
      IF (IALL.EQ.0) THEN                                                # B01800
      WRITE (*,*)'ENTER GRID COORDINATES:'                               # B01810
      WRITE (*,*)'     MINIMUM COLUMN VALUE,  MAXIMUM COLUMN VALUE'      # B01820
      READ (*,*) JMIN,JMAX                                               # B01830
      IF (JMAX.LT.1.OR.JMAX.GT.NCOL) JMAX=NCOL                           # B01840
      WRITE (*,*)'     MINIMUM ROW VALUE,  MAXIMUM ROW VALUE'            # B01850
      READ (*,*) IMIN,IMAX                                               # B01860
      IF (IMAX.LT.1.OR.IMAX.GT.NROW) IMAX=NROW                           # B01870
      END IF                                                             # B01880
      XMIN= XMN(JMIN)                                                    # B01890
      XMAX= XMX(JMAX)                                                    # B01900
      YMIN= YMN(IMAX)                                                    # B01910
      YMAX= YMX(IMIN)                                                    # B01920
```

```
      VEX=1.0E+0                                                        # B01930
      ELSE IF(IVIEW.EQ.2) THEN                                          # B01940
      IMIN=1                                                            # B01950
      IMAX=NROW                                                         # B01960
      KMIN=1                                                            # B01970
      KMAX=NLAY                                                         # B01980
      IF (IALL.EQ.0) THEN                                               # B01990
      WRITE (*,*)'ENTER GRID COORDINATES:'                              # B02000
      WRITE (*,*)'     MINIMUM ROW VALUE,  MAXIMUM ROW VALUE'           # B02010
      READ (*,*) IMIN,IMAX                                              # B02020
      IF (IMAX.LT.1.OR.IMAX.GT.NROW) IMAX=NROW                          # B02030
      WRITE (*,*)'     MINIMUM LAYER VALUE,  MAXIMUM LAYER VALUE'       # B02040
      READ (*,*) KMIN,KMAX                                              # B02050
      IF (KMAX.LT.1.OR.KMAX.GT.NLAY) KMAX=NLAY                          # B02060
      END IF                                                            # B02070
      WRITE (*,*)'WHAT IS THE VERTICAL EXAGGERATION ?'                  # B02080
      READ (*,*) VEX                                                    # B02090
      DO 20 N=1,NLAY                                                    # B02100
      ZMN(N)=VEX*ZMN(N)                                                 # B02110
20    ZMX(N)=VEX*ZMX(N)                                                 # B02120
      YMIN= YMN(IMAX)                                                   # B02130
      YMAX= YMX(IMIN)                                                   # B02140
      ZMIN= ZMN(KMAX)                                                   # B02150
      ZMAX= ZMX(KMIN)                                                   # B02160
      CALL NEWIB (2,IBOUND,IBYZ,NCOL,NROW,NLAY,NROW,JCOL,HEAD)          # B02170
      ELSE IF(IVIEW.EQ.3) THEN                                          # B02180
      JMIN=1                                                            # B02190
      JMAX=NCOL                                                         # B02200
      KMIN=1                                                            # B02210
      KMAX=NLAY                                                         # B02220
      IF (IALL.EQ.0) THEN                                               # B02230
      WRITE (*,*)'ENTER GRID COORDINATES:'                              # B02240
      WRITE (*,*)'     MINIMUM COLUMN VALUE,  MAXIMUM COLUMN VALUE'     # B02250
      READ (*,*) JMIN,JMAX                                              # B02260
      IF (JMAX.LT.1.OR.JMAX.GT.NCOL) JMAX=NCOL                          # B02270
      WRITE (*,*)'     MINIMUM LAYER VALUE,  MAXIMUM LAYER VALUE'       # B02280
      READ (*,*) KMIN,KMAX                                              # B02290
      IF (KMAX.LT.1.OR.KMAX.GT.NLAY) KMAX=NLAY                          # B02300
      END IF                                                            # B02310
      WRITE (*,*)'WHAT IS THE VERTICAL EXAGGERATION ?'                  # B02320
      READ (*,*) VEX                                                    # B02330
      DO 30 N=1,NLAY                                                    # B02340
      ZMN(N)=VEX*ZMN(N)                                                 # B02350
30    ZMX(N)=VEX*ZMX(N)                                                 # B02360
      XMIN= XMN(JMIN)                                                   # B02370
      XMAX= XMX(JMAX)                                                   # B02380
      ZMIN= ZMN(KMAX)                                                   # B02390
      ZMAX= ZMX(KMIN)                                                   # B02400
      CALL NEWIB (3,IBOUND,IBXZ,NCOL,NROW,NLAY,NCOL,IROW,HEAD)          # B02410
      END IF                                                            # B02420
C                                                                       # B02430
C  CHANGE ZO"   "DES IN IBOUND ARRAY                                    # B02440
C                                                                       # B02450
      IF (ITYPE.NE.5) THEN                                              # B02460
      MES= 'DO YOU WANT TO CHANGE ANY OF THE ZONE CODES IN THE IBOUND AR# B02470
     1RAY ?'                                                            # B02480
      CALL YESNO (MES,IANS)                                             # B02490
      IF (IANS.EQ.1) THEN                                               # B02500
40    WRITE (*,*) 'WHAT TYPE OF CHANGE DO YOU WANT TO MAKE ?'           # B02510
      WRITE (*,*) '   1 = CHANGE AN ENTIRE LAYER'                       # B02520
      WRITE (*,*) '   2 = CHANGE AN INDIVIDUAL CELL'                    # B02530
      WRITE (*,*) '   3 = CHANGE ALL CELLS IN A BLOCK OF CELLS'         # B02540
      READ (*,*) IANS                                                   # B02550
      IF(IANS.EQ.1) THEN                                                # B02560
```

```
      WRITE (*,*) 'ENTER THE LAYER NUMBER:'                                  # B02570
      READ (*,*) NLAYER                                                      # B02580
      WRITE (*,*) 'ENTER THE NEW ZONE CODE:'                                 # B02590
      READ (*,*) NUM                                                         # B02600
      DO 50  I=1,NROW                                                        # B02610
      DO 50  J=1,NCOL                                                        # B02620
      ITEMP=IBOUND(J,I,NLAYER)                                               # B02630
      IF(ITEMP.NE.0) THEN                                                    # B02640
      IBOUND(J,I,NLAYER)=NUM                                                 # B02650
      IF(ITEMP.LT.0) IBOUND(J,I,NLAYER)= -IBOUND(J,I,NLAYER)                 # B02660
      END IF                                                                 # B02670
50    CONTINUE                                                               # B02680
      ELSE IF (IANS.EQ.2) THEN                                               # B02690
      WRITE (*,*) 'ENTER THE CELL INDICES:  J  I  K'                         # B02700
      READ (*,*) J,I,K                                                       # B02710
      WRITE (*,*) 'ENTER THE NEW ZONE CODE:'                                 # B02720
      READ (*,*) NUM                                                         # B02730
      ITEMP=IBOUND(J,I,K)                                                    # B02740
      IF(ITEMP.NE.0) THEN                                                    # B02750
      IBOUND(J,I,K)=NUM                                                      # B02760
      IF(ITEMP.LT.0) IBOUND(J,I,K)= -IBOUND(J,I,K)                           # B02770
      END IF                                                                 # B02780
      ELSE IF (IANS.EQ.3) THEN                                               # B02790
      WRITE (*,*) 'ENTER THE BOUNDARIES OF THE BLOCK OF CELLS:'              # B02800
      WRITE (*,*) '    MIN COLUMN, MAX COLUMN, MIN ROW, MAX ROW, MIN LAY#B02810
     1ER, MAX LAYER'                                                         # B02820
      READ (*,*) JJ1,JJ2,II1,II2,KK1,KK2                                    # B02830
      IF (JJ2.LT.1.OR.JJ2.GT.NCOL) JJ2=NCOL                                  # B02840
      IF (II2.LT.1.OR.II2.GT.NROW) II2=NROW                                  # B02850
      IF (KK2.LT.1.OR.KK2.GT.NLAY) KK2=NLAY                                  # B02860
      WRITE (*,*) 'ENTER THE NEW ZONE CODE:'                                 # B02870
      READ (*,*) NUM                                                         # B02880
      DO 60  K=KK1,KK2                                                       # B02890
      DO 60  I=II1,II2                                                       # B02900
      DO 60  J=JJ1,JJ2                                                       # B02910
      ITEMP=IBOUND(J,I,K)                                                    # B02920
      IF(ITEMP.NE.0) THEN                                                    # B02930
      IBOUND(J,I,K)=NUM                                                      # B02940
      IF(ITEMP.LT.0) IBOUND(J,I,K)= -IBOUND(J,I,K)                           # B02950
      END IF                                                                 # B02960
60    CONTINUE                                                               # B02970
      END IF                                                                 # B02980
      MES= 'DO YOU WANT TO CHANGE SOME MORE ZONE CODES ?'                    # B02990
      CALL YESNO (MES,IANS)                                                  # B03000
      IF (IANS.EQ.1) GO TO 40                                                # B03010
      END IF                                                                 # B03020
      END IF                                                                 # B03030
C                                                                            # B03040
      MES= 'DO YOU WANT A COLOR PLOT ?'                                      # B03050
      CALL YESNO (MES,IANS)                                                  # B03060
      NCLR= -1                                                               # B03070
      IF (IANS.EQ.0.AND.T[illegible]EQ.1) THEN                               # B03080
      WRITE (*,*)'WHAT TYPE OF LINE PATTERNS SHOULD BE USED ?'               # B03090
      WRITE (*,*)'   0 = CYCLE THROUGH LINE PATTERNS (SOLID, DASH, DOT)'#B03100
      WRITE (*,*)'   1 = USE A SINGLE LINE PATTERN FOR ALL PATH LINES'       # B03110
      READ (*,*) NNN                                                         # B03120
      IF(NNN.EQ.0) THEN                                                      # B03130
      NCLR= -4                                                               # B03140
      ELSE                                                                   # B03150
      WRITE (*,*)'ENTER THE LINE PATTERN:'                                   # B03160
      WRITE (*,*)'   1 = SOLID'                                              # B03170
      WRITE (*,*)'   2 = DASHED'                                             # B03180
      WRITE (*,*)'   3 = DOTTED'                                             # B03190
      READ (*,*) NNN                                                         # B03200
```

```
      IF(NNN.LT.1.OR.NNN.GT.3) NNN=1                                    # B03210
      NCLR= -NNN                                                        # B03220
      END IF                                                            # B03230
      ELSE IF(IANS.EQ.1) THEN                                           # B03240
      WRITE (*,*)'HOW SHOULD COLORS BE CHOSEN FOR DIFFERENT DATA GROUPS # B03250
     1?'                                                                # B03260
      WRITE (*,*)'   0 = CYCLE THROUGH COLORS'                          # B03270
      WRITE (*,*)'   1 = ALL LINES AND (OR) DATA POINTS PLOTTED IN ONE  # B03280
     1COLOR'                                                            # B03290
      READ (*,*) NCLR                                                   # B03300
      IF(NCLR.GT.0) THEN                                                # B03310
      WRITE (*,*)'ENTER THE COLOR:'                                     # B03320
      WRITE (*,*)'   1 = BLACK/WHITE'                                   # B03330
      WRITE (*,*)'   2 = RED'                                           # B03340
      WRITE (*,*)'   3 = GREEN'                                         # B03350
      WRITE (*,*)'   4 = BLUE'                                          # B03360
      READ (*,*) NCLR                                                   # B03370
      IF(NCLR.LT.1.OR.NCLR.GT.4) NCLR=1                                 # B03380
      END IF                                                            # B03390
      END IF                                                            # B03400
C                                                                       # B03410
      MES= 'DRAW THE PAGE BORDER ?'                                     # B03420
      CALL YESNO (MES,IBRDR)                                            # B03430
C                                                                       # B03440
C  OPEN GKS LOG/ERR FILE                                                # B03450
      CALL OPNFIL (IUGKS,'GKS.LOG',4,I7,0,3)                            # B03460
C                                                                       # B03470
C  OPEN GKS                                                             # B03480
C                                                                       # B03490
      CALL GOPKS(IUGKS,1000)                                            # B03500
      CALL GOPWK(1,1,MODEL)                                             # B03510
      CALL GACWK(1)                                                     # B03520
C                                                                       # B03530
C  DETERMINE IF DEVICE HAS COLOR                                        # B03540
C  DETERMIN DIMENSIONS OF DEVICE & SET PAGE SIZE                        # B03550
C                                                                       # B03560
      CALL GQCF(MODEL,IERR,NCOLI,ICOLA,NPCI)                            # B03570
      CALL GQDSP(MODEL,IERDSP,IDCUN,RXM,RYM,NPIXX,NPIXY)                # B03580
      WRITE (IUGKS,9005) MODEL                                          # B03590
9005  FORMAT('OUTPUT DEVICE MODEL =',I9)                                # B03600
      IF (ICOLA.EQ.1) THEN                                              # B03610
      WRITE (IUGKS,*) 'COLOR DEVICE'                                    # B03620
      ELSE IF (ICOLA.EQ.0) THEN                                         # B03630
      WRITE (IUGKS,*) 'MONOCHROME DEVICE'                               # B03640
      END IF                                                            # B03650
      IF(IDCUN.EQ.0) THEN                                               # B03660
      WRITE (IUGKS,*) 'DEVICE COORDINATES ARE IN METERS.'               # B03670
      WRITE (IUGKS,9006) RXM,RYM                                        # B03680
9006  FORMAT('SCREEN DIMENSIONS IN METERS:'/                            # B03690
     1'  X DIMENSION =',F9.6,'  Y DIMENSION =',F9.6)                    # B03700
      ELSE IF (IDCUN.EQ.1) THEN                                         # B03710
      WRITE (IUGKS,*) 'DEVICE COORDINATES ARE IN ARBITRARY SCREEN COORDI# B03720
     1NATES.'                                                           # B03730
      WRITE (IUGKS,9007) RXM,RYM                                        # B03740
9007  FORMAT('SCREEN DIMENSIONS IN DEVICE COORDINATES:'/                # B03750
     1'  X DIMENSION =',F10.3,'   Y DIMENSION =',F10.3)                 # B03760
      END IF                                                            # B03770
      WRITE (IUGKS,9008) NPIXX,NPIXY                                    # B03780
9008  FORMAT(I7,' PIXELS IN X DIRECTION'/                               # B03790
     1I7,' PIXELS IN Y DIRECTION')                                      # B03800
      CNVICH=39.3701                                                    # B03810
C IF DEVICE COORDINATES ARE IN METERS, CONVERT PAGE SIZE TO INCHES      # B03820
      IF(IDCUN.EQ.0) THEN                                               # B03830
      PLONG=RXM*CNVICH                                                  # B03840
```

```
      PSHORT=RYM*CNVICH                                                  # B03850
C OTHERWISE, SET PAGE DIMENSIONS TO THEIR DIMENSIONS IN THE              # B03860
C DEVICE COORDINATES RETURNED BY THE GKS QUERY.                          # B03870
      ELSE                                                               # B03880
      PLONG=RXM                                                          # B03890
      PSHORT=RYM                                                         # B03900
      END IF                                                             # B03910
C  IF DEVICE HAS COLOR, SET THE COLORS TO RED, GREEN, AND BLUE           # B03920
      IF (ICOLA.EQ.1) THEN                                               # B03930
      CALL GSCR(MODEL,2,1.0,0.0,0.0)                                     # B03940
      CALL GSCR(MODEL,3,0.0,1.0,0.0)                                     # B03950
      CALL GSCR(MODEL,4,0.0,0.0,1.0)                                     # B03960
      END IF                                                             # B03970
C  DETERMINE THE SIZE OF THE PLOT, AND SET SCALING FACTORS.              # B03980
      IF(IVIEW.EQ.1) THEN                                                # B03990
      CALL SCPLOT (PLONG,PL,PSHORT,PS,IPAGE,XMIN,XMAX,YMIN,YMAX,XL,      # B04000
     1XR,YB,YT,JUNITS,KIND)                                              # B04010
      ELSE IF(IVIEW.EQ.2) THEN                                           # B04020
      CALL SCPLOT (PLONG,PL,PSHORT,PS,IPAGE,YMIN,YMAX,ZMIN,ZMAX,XL,      # B04030
     1XR,YB,YT,JUNITS,KIND)                                              # B04040
      ELSE IF(IVIEW.EQ.3) THEN                                           # B04050
      CALL SCPLOT (PLONG,PL,PSHORT,PS,IPAGE,XMIN,XMAX,ZMIN,ZMAX,XL,      # B04060
     1XR,YB,YT,JUNITS,KIND)                                              # B04070
      END IF                                                             # B04080
C  ADD SPACE FOR A BORDER AROUND THE ACTIVE DRAWING RECTANGLE FOR THE    # B04090
C  PLOT. TITLE AND SCALE BAR WILL BE PUT IN THE BORDER REGION AT BOTTOM  # B04100
C  OF PAGE.                                                              # B04110
      WCDC= (XR-XL)/PL                                                   # B04120
      XLL=XL-0.068*WCDC*PLONG                                            # B04130
      XRR=XR+0.068*WCDC*PLONG                                            # B04140
      YBB=YB-0.176*WCDC*PSHORT                                           # B04150
      YTT=YT+0.118*WCDC*PSHORT                                           # B04160
      WRITE (IUGKS,*) 'COORDINATES OF DEVICE IN WORLD COORDINATES:'      # B04170
      WRITE(IUGKS,9001) XLL,YBB,XRR,YTT                                  # B04180
9001  FORMAT(' (X LEFT, Y BOTTOM) = (',1PE12.5,' , ',E12.5,')'/          # B04190
     1' (X RIGHT, Y TOP) = (',E12.5,' , ',E12.5,')')                     # B04200
      WRITE (IUGKS,*) 'COORDINATES OF THE ACTIVE PLOT RECTANGLE (WC):'   # B04210
      WRITE(IUGKS,9001) XL,YB,XR,YT                                      # B04220
C  SET SIZE OF ATIVE PLOT WINDOW IN WORLD COORDINATES (WC)               # B04230
      SIZEX=XRR-XLL                                                      # B04240
      SIZEY=YTT-YBB                                                      # B04250
      SIZE=SIZEX                                                         # B04260
      IF(SIZEY.GT.SIZEX) SIZE=SIZEY                                      # B04270
      CALL GSWN(1,XLL,XRR,YBB,YTT)                                       # B04280
      XNC=SIZEX/SIZE                                                     # B04290
      YNC=SIZEY/SIZE                                                     # B04300
C  SET GLOBAL VIEWPORT IN NORMALIZED TRANSFORMED COORDINATES (NTC)       # B04310
      CALL GSVP(1,0.,XNC,0.,YNC)                                         # B04320
C  SET WORKSTATION WINDOW SIZE (NTC)                                     # B04330
      CALL GSWKWN(1,0.,XNC,0.,YNC)                                       # B04340
C  SET WORKSTATION VIEWPORT IN DEVICE COORDINATES (DC)                   # B04350
      CALL GSWKVP(1,0.,RXM,0.,RYM)                                       # B04360
      CALL GSELNT(1)                                                     # B04370
      CALL GSASF(LASF)                                                   # B04380
C  SET TEXT FONT AND CHARACTER EXPANSION FACTOR                          # B04390
      CALL GSTXFP(1,1)                                                   # B04400
      CALL GSCHXP(1.)                                                    # B04410
C                                                                        # B04420
      PX=PL                                                              # B04430
      PY=PS                                                              # B04440
      IF(IBRDR.NE.0) THEN                                                # B04450
      XBRDR(1)=XLL                                                       # B04460
      YBRDR(1)=YBB                                                       # B04470
      XBRDR(2)=XRR                                                       # B04480
```

```
      YBRDR(2)=YBRDR(1)                                                 # B04490
      XBRDR(3)=XBRDR(2)                                                 # B04500
      YBRDR(3)=YTT                                                      # B04510
      XBRDR(4)=XBRDR(1)                                                 # B04520
      YBRDR(4)=YBRDR(3)                                                 # B04530
      XBRDR(5)=XBRDR(1)                                                 # B04540
      YBRDR(5)=YBRDR(1)                                                 # B04550
      CALL GPL(5,XBRDR,YBRDR)                                           # B04560
      END IF                                                            # B04570
C                                                                       # B04580
      IF(IVIEW.EQ.1) THEN                                               # B04590
      CALL DGRID (IMIN,IMAX,JMIN,JMAX,XMN,XMX,YMN,YMX,IBOUND(1,1,LAYER),# B04600
     1 NCOL,NROW,IGL,0)                                                 # B04610
      IF(IUNIT(6).GT.0) CALL PGHB(IMIN,IMAX,JMIN,JMAX,1,NLAY,           # B04620
     1 XMN,XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(6),NCLR,I7,   # B04630
     2 IPROJ,LAYER)                                                     # B04640
      IF(IUNIT(5).GT.0) CALL PDRAIN(IMIN,IMAX,JMIN,JMAX,1,NLAY,         # B04650
     1 XMN,XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(5),NCLR,I7,   # B04660
     2 IPROJ,LAYER)                                                     # B04670
      IF(IUNIT(4).GT.0) CALL PRIVER(IMIN,IMAX,JMIN,JMAX,1,NLAY,         # B04680
     1 XMN,XMX,YMN,YMX,NCOL,NROW,NLAY,IUNIT(4),NCLR,IVIEW,              # B04690
     2 IBUFF,I7)                                                        # B04700
      IF(IUNIT(2).GT.0) CALL PWELLS(IMIN,IMAX,JMIN,JMAX,1,NLAY,         # B04710
     1 XMN,XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(2),NCLR,I7,   # B04720
     2 IPROJ,LAYER)                                                     # B04730
      ELSE IF(IVIEW.EQ.2) THEN                                          # B04740
      CALL DGRID (KMIN,KMAX,IMIN,IMAX,YMN,YMX,ZMN,ZMX,IBYZ,NROW,NLAY,   # B04750
     1IGL,1)                                                            # B04760
      CALL PCBEDS (NCON,NLAY,ZMN,ZMX,IMIN,IMAX,YMN,YMX,NROW,IBYZ,       # B04770
     1 KMIN,KMAX)                                                       # B04780
      IF(IUNIT(6).GT.0) CALL PGHB(IMIN,IMAX,JCOL,JCOL,KMIN,KMAX,XMN,    # B04790
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(6),NCLR,I7,       # B04800
     2 IPROJ,JCOL)                                                      # B04810
      IF(IUNIT(5).GT.0) CALL PDRAIN(IMIN,IMAX,JCOL,JCOL,KMIN,KMAX,XMN,  # B04820
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(5),NCLR,I7,       # B04830
     2 IPROJ,JCOL)                                                      # B04840
      IF(IUNIT(4).GT.0) CALL PRIVER(IMIN,IMAX,JCOL,JCOL,KMIN,KMAX,XMN,  # B04850
     1 XMX,YMN,YMX,NCOL,NROW,NLAY,IUNIT(4),NCLR,IVIEW,IBUFF,I7)         # B04860
      IF(IUNIT(2).GT.0) CALL PWELLS(IMIN,IMAX,JCOL,JCOL,KMIN,KMAX,XMN,  # B04870
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(2),NCLR,I7,       # B04880
     2 IPROJ,JCOL)                                                      # B04890
      ELSE IF(IVIEW.EQ.3) THEN                                          # B04900
      CALL DGRID (KMIN,KMAX,JMIN,JMAX,XMN,XMX,ZMN,ZMX,IBXZ,NCOL,NLAY,   # B04910
     1IGL,0)                                                            # B04920
      CALL PCBEDS (NCON,NLAY,ZMN,ZMX,JMIN,JMAX,XMN,XMX,NCOL,IBXZ,       # B04930
     1 KMIN,KMAX)                                                       # B04940
      IF(IUNIT(6).GT.0) CALL PGHB(IROW,IROW,JMIN,JMAX,KMIN,KMAX,XMN,    # B04950
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(6),NCLR,I7,       # B04960
     5 IPROJ,IROW)                                                      # B04970
      IF(IUNIT(5).GT.0) CALL PDRAIN(IROW,IROW,JMIN,JMAX,KMIN,KMAX,XMN,  # B04980
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(5),NCLR,I7,       # B04990
     5 IPROJ,IROW)                                                      # B05000
      IF(IUNIT(4).GT.0) CALL PRIVER(IROW,IROW,JMIN,JMAX,KMIN,KMAX,XMN,  # B05010
     1 XMX,YMN,YMX,NCOL,NROW,NLAY,IUNIT(4),NCLR,IVIEW,IBUFF,I7)         # B05020
      IF(IUNIT(2).GT.0) CALL PWELLS(IROW,IROW,JMIN,JMAX,KMIN,KMAX,XMN,  # B05030
     1 XMX,YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUNIT(2),NCLR,I7,       # B05040
     5 IPROJ,IROW)                                                      # B05050
      END IF                                                            # B05060
C                                                                       # B05070
      IF(ITYPE.EQ.1.AND.IVIEW.EQ.1) CALL LINES(IUL,IUCF,XMIN,XMAX,      # B05080
     1YMIN,YMAX,ZMN,ZMX,IDIR,IBOUND,NCOL,NROW,NLAY,IVIEW,IPTS,          # B05090
     2IPROJ,LAYER,NCLR,ISKIP,TMAX,IZPART,NPART2)                        # B05100
      IF(ITYPE.EQ.1.AND.IVIEW.EQ.2) CALL LINES(IUL,IUCF,YMIN,YMAX,      # B05110
     1ZMIN,ZMAX,ZMN,ZMX,IDIR,IBOUND,NCOL,NROW,NLAY,IVIEW,IPTS,          # B05120
```

```
     2IPROJ,JCOL,NCLR,ISKIP,TMAX,IZPART,NPART2)                          # B05130
      IF(ITYPE.EQ.1.AND.IVIEW.EQ.3) CALL LINES(IUL,IUCF,XMIN,XMAX,       # B05140
     1ZMN,ZMX,IDIR,IBOUND,NCOL,NROW,NLAY,IVIEW,IPTS,                     # B05150
     2IPROJ,IROW,NCLR,ISKIP,TMAX,IZPART,NPART2)                          # B05160
      IF(ITYPE.EQ.2) CALL POINTS (IUCF,1,0,IMIN,IMAX,JMIN,JMAX,XMN,XMX,  # B05170
     1YMN,YMX,IBOUND,NCOL,NROW,NLAY,IPLOT,NCLR)                          # B05180
      IF(ITYPE.EQ.3) CALL POINTS (IUCF,1,1,IMIN,IMAX,JMIN,JMAX,XMN,XMX,  # B05190
     1YMN,YMX,IBOUND,NCOL,NROW,NLAY,IPLOT,NCLR)                          # B05200
      IF(ITYPE.EQ.4) CALL POINTS (IUCF,0,1,IMIN,IMAX,JMIN,JMAX,XMN,XMX,  # B05210
     1YMN,YMX,IBOUND,NCOL,NROW,NLAY,IPLOT,NCLR)                          # B05220
      IF(ITYPE.EQ.5) THEN                                                # B05230
      IF(IVIEW.EQ.1) THEN                                                # B05240
      CALL PTIMS (I4,XMIN,XMAX,YMIN,YMAX,ZMN,ZMX,                        # B05250
     1NLAY,IVIEW,NPSTPS,ISPLOT,51,IPROJ,LAYER,NCLR)                      # B05260
      ELSE IF (IVIEW.EQ.2) THEN                                          # B05270
      CALL PTIMS (I4,YMIN,YMAX,ZMIN,ZMAX,ZMN,ZMX,                        # B05280
     1NLAY,IVIEW,NPSTPS,ISPLOT,51,IPROJ,JCOL,NCLR)                       # B05290
      ELSE IF (IVIEW.EQ.3) THEN                                          # B05300
      CALL PTIMS (I4,XMIN,XMAX,ZMIN,ZMAX,ZMN,ZMX,                        # B05310
     1NLAY,IVIEW,NPSTPS,ISPLOT,51,IPROJ,IROW,NCLR)                       # B05320
      END IF                                                             # B05330
      END IF                                                             # B05340
      CALL PKCLR(1)                                                      # B05350
      CALL NOTATE (XL,XR,YB,YT,PX,PY,JUNITS,TITLE,VEX,IVIEW)             # B05360
C                                                                        # B05370
C  CLOSE GKS                                                             # B05380
C                                                                        # B05390
      IF (IWAIT.EQ.1) READ(*,'(A)') QUIT                                 # B05400
      CALL GDAWK(1)                                                      # B05410
      CALL GCLWK(1)                                                      # B05420
      CALL GCLKS                                                         # B05430
      RETURN                                                             # B05440
      END                                                                # B05450
C-----END OF ROUTINE----------------------------------------------------# B05460
C                                                                        # C00010
C---GKS Version 1.0    March 1, 1990                                     # C00020
C************************************************************************# C00030
C                                                                        # C00040
C                         POINTS                                         # C00050
C                                                                        # C00060
C  THIS ROUTINE PLOTS POINTS                                             # C00070
C                                                                        # C00080
C************************************************************************# C00090
C                                                                        # C00100
      SUBROUTINE POINTS (IU,IZN,IEPT,IMIN,IMAX,JMIN,JMAX,XMN,XMX,        # C00110
     1YMN,YMX,IBOUND,NCOL,NROW,NLAY,IPLOT,NCLR)                          # C00120
      DIMENSION XMN(NCOL),XMX(NCOL),YMN(NROW),YMX(NROW),                 # C00130
     1 IBOUND(NCOL,NROW,NLAY)                                            # C00140
      IOLD=1                                                             # C00150
      CALL GSMK(1)                                                       # C00160
      CALL GSMKSC(.1)                                                    # C00170
      IF(NCLR.GT.0) THEN                                                 # C00180
      KCLR=NCLR                                                          # C00190
      CALL PKCLR(KCLR)                                                   # C00200
      END IF                                                             # C00210
10     READ(IU,*,END=40) IZL,JLAST,ILAST,KLAST,XLAST,YLAST,ZLAST,ZLLAST,# C00220
     1T,XFRST,YFRST,ZLFRST,JFRST,IFRST,KFRST,IZF                         # C00230
      IZL= ABS(IBOUND(JLAST,ILAST,KLAST))                                # C00240
      IZF= ABS(IBOUND(JFRST,IFRST,KFRST))                                # C00250
      X=XLAST                                                            # C00260
      Y=YLAST                                                            # C00270
      J=JLAST                                                            # C00280
      I=ILAST                                                            # C00290
      K=KLAST                                                            # C00300
```

```
      IZONE=IZF                                                          # C00310
      IF(IZN.EQ.1) IZONE=IZL                                             # C00320
      DLR=XMX(J)-XMN(J)                                                  # C00330
      DLC=YMX(I)-YMN(I)                                                  # C00340
      IF(IEPT.EQ.0) THEN                                                 # C00350
      X=XFRST                                                            # C00360
      Y=YFRST                                                            # C00370
      J=JFRST                                                            # C00380
      I=IFRST                                                            # C00390
      K=KFRST                                                            # C00400
      DLR=(XMX(J)-XMN(J))/4.5                                            # C00410
      DLC=(YMX(I)-YMN(I))/4.5                                            # C00420
      END IF                                                             # C00430
      ISKIP=0                                                            # C00440
      IF(IZONE.LE.1.AND.IEPT.EQ.0) ISKIP=1                               # C00450
      IF(ISKIP.EQ.0) THEN                                                # C00460
      IF(J.LT.JMIN.OR.J.GT.JMAX) GO TO 10                                # C00470
      IF(I.LT.IMIN.OR.I.GT.IMAX) GO TO 10                                # C00480
      IF(NCLR.EQ.0) THEN                                                 # C00490
      KCLR=IZONE                                                         # C00500
      IF(IZONE.NE.IOLD) CALL PKCLR (KCLR)                                # C00510
      END IF                                                             # C00520
      IF(IEPT.EQ.0) THEN                                                 # C00530
      ISYM=IZONE+1                                                       # C00540
20    IF(ISYM.LE.5) GO TO 30                                             # C00550
      ISYM=ISYM-4                                                        # C00560
      GO TO 20                                                           # C00570
30    CONTINUE                                                           # C00580
      CALL SYMBL (X,Y,DLR,DLC,ISYM)                                      # C00590
      ELSE                                                               # C00600
      IPLT=1                                                             # C00610
      IF(IPLOT.GT.0) THEN                                                # C00620
      IF(IBOUND(JLAST,ILAST,KLAST).NE.IPLOT) IPLT=0                      # C00630
      END IF                                                             # C00640
      IF(IPLT.EQ.1) CALL GPM(1,X,Y)                                      # C00650
      END IF                                                             # C00660
      IOLD=IZONE                                                         # C00670
      END IF                                                             # C00680
      GO TO 10                                                           # C00690
40    CONTINUE                                                           # C00700
      KCLR=1                                                             # C00710
      CALL PKCLR(KCLR)                                                   # C00720
      RETURN                                                             # C00730
      END                                                                # C00740
C-----END OF ROUTINE-------------------------------------------------------# C00750
C                                                                        # D00010
C---GKS Version 1.0    March 1, 1990                                     # D00020
C************************************************************************# D00030
C                                                                        # D00040
C                           LINES                                        # D00050
C                                                                        # D00060
C  THIS ROUTINE DRAWS PATH LINES                                         # D00070
C                                                                        # D00080
C************************************************************************# D00090
C                                                                        # D00100
      SUBROUTINE LINES (IU,IUEP,XMIN,XMAX,YMIN,YMAX,ZMN,ZMX,IDIR,       # D00110
     1IBOUND,NCOL,NROW,NLAY,IVIEW,IPTS,IPROJ,NSEC,NCLR,ISKIP,TMAX,      # D00120
     2IZPART,NPART2)                                                    # D00130
      DIMENSION IBOUND(NCOL,NROW,NLAY),ZMN(NLAY),ZMX(NLAY),             # D00140
     1IZPART(NPART2)                                                    # D00150
      DIMENSION X(2),Y(2)                                                # D00160
C                                                                        # D00170
      ISKP=0                                                             # D00180
      CALL EZONES (IBOUND,IZPART,NCOL,NROW,NLAY,NPART2,IUEP)             # D00190
```

```
      IF (IDIR.EQ.0.AND.ISKIP.EQ.1) ISKP=1                              # D00200
C                                                                       # D00210
C  CALCULATE SIZE OF MARKER                                             # D00220
C                                                                       # D00230
      SMARK1=XMAX-XMIN                                                  # D00240
      SMARK2=YMAX-YMIN                                                  # D00250
      IF(SMARK2.GT.SMARK1) THEN                                         # D00260
      SMARK1=SMARK2                                                     # D00270
      ELSE                                                              # D00280
      SMARK2=SMARK1                                                     # D00290
      END IF                                                            # D00300
      SMARK1= 0.007*SMARK1                                              # D00310
      SMARK2= 0.007*SMARK2                                              # D00320
C                                                                       # D00330
      IOLD=1                                                            # D00340
      IF(NCLR.GE.0) THEN                                                # D00350
      KCLR=NCLR                                                         # D00360
      IF(KCLR.EQ.0) KCLR=1                                              # D00370
      CALL PKCLR (KCLR)                                                 # D00380
      END IF                                                            # D00390
      IF (NCLR.LT.-1.AND.NCLR.GT.-4) THEN                               # D00400
      NNN= -NCLR                                                        # D00410
      CALL PKPAT (NNN)                                                  # D00420
      END IF                                                            # D00430
      IC=0                                                              # D00440
      IPC=1                                                             # D00450
10     CONTINUE                                                         # D00460
      IF(IVIEW.EQ.1) THEN                                               # D00470
      READ(IU,*,END=30) IP,X(2),Y(2),ZZL,ZZ,TT,J,I,K                   # D00480
      LL=K                                                              # D00490
      ELSE IF (IVIEW.EQ.2) THEN                                         # D00500
      READ(IU,*,END=30) IP,XX,X(2),ZZL,ZZ,TT,J,I,K                     # D00510
      LL=J                                                              # D00520
      IF(ZZL.GE.0.0) Y(2)= (1.0-ZZL)*ZMN(K) + ZZL*ZMX(K)                # D00530
      IF(ZZL.LT.0.0) Y(2)= (1.0E+0+ZZL)*ZMN(K) - ZZL*ZMX(K+1)           # D00540
      ELSE IF (IVIEW.EQ.3) THEN                                         # D00550
      READ(IU,*,END=30) IP,X(2),YY,ZZL,ZZ,TT,J,I,K                     # D00560
      LL=I                                                              # D00570
      IF(ZZL.GE.0.0) Y(2)= (1.0-ZZL)*ZMN(K) + ZZL*ZMX(K)                # D00580
      IF(ZZL.LT.0.0) Y(2)= (1.0E+0+ZZL)*ZMN(K) - ZZL*ZMX(K+1)           # D00590
      END IF                                                            # D00600
      IF(IC.EQ.0.OR.IP.NE.IPC) THEN                                     # D00610
      IF(IDIR.EQ.1) THEN                                                # D00620
      IZONE= IZPART(2*IP-1)                                             # D00630
      ELSE                                                              # D00640
      IZONE= IZPART(2*IP)                                               # D00650
      END IF                                                            # D00660
      IZONE= ABS(IZONE)                                                 # D00670
      IF(NCLR.EQ.0) THEN                                                # D00680
      IF(IZONE.NE.IOLD.OR.IC.EQ.0) CALL PKCLR (IZONE)                   # D00690
      ELSE IF (NCLR.EQ.-4) THEN                                         # D00700
      IF(IZONE.NE.IOLD.OR.IC.EQ.0) CALL PKPAT (IZONE)                   # D00710
      END IF                                                            # D00720
      IOLD=IZONE                                                        # D00730
      X(1)=X(2)                                                         # D00740
      Y(1)=Y(2)                                                         # D00750
      LSEC=LL                                                           # D00760
      IC=1                                                              # D00770
      IPC=IP                                                            # D00780
      GO TO 10                                                          # D00790
      END IF                                                            # D00800
      IF(X(2).GT.XMAX.OR.X(1).GT.XMAX) GO TO 20                         # D00810
      IF(X(2).LT.XMIN.OR.X(1).LT.XMIN) GO TO 20                         # D00820
      IF(Y(2).GT.YMAX.OR.Y(1).GT.YMAX) GO TO 20                         # D00830
```

```
      IF(Y(2).LT.YMIN.OR.Y(1).LT.YMIN) GO TO 20                          # D00840
      IF (ISKP.EQ.0.OR.IZONE.GT.1) THEN                                  # D00850
      IF (IPROJ.EQ.1) THEN                                               # D00860
      IF (LSEC.EQ.NSEC) THEN                                             # D00870
      IF (ABS(TT).LE.TMAX) THEN                                          # D00880
      CALL GPL(2,X,Y)                                                    # D00890
      IF(TT.LT.0.0.AND.IPTS.EQ.1) CALL SYMBL(X(2),Y(2),SMARK1,SMARK2,6)  # D00900
      END IF                                                             # D00910
      END IF                                                             # D00920
      ELSE IF(IPROJ.EQ.0) THEN                                           # D00930
      IF (ABS(TT).LE.TMAX) THEN                                          # D00940
      CALL GPL(2,X,Y)                                                    # D00950
      IF(TT.LT.0.0.AND.IPTS.EQ.1) CALL SYMBL(X(2),Y(2),SMARK1,SMARK2,6)  # D00960
      END IF                                                             # D00970
      END IF                                                             # D00980
      END IF                                                             # D00990
20    X(1)=X(2)                                                          # D01000
      Y(1)=Y(2)                                                          # D01010
      LSEC=LL                                                            # D01020
      GO TO 10                                                           # D01030
30    CONTINUE                                                           # D01040
      IF (NCLR.LT.0) THEN                                                # D01050
      CALL GSLN(1)                                                       # D01060
      END IF                                                             # D01070
      RETURN                                                             # D01080
      END                                                                # D01090
C-----END OF ROUTINE----------------------------------------------------# D01100
C                                                                        # E00010
C---GKS Version 1.0    March 1, 1990                                     # E00020
C************************************************************************# E00030
C                                                                        # E00040
C                              EZONES                                    # E00050
C                                                                        # E00060
C  THIS ROUTINE READS THE ENDPOINT FILE TO GET THE ZONE CODES            # E00070
C  FOR THE INITIAL AND FINAL PARTICLE LOCATIONS. IT PUTS THE ZONE        # E00080
C  CODES IN THE ARRAY "IZPART"                                           # E00090
C                                                                        # E00100
C************************************************************************ # E00110
C                                                                        # E00120
      SUBROUTINE EZONES (IBOUND,IZPART,NCOL,NROW,NLAY,NPART2,IUEP)       # E00130
      DIMENSION IBOUND(NCOL,NROW,NLAY),IZPART(NPART2)                    # E00140
C                                                                        # E00150
      KOUNT=0                                                            # E00160
10    READ(IUEP,*,END=20 ) IZL,JLAST,ILAST,KLAST,XLAST,YLAST,ZLAST,      # E00170
     1ZLLAST,T,XFRST,YFRST,ZLFRST,JFRST,IFRST,KFRST,IZF                  # E00180
      KOUNT=KOUNT+1                                                      # E00190
      N= 2*KOUNT - 1                                                     # E00200
      IZPART(N)= IBOUND (JFRST,IFRST,KFRST)                              # E00210
      IZPART(N+1)= IBOUND (JLAST,ILAST,KLAST)                            # E00220
      GO TO 10                                                           # E00230
20    CONTINUE                                                           # E00240
      RETURN                                                             # E00250
      END                                                                # E00260
C-----END OF ROUTINE----------------------------------------------------# E00270
C---GKS Version 1.0    March 1, 1990                                     # F00010
C************************************************************************# F00020
C                                                                        # F00030
C                         PWELLS                                         # F00040
C                                                                        # F00050
C  THIS ROUTINE READS WELL DATA AND PLOTS WELLS                          # F00060
C                                                                        # F00070
C************************************************************************# F00080
C                                                                        # F00090
      SUBROUTINE PWELLS (IMIN,IMAX,JMIN,JMAX,KMIN,KMAX,XMN,XMX,         # F00100
```

```
     1YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUWELL,NCLR,I7,IPROJ,ISLICE) # F00110
      DIMENSION XMN(NCOL),XMX(NCOL),YMN(NROW),YMX(NROW),ZMN(NLAY),      # F00120
     1ZMX(NLAY)                                                         # F00130
C                                                                       # F00140
      IV= IVIEW*IPROJ                                                   # F00150
      IF(NCLR.GE.0) THEN                                                # F00160
      CALL PKCLR (2)                                                    # F00170
      END IF                                                            # F00180
      WRITE(I7,5000)                                                    # F00190
5000  FORMAT(' ----------------------------------------------------------# F00200
     1----------------------')                                          # F00210
      WRITE(I7,*) 'WELL DATA BEING READ...'                             # F00220
      READ(IUWELL,5010) MXW,ID                                          # F00230
      READ(IUWELL,5010) NWELL                                           # F00240
5010   FORMAT(2I10)                                                     # F00250
      DO 10 N=1,NWELL                                                   # F00260
      READ(IUWELL,5020) K,I,J,Q,IFACE                                   # F00270
      IF(IFACE.NE.0) GO TO 10                                           # F00280
      IF(I.LT.IMIN.OR.I.GT.IMAX) GO TO 10                               # F00290
      IF(J.LT.JMIN.OR.J.GT.JMAX) GO TO 10                               # F00300
      IF(K.LT.KMIN.OR.K.GT.KMAX) GO TO 10                               # F00310
      IF (IV.EQ.1.AND.K.NE.ISLICE) GO TO 10                             # F00320
      IF (IVIEW.EQ.2.AND.J.NE.ISLICE) GO TO 10                          # F00330
      IF (IVIEW.EQ.3.AND.I.NE.ISLICE) GO TO 10                          # F00340
      IF(IVIEW.EQ.1) THEN                                               # F00350
      X=(XMX(J)+XMN(J))/2.0                                             # F00360
      Y=(YMX(I)+YMN(I))/2.0                                             # F00370
      DX=XMX(J)-XMN(J)                                                  # F00380
      DY=YMX(I)-YMN(I)                                                  # F00390
      ELSE IF(IVIEW.EQ.2) THEN                                          # F00400
      X=(YMX(I)+YMN(I))/2.0                                             # F00410
      Y=(ZMX(K)+ZMN(K))/2.0                                             # F00420
      DX=YMX(I)-YMN(I)                                                  # F00430
      DY=ZMX(K)-ZMN(K)                                                  # F00440
      ELSE IF(IVIEW.EQ.3) THEN                                          # F00450
      X=(XMX(J)+XMN(J))/2.0                                             # F00460
      Y=(ZMX(K)+ZMN(K))/2.0                                             # F00470
      DX=XMX(J)-XMN(J)                                                  # F00480
      DY=ZMX(K)-ZMN(K)                                                  # F00490
      END IF                                                            # F00500
      IF (IVIEW.EQ.1) THEN                                              # F00510
      CALL SYMBL (X,Y,DX,DY,1)                                          # F00520
      ELSE                                                              # F00530
      CALL SYMBL (X,Y,DX,DY,0)                                          # F00540
      END IF                                                            # F00550
10    CONTINUE                                                          # F00560
5020   FORMAT(3I10,F10.0,I10)                                           # F00570
      WRITE(I7,*) '  WELL DATA HAS BEEN READ...'                        # F00580
      IF (NCLR.GE.0) CALL PKCLR (1)                                     # F00590
      RETURN                                                            # F00600
      END                                                               # F00610
C-----END OF ROUTINE----------------------------------------------------# F00620
C                                                                       # G00010
C---GKS Version 1.0    March 1, 1990                                    # G00020
C***********************************************************************# G00030
C                                                                       # G00040
C                              SYMBL                                    # G00050
C                                                                       # G00060
C  THIS ROUTINE DRAWS SYMBOLS                                           # G00070
C                                                                       # G00080
C***********************************************************************# G00090
C                                                                       # G00100
      SUBROUTINE SYMBL (XC,YC,DELR,DELC,ISYM)                           # G00110
      DIMENSION X(20),Y(20)                                             # G00120
```

```
DMIN=DELR                                                    # G00130
IF(DELC.LT.DELR) DMIN=DELC                                   # G00140
IF(ISYM.EQ.0) THEN                                           # G00150
X(1)= XC-0.45*DELR                                           # G00160
Y(1)= YC-0.45*DELC                                           # G00170
X(2)= XC+0.45*DELR                                           # G00180
Y(2)= Y(1)                                                   # G00190
X(3)= X(2)                                                   # G00200
Y(3)= YC+0.45*DELC                                           # G00210
X(4)= X(1)                                                   # G00220
Y(4)= Y(3)                                                   # G00230
X(5)= X(1)                                                   # G00240
Y(5)= Y(1)                                                   # G00250
CALL GPL(5,X,Y)                                              # G00260
ELSE IF(ISYM.EQ.1) THEN                                      # G00270
X(1)= XC-0.1625*DMIN                                         # G00280
Y(1)= YC-0.25*DMIN                                           # G00290
X(2)= XC+0.1625*DMIN                                         # G00300
Y(2)= Y(1)                                                   # G00310
X(3)= XC+0.25*DMIN                                           # G00320
Y(3)= YC-0.1625*DMIN                                         # G00330
X(4)= X(3)                                                   # G00340
Y(4)= YC+0.1625*DMIN                                         # G00350
X(5)= XC+0.1625*DMIN                                         # G00360
Y(5)= YC+0.25*DMIN                                           # G00370
X(6)= XC-0.1625*DMIN                                         # G00380
Y(6)= Y(5)                                                   # G00390
X(7)= XC-0.25*DMIN                                           # G00400
Y(7)= YC+0.1625*DMIN                                         # G00410
X(8)= X(7)                                                   # G00420
Y(8)= YC-0.1625*DMIN                                         # G00430
X(9)= X(1)                                                   # G00440
Y(9)= Y(1)                                                   # G00450
CALL GPL(9,X,Y)                                              # G00460
ELSE IF(ISYM.EQ.2) THEN                                      # G00470
X(1)= XC-0.5*DMIN                                            # G00480
Y(1)= YC+0.5*DMIN                                            # G00490
X(2)= XC+0.5*DMIN                                            # G00500
Y(2)= YC-0.5*DMIN                                            # G00510
CALL GPL(2,X,Y)                                              # G00520
YTEMP=Y(1)                                                   # G00530
Y(1)=Y(2)                                                    # G00540
Y(2)=YTEMP                                                   # G00550
CALL GPL(2,X,Y)                                              # G00560
ELSE IF(ISYM.EQ.3) THEN                                      # G00570
X1= XC-0.5*DMIN                                              # G00580
Y1= YC-0.433*DMIN                                            # G00590
X2= XC+0.5*DMIN                                              # G00600
Y2= Y1                                                       # G00610
X3= XC                                                       # G00620
Y3= YC+0.433*DMIN                                            # G00630
X(1)=X1                                                      # G00640
Y(1)=Y1                                                      # G00650
X(2)=X2                                                      # G00660
Y(2)=Y2                                                      # G00670
CALL GPL(2,X,Y)                                              # G00680
X(1)=X2                                                      # G00690
Y(1)=Y2                                                      # G00700
X(2)=X3                                                      # G00710
Y(2)=Y3                                                      # G00720
CALL GPL(2,X,Y)                                              # G00730
X(1)=X3                                                      # G00740
Y(1)=Y3                                                      # G00750
X(2)=X1                                                      # G00760
```

```
      Y(2)=Y1                                                          # G00770
      CALL GPL(2,X,Y)                                                  # G00780
      ELSE IF (ISYM.EQ.4) THEN                                         # G00790
      X(1)=XC                                                          # G00800
      X(2)=XC                                                          # G00810
      Y(1)=YC-0.5*DMIN                                                 # G00820
      Y(2)=YC+0.5*DMIN                                                 # G00830
      CALL GPL(2,X,Y)                                                  # G00840
      X(1)=XC-0.5*DMIN                                                 # G00850
      X(2)=XC+0.5*DMIN                                                 # G00860
      Y(1)=YC                                                          # G00870
      Y(2)=YC                                                          # G00880
      CALL GPL(2,X,Y)                                                  # G00890
      X(1)=XC-0.5*DMIN                                                 # G00900
      Y(1)=YC-0.5*DMIN                                                 # G00910
      X(2)=XC+0.5*DMIN                                                 # G00920
      Y(2)=Y(1)                                                        # G00930
      CALL GPL(2,X,Y)                                                  # G00940
      X(1)=X(2)                                                        # G00950
      Y(2)=YC+0.5*DMIN                                                 # G00960
      CALL GPL(2,X,Y)                                                  # G00970
      Y(1)=Y(2)                                                        # G00980
      X(2)=XC-0.5*DMIN                                                 # G00990
      CALL GPL(2,X,Y)                                                  # G01000
      X(1)=X(2)                                                        # G01010
      Y(2)=YC-0.5*DMIN                                                 # G01020
      CALL GPL(2,X,Y)                                                  # G01030
      ELSE IF (ISYM.EQ.5) THEN                                         # G01040
      X(1)=XC                                                          # G01050
      X(2)=XC                                                          # G01060
      Y(1)=YC-0.5*DMIN                                                 # G01070
      Y(2)=YC+0.5*DMIN                                                 # G01080
      CALL GPL(2,X,Y)                                                  # G01090
      X(1)=XC-0.5*DMIN                                                 # G01100
      X(2)=XC+0.5*DMIN                                                 # G01110
      Y(1)=YC                                                          # G01120
      Y(2)=YC                                                          # G01130
      CALL GPL(2,X,Y)                                                  # G01140
      ELSE IF (ISYM.EQ.6) THEN                                         # G01150
      X(1)=XC-0.5*DMIN                                                 # G01160
      Y(1)=YC-0.5*DMIN                                                 # G01170
      X(2)=XC+0.5*DMIN                                                 # G01180
      Y(2)=Y(1)                                                        # G01190
      CALL GPL(2,X,Y)                                                  # G01200
      X(1)=X(2)                                                        # G01210
      Y(2)=YC+0.5*DMIN                                                 # G01220
      CALL GPL(2,X,Y)                                                  # G01230
      Y(1)=Y(2)                                                        # G01240
      X(2)=XC-0.5*DMIN                                                 # G01250
      CALL GPL(2,X,Y)                                                  # G01260
      X(1)=X(2)                                                        # G01270
      Y(2)=YC-0.5*DMIN                                                 # G01280
      CALL GPL(2,X,Y)                                                  # G01290
      END IF                                                           # G01300
      RETURN                                                           # G01310
      END                                                              # G01320
C-----END OF ROUTINE-------------------------------------------------# G01330
```

```
C                                                                       # H00010
C---GKS Version 1.0    March 1, 1990                                    # H00020
C***********************************************************************# H00030
C                                                                       # H00040
C                              YESNO                                    # H00050
C                                                                       # H00060
C  THIS ROUTINE PRINTS QUESTION PROMPT REQUIRING Y OR N ANSWER. Y AND N # H00070
C  ARE CONVERTED TO NUMERICAL 0 AND 1 RESPONSE                          # H00080
C                                                                       # H00090
C***********************************************************************# H00100
C                                                                       # H00110
      SUBROUTINE YESNO(MES,IANS)                                        # H00120
      CHARACTER*79 MES                                                  # H00130
      CHARACTER*1 IA                                                    # H00140
      CHARACTER*10 ANSWR                                                # H00150
10    WRITE(*,5000) MES                                                 # H00160
      WRITE(*,*) '   [ Y = YES;     N OR <CR> = NO ]'                   # H00170
      READ(*,'(A10)') ANSWR                                             # H00180
5000  FORMAT(1X,A79)                                                    # H00190
      IA= ' '                                                           # H00200
      DO 20 N=1,10                                                      # H00210
      IF (ANSWR(N:N).NE.' ') THEN                                       # H00220
      IA= ANSWR(N:N)                                                    # H00230
      GO TO 30                                                          # H00240
      END IF                                                            # H00250
20    CONTINUE                                                          # H00260
30    CONTINUE                                                          # H00270
      IANS= -1                                                          # H00280
      IF(IA.EQ.'Y'.OR.IA.EQ.'y') IANS=1                                 # H00290
      IF(IA.EQ.'N'.OR.IA.EQ.'n'.OR.IA.EQ.' ') IANS=0                    # H00300
      IF(IANS.EQ.-1) GO TO 10                                           # H00310
      RETURN                                                            # H00320
      END                                                               # H00330
C-----END OF ROUTINE-------------------------------------------------- # H00340
C                                                                       # I00010
C---GKS Version 1.0    March 1, 1990                                    # I00020
C***********************************************************************# I00030
C                                                                       # I00040
C                          DGRID                                        # I00050
C                                                                       # I00060
C  THIS ROUTINE DRAWS THE GRID                                          # I00070
C                                                                       # I00080
C***********************************************************************# I00090
C                                                                       # I00100
      SUBROUTINE DGRID(IMIN,IMAX,JMIN,JMAX,XMN,XMX,YMN,YMX,IBOUND,NCOL, # I00110
     1NROW,IGL,IXS)                                                     # I00120
      DIMENSION IBOUND(NCOL,NROW),XMN(NCOL),XMX(NCOL),YMN(NROW),        # I00130
     1YMX(NROW)                                                         # I00140
      DIMENSION XS(2),YS(2)                                             # I00150
C                                                                       # I00160
      DO 10 I=IMIN,IMAX                                                 # I00170
      DO 10 J=JMIN,JMAX                                                 # I00180
      ITOP=0                                                            # I00190
      IF(I.GT.1) ITOP=IBOUND(J,I-1)                                     # I00200
      XS(1)= XMN(J)                                                     # I00210
      YS(1)= YMX(I)                                                     # I00220
      XS(2)= XMX(J)                                                     # I00230
      YS(2)=YS(1)                                                       # I00240
      IF (ITOP.EQ.0.AND.IBOUND(J,I).EQ.0) GO TO 10                      # I00250
      IF(ITOP.EQ.0.OR.IBOUND(J,I).EQ.0) THEN                            # I00260
      CALL GSLN(1)                                                      # I00270
      CALL GPL(2,XS,YS)                                                 # I00280
      ELSE IF(IGL.EQ.1.OR.I.EQ.IMIN) THEN                               # I00290
      IF(I.NE.IMIN) CALL GSLN(1)                                        # I00300
```

```
      IF(I.EQ.IMIN) CALL GSLN(2)                                        # I00310
      CALL GPL(2,XS,YS)                                                 # I00320
      END IF                                                            # I00330
10     CONTINUE                                                         # I00340
      DO 20 J=JMIN,JMAX                                                 # I00350
      IBOT=0                                                            # I00360
      IF(IMAX.LT.NROW) IBOT=IBOUND(J,IMAX+1)                            # I00370
      XS(1)= XMN(J)                                                     # I00380
      YS(1)= YMN(IMAX)                                                  # I00390
      XS(2)= XMX(J)                                                     # I00400
      YS(2)=YS(1)                                                       # I00410
      IF(IBOT.EQ.0.AND.IBOUND(J,IMAX).EQ.0) GO TO 20                    # I00420
      IF(IBOT.EQ.0.OR.IBOUND(J,IMAX).EQ.0) THEN                         # I00430
      CALL GSLN(1)                                                      # I00440
      CALL GPL(2,XS,YS)                                                 # I00450
      ELSE                                                              # I00460
      CALL GSLN(2)                                                      # I00470
      CALL GPL(2,XS,YS)                                                 # I00480
      END IF                                                            # I00490
20     CONTINUE                                                         # I00500
      DO 30 J=JMIN,JMAX                                                 # I00510
      DO 30 I=IMIN,IMAX                                                 # I00520
      JLEFT=0                                                           # I00530
      IF(J.GT.1) JLEFT=IBOUND(J-1,I)                                    # I00540
      XS(1)= XMN(J)                                                     # I00550
      IF(IXS.EQ.1) XS(1)=XMX(J)                                         # I00560
      YS(1)= YMX(I)                                                     # I00570
      XS(2)=XS(1)                                                       # I00580
      YS(2)=YMN(I)                                                      # I00590
      IF(JLEFT.EQ.0.AND.IBOUND(J,I).EQ.0) GO TO 30                      # I00600
      IF(JLEFT.EQ.0.OR.IBOUND(J,I).EQ.0) THEN                           # I00610
      CALL GSLN(1)                                                      # I00620
      CALL GPL(2,XS,YS)                                                 # I00630
      ELSE IF(IGL.EQ.1.OR.J.EQ.JMIN) THEN                               # I00640
      IF(J.NE.JMIN) CALL GSLN(1)                                        # I00650
      IF(J.EQ.JMIN) CALL GSLN(2)                                        # I00660
      CALL GPL(2,XS,YS)                                                 # I00670
      END IF                                                            # I00680
30    CONTINUE                                                          # I00690
      DO 40 I=IMIN,IMAX                                                 # I00700
      JRIGHT=0                                                          # I00710
      IF(JMAX.LT.NCOL) JRIGHT=IBOUND(JMAX+1,I)                          # I00720
      XS(1)= XMX(JMAX)                                                  # I00730
      IF(IXS.EQ.1) XS(1)=XMN(JMAX)                                      # I00740
      YS(1)= YMX(I)                                                     # I00750
      XS(2)=XS(1)                                                       # I00760
      YS(2)=YMN(I)                                                      # I00770
      IF(JRIGHT.EQ.0.AND.IBOUND(JMAX,I).EQ.0) GO TO 40                  # I00780
      IF(JRIGHT.EQ.0.OR.IBOUND(JMAX,I).EQ.0) THEN                       # I00790
      CALL GSLN(1)                                                      # I00800
      CALL GPL(2,XS,YS)                                                 # I00810
      ELSE                                                              # I00820
      CALL GSLN(2)                                                      # I00830
      CALL GPL(2,XS,YS)                                                 # I00840
      END IF                                                            # I00850
40    CONTINUE                                                          # I00860
      CALL GSLN(1)                                                      # I00870
      CALL PKCLR (1)                                                    # I00880
      RETURN                                                            # I00890
      END                                                               # I00900
C-----END OF ROUTINE----------------------------------------------------# I00910
```

```
C                                                                       # J00010
C---GKS Version 1.0    March 1, 1990                                    # J00020
C***********************************************************************# J00030
C                                                                       # J00040
C                          SCPLOT                                       # J00050
C                                                                       # J00060
C  THIS ROUTINE DETERMINES THE SIZE OF THE PLOT AND COMPUTES THE SCALE  # J00070
C  OF THE PLOT                                                          # J00080
C                                                                       # J00090
C***********************************************************************# J00100
C                                                                       # J00110
      SUBROUTINE SCPLOT (PLONG,PL,PSHORT,PS,IPAGE,XMIN,XMAX,YMIN,YMAX,  # J00120
     1XL,XR,YB,YT,JUNITS,KIND)                                          # J00130
      CHARACTER*79 MES                                                  # J00140
      IPAGE=1                                                           # J00150
      IF (KIND.NE.0) THEN                                               # J00160
      WRITE (*,*)'WHAT ARE THE UNITS OF DISTANCE IN THE MODEL ?'        # J00170
      WRITE (*,*)'   1 = FEET'                                          # J00180
      WRITE (*,*)'   2 = METERS'                                        # J00190
      READ (*,*) JUNITS                                                 # J00200
      ELSE                                                              # J00210
      JUNITS=1                                                          # J00220
      END IF                                                            # J00230
      UNITS=12.0                                                        # J00240
      IF(JUNITS.EQ.2) UNITS=39.0                                        # J00250
      PL=(1.0-0.136)*PLONG                                              # J00260
      PS=(1.0-0.294)*PSHORT                                             # J00270
      XLNG=XMAX-XMIN                                                    # J00280
      YLNG=YMAX-YMIN                                                    # J00290
        R=YLNG/XLNG                                                     # J00300
        PSPL=PS/PL                                                      # J00310
        IF(R.LE.PSPL) THEN                                              # J00320
          SCA=XLNG*UNITS/PL                                             # J00330
          YLNG=PSPL*XLNG                                                # J00340
        ELSE                                                            # J00350
          SCA=YLNG*UNITS/PS                                             # J00360
          XLNG=YLNG/PSPL                                                # J00370
        END IF                                                          # J00380
      SCMAX=SCA                                                         # J00390
      IF (KIND.NE.0) THEN                                               # J00400
      WRITE (*,5000) SCMAX                                              # J00410
5000  FORMAT(' MAXIMUM SIZE PLOT IS AT SCALE OF (1:',F11.0,')')         # J00420
      MES= 'IS THIS SCALE ACCEPTABLE ?'                                 # J00430
      CALL YESNO(MES,IANS)                                              # J00440
      ELSE                                                              # J00450
      IANS=1                                                            # J00460
      END IF                                                            # J00470
      IF(IANS.EQ.0) THEN                                                # J00480
10    WRITE (*,*)'ENTER NEW SCALE -- (1:M)'                             # J00490
      WRITE (*,*)'  ENTER: M'                                           # J00500
      READ (*,*) SCA                                                    # J00510
      IF(SCA.LT.SCMAX) THEN                                             # J00520
      WRITE (*,5010) SCMAX                                              # J00530
5010   FORMAT(' VALUE OF M MUST BE LARGER THAN ',F11.0)                 # J00540
      GO TO 10                                                          # J00550
      END IF                                                            # J00560
      XLNG=XLNG*SCA/SCMAX                                               # J00570
      YLNG=YLNG*SCA/SCMAX                                               # J00580
      END IF                                                            # J00590
      XL= ((XMAX+XMIN)/2.) - XLNG/2.0                                   # J00600
      XR= XL + XLNG                                                     # J00610
      YB= ((YMAX+YMIN)/2.) - YLNG/2.0                                   # J00620
      YT= YB +YLNG                                                      # J00630
      CALL GCLRWK (1,0)                                                 # J00640
```

```
      RETURN                                                            # J00650
      END                                                               # J00660
C-----END OF ROUTINE----------------------------------------------------# J00670
C                                                                       # K00010
C---GKS Version 1.0    March 1, 1990                                    # K00020
C***********************************************************************# K00030
C                                                                       # K00040
C                          PRIVER                                       # K00050
C                                                                       # K00060
C  THIS ROUTINE READS RIVER DATA AND PLOTS RIVER CELL LOCATIONS         # K00070
C                                                                       # K00080
C***********************************************************************# K00090
C                                                                       # K00100
      SUBROUTINE PRIVER (IMIN,IMAX,JMIN,JMAX,KMIN,KMAX,XMN,XMX,YMN,    # K00110
     1 YMX,NCOL,NROW,NLAY,IURIV,NCLR,IVIEW,IBUFF,I7)                    # K00120
      DIMENSION XMN(NCOL),XMX(NCOL),YMX(NROW),YMN(NROW),                # K00130
     1 IBUFF(NCOL,NROW,NLAY)                                            # K00140
C                                                                       # K00150
      DO 10 K=1,NLAY                                                    # K00160
      DO 10 I=1,NROW                                                    # K00170
      DO 10 J=1,NCOL                                                    # K00180
10     IBUFF(J,I,K)=0                                                   # K00190
      IF (NCLR.GE.0) THEN                                               # K00200
      CALL PKCLR (4)                                                    # K00210
      END IF                                                            # K00220
      WRITE(I7,5000)                                                    # K00230
5000  FORMAT(' ---------------------------------------------------------# K00240
     1----------------------')                                          # K00250
      WRITE(I7,*) 'RIVER DATA NOW BEING READ...'                        # K00260
      READ(IURIV,5010) MX,IR                                            # K00270
      READ(IURIV,5010) ITMP                                             # K00280
5010   FORMAT(3I10,3F10.0,I10)                                          # K00290
      IF(ITMP.EQ.0) GO TO 30                                            # K00300
      DO 20 N=1,ITMP                                                    # K00310
      READ(IURIV,5010) K,I,J,H,C,RB,IFACE                               # K00320
      IF(I.LT.IMIN.OR.I.GT.IMAX) GO TO 20                               # K00330
      IF(J.LT.JMIN.OR.J.GT.JMAX) GO TO 20                               # K00340
      IF(K.LT.KMIN.OR.K.GT.KMAX) GO TO 20                               # K00350
      IF (IBUFF(J,I,K).EQ.1) GO TO 20                                   # K00360
      IBUFF(J,I,K)=1                                                    # K00370
      IF (IFACE.GT.6) GO TO 20                                          # K00380
      IF (IVIEW.EQ.1) THEN                                              # K00390
      DX=XMX(J)-XMN(J)                                                  # K00400
      DY=YMX(I)-YMN(I)                                                  # K00410
      X= (XMX(J)+XMN(J))/2.0                                            # K00420
      Y= (YMX(I)+YMN(I))/2.0                                            # K00430
      CALL SYMBL (X,Y,DX,DY,0)                                          # K00440
      END IF                                                            # K00450
20    CONTINUE                                                          # K00460
30    CONTINUE                                                          # K00470
      WRITE(I7,*) '  RIVER DATA HAS BEEN READ'                          # K00480
      IF (NCLR.GE.0) CALL PKCLR (1)                                     # K00490
      RETURN                                                            # K00500
      END                                                               # K00510
C-----END OF ROUTINE----------------------------------------------------# K00520
```

```
C                                                                       # L00010
C---GKS Version 1.0    March 1, 1990                                    # L00020
C***********************************************************************# L00030
C                                                                      *# L00040
C                              SPACE                                   *# L00050
C                                                                      *# L00060
C  THIS ROUTINE ALOCATES SPACE IN THE MASTER ARRAY AND COMPUTES         # L00070
C  POINTERS FOR ARRAY LOCATIONS WITHIN THE MASTER ARRAY                 # L00080
C                                                                       # L00090
C***********************************************************************# L00100
C                                                                       # L00110
      SUBROUTINE SPACE(LENA,NCOL,NROW,NLAY,NCBL,IGRID,NZDIM,NUNIT,      # L00120
     1LCIBOU,LCXMX,LCXMN,LCDX,LCYMX,LCYMN,LCDY,LCZBOT,LCZTOP,           # L00130
     2LCDZ,LCZMX,LCZMN,LCHEAD,LCLAYC,LCNCON,LCDZCB,LCIBUF,              # L00140
     3LCIUN,LCIBYZ,LCIBXZ,LCIZN,NPART,IU,I7,LIMIT)                      # L00150
C                                                                       # L00160
      INTEGER*4 LENA,ISUM,ISM1,NDIF                                     # L00170
      INTEGER*4 LCIBOU,LCXMX,LCXMN,LCDX,LCYMX,LCYMN,LCDY,LCZBOT,LCZTOP, # L00180
     2LCDZ,LCZMX,LCZMN,LCHEAD,LCLAYC,LCNCON,LCDZCB,LCIBUF,              # L00190
     3LCIUN,LCIBYZ,LCIBXZ,LCIZN                                         # L00200
C                                                                       # L00210
      NUNIT=8                                                           # L00220
C                                                                       # L00230
C  ENTER DIMENSION DATA                                                 # L00240
C                                                                       # L00250
      READ(IU,5000) NCOL,NROW,NLAY,NCBL,IGRID                           # L00260
5000  FORMAT(8I10)                                                      # L00270
C                                                                       # L00280
      NCRL=NCOL*NROW*NLAY                                               # L00290
C                                                                       # L00300
      IF(IGRID.EQ.1) THEN                                               # L00310
      NZDIM=NLAY                                                        # L00320
      ELSE                                                              # L00330
      NZDIM=NCRL                                                        # L00340
      END IF                                                            # L00350
C                                                                       # L00360
      ISUM=1                                                            # L00370
      LCIBOU=ISUM                                                       # L00380
      ISUM=ISUM+NCRL                                                    # L00390
      LCXMX=ISUM                                                        # L00400
      ISUM=ISUM+NCOL                                                    # L00410
      LCXMN=ISUM                                                        # L00420
      ISUM=ISUM+NCOL                                                    # L00430
      LCDX=ISUM                                                         # L00440
      ISUM=ISUM+NCOL                                                    # L00450
      LCYMX=ISUM                                                        # L00460
      ISUM=ISUM+NROW                                                    # L00470
      LCYMN=ISUM                                                        # L00480
      ISUM=ISUM+NROW                                                    # L00490
      LCDY=ISUM                                                         # L00500
      ISUM=ISUM+NROW                                                    # L00510
      LCZBOT=ISUM                                                       # L00520
      ISUM=ISUM+NZDIM                                                   # L00530
      LCZTOP=ISUM                                                       # L00540
      ISUM=ISUM+NZDIM                                                   # L00550
      LCDZ=ISUM                                                         # L00560
      ISUM=ISUM+NLAY                                                    # L00570
      LCZMX=ISUM                                                        # L00580
      ISUM=ISUM+NLAY                                                    # L00590
      LCZMN=ISUM                                                        # L00600
      ISUM=ISUM+NLAY                                                    # L00610
      LCHEAD=ISUM                                                       # L00620
      ISUM=ISUM+NCRL                                                    # L00630
      LCLAYC=ISUM                                                       # L00640
```

```
      ISUM=ISUM+NLAY                                                    # L00650
      LCNCON=ISUM                                                       # L00660
      ISUM=ISUM+NLAY                                                    # L00670
      LCDZCB=ISUM                                                       # L00680
      ISUM=ISUM+NLAY                                                    # L00690
      LCIBUF=ISUM                                                       # L00700
      ISUM=ISUM + NCRL                                                  # L00710
      LCIUN=ISUM                                                        # L00720
      ISUM=ISUM + NUNIT                                                 # L00730
      LCIBYZ=ISUM                                                       # L00740
      ISUM=ISUM + (NROW*NLAY)                                           # L00750
      LCIBXZ=ISUM                                                       # L00760
      ISUM=ISUM + (NCOL*NLAY)                                           # L00770
      ISM1=ISUM-1                                                       # L00780
      NDIF=LENA-ISM1                                                    # L00790
      IF(LIMIT.EQ.1) THEN                                               # L00800
      IF (NDIF.GT.32000) NDIF=32000                                     # L00810
      END IF                                                            # L00820
      NPART= NDIF/2                                                     # L00830
      IF (NPART.LT.1) THEN                                              # L00840
      WRITE (*,*)                                                       # L00850
     1'THE SIZE OF THE MASTER ARRAY IS NOT LARGE ENOUGH.'               # L00860
      WRITE (*,*) 'DIMENSION THE MASTER ARRAY:   A(LENA)'               # L00870
      WRITE (*,5010) ISM1                                               # L00880
      WRITE (I7,*)                                                      # L00890
     1'THE SIZE OF THE MASTER ARRAY IS NOT LARGE ENOUGH.'               # L00900
      WRITE (I7,*) 'DIMENSION THE MASTER ARRAY:   A(LENA)'              # L00910
      WRITE (I7,5010) ISM1                                              # L00920
5010  FORMAT (' WITH   LENA =',I8,' + 2 x (NUMBER OF PARTICLES)')       # L00930
      STOP                                                              # L00940
      END IF                                                            # L00950
      LCIZN=ISUM                                                        # L00960
      ISUM=ISUM + (2*NPART)                                             # L00970
      ISM1=ISUM-1                                                       # L00980
      WRITE (*,5020) NPART                                              # L00990
5020  FORMAT (' THE MAXIMUM NUMBER OF PARTICLES FOR THIS RUN IS ',I8)   # L01000
      WRITE (I7,5030) NPART                                             # L01010
5030  FORMAT(' MAXIMUM NUMBER OF PARTICLES IS ',I8)                     # L01020
      WRITE (I7,5040) ISM1,LENA                                         # L01030
5040  FORMAT(I8,' ELEMENTS OUT OF ',I8,' USED IN "A" ARRAY')            # L01040
      IF(ISM1.GT.LENA) STOP                                             # L01050
      RETURN                                                            # L01060
      END                                                               # L01070
C-----END OF ROUTINE----------------------------------------------------# L01080
C                                                                       # M00010
C---GKS Version 1.0    March 1, 1990                                    # M00020
C***********************************************************************# M00030
C                                                                      *# M00040
C                              DATIN                                   *# M00050
C                                                                      *# M00060
C  THIS ROUTINE READS IN DATA                                           # M00070
C                                                                       # M00080
C***********************************************************************# M00090
C                                                                       # M00100
      SUBROUTINE DATIN (LAYCON,NCON,HEAD,XMX,XMN,YMX,YMN,DELR,          # M00110
     1DELC,DELZ,DELZCB,ZTOP,ZBOT,ZMN,ZMX,IBOUND,IUNIT,NCOL,NROW,NLAY,   # M00120
     2NUNIT,NZDIM,IGRID,NCBL,I1,I7)                                     # M00130
C                                                                       # M00140
      CHARACTER*16 TEXT                                                 # M00150
      INTEGER*4 KSTP,KPER,NC,NR,K                                       # M00160
      DIMENSION LAYCON(NLAY),NCON(NLAY),DELR(NCOL),DELC(NROW),DELZ(NLAY)# M00170
     1,DELZCB(NLAY),XMX(NCOL),YMX(NROW),ZTOP(NZDIM),ZBOT(NZDIM),        # M00180
     2HEAD(NCOL,NROW,NLAY),IBOUND(NCOL,NROW,NLAY),XMN(NCOL),YMN(NROW),  # M00190
     4IUNIT(NUNIT),ZMN(NLAY),ZMX(NLAY)                                  # M00200
```

```
C                                                                       # M00210
      IU=I1                                                             # M00220
      WRITE (*,*)'READING MODPATH DATA FILES...'                        # M00230
      WRITE(I7,5000)                                                    # M00240
5000  FORMAT(' ------------------------------------------------------------# M00250
     1----------------------')                                          # M00260
      WRITE(I7,5010) NCOL,NROW,NLAY,NCBL                                # M00270
5010  FORMAT(I5,' COLUMNS',I5,' ROWS',I5,' LAYERS',I5,' CONFINING LAYERS# M00280
     1')                                                                # M00290
      WRITE(I7,5020) IGRID                                              # M00300
5020  FORMAT(' IGRID (GRID TYPE CODE) IS',I2)                           # M00310
C                                                                       # M00320
C  READ UNIT NUMBERS FOR INPUT FILES                                    # M00330
C                                                                       # M00340
      READ(IU,5030) (IUNIT(N),N=1,NUNIT)                                # M00350
5030  FORMAT(16I5)                                                      # M00360
      WRITE(I7,5000)                                                    # M00370
      WRITE(I7,5040) (IUNIT(N),N=1,8)                                   # M00380
5040  FORMAT(' IUNIT ARRAY: ',8I4)                                      # M00390
C                                                                       # M00400
C  LAYER TYPE CODES                                                     # M00410
C                                                                       # M00420
      READ(I1,5050) (LAYCON(N),N=1,NLAY)                                # M00430
5050  FORMAT(40I2)                                                      # M00440
      WRITE(I7,5000)                                                    # M00450
      WRITE(I7,*) 'LAYCON (LAYER TYPE CODES):'                          # M00460
      WRITE(I7,5060) (LAYCON(N),N=1,NLAY)                               # M00470
5060   FORMAT(25I3)                                                     # M00480
C                                                                       # M00490
C  CONFINING BED CODES                                                  # M00500
C                                                                       # M00510
      IF(NCBL.GT.0) THEN                                                # M00520
      READ(I1,5050) (NCON(N),N=1,NLAY)                                  # M00530
      ELSE                                                              # M00540
      DO 10 N=1,NLAY                                                    # M00550
10    NCON(N)=0                                                         # M00560
      END IF                                                            # M00570
      IF(NCBL.EQ.0) THEN                                                # M00580
      WRITE(I7,5000)                                                    # M00590
      WRITE(I7,*) 'NO CONFINING LAYERS.  NCON = 0 FOR ALL LAYERS.'      # M00600
      ELSE                                                              # M00610
      WRITE(I7,5000)                                                    # M00620
      WRITE(I7,*) 'NCON (CONFINING LAYER CODES):'                       # M00630
      WRITE(I7,5060) (NCON(N),N=1,NLAY)                                 # M00640
      END IF                                                            # M00650
      IF(NCBL.GT.0) THEN                                                # M00660
      NN=0                                                              # M00670
      DO 20 N=1,NLAY                                                    # M00680
      IF(NCON(N).EQ.0) GO TO 20                                         # M00690
      NN=NN+1                                                           # M00700
      NCON(N)=NN                                                        # M00710
20    CONTINUE                                                          # M00720
      END IF                                                            # M00730
C                                                                       # M00740
C  DELR, GRID SPACING ALONG A ROW (X-DIRECTION)                         # M00750
C                                                                       # M00760
      WRITE(I7,5000)                                                    # M00770
      WRITE(I7,*) 'DELR ARRAY NOW BEING READ...'                        # M00780
      CALL IN1DR (IU,DELR,NCOL,I7)                                      # M00790
C                                                                       # M00800
C     COMPUTE XMAX ARRAY                                                # M00810
C                                                                       # M00820
      XMX(1)=DELR(1)                                                    # M00830
      XMN(1)=0.0                                                        # M00840
```

```
      DO 30 J=2,NCOL                                                    # M00850
      XMN(J)=XMX(J-1)                                                   # M00860
30     XMX(J)=XMX(J-1)+DELR(J)                                          # M00870
C                                                                       # M00880
C  DELC, GRID SPACING ALONG A COLUMN (Y-DIRECTION)                      # M00890
C                                                                       # M00900
      WRITE(I7,5000)                                                    # M00910
      WRITE(I7,*) 'DELC ARRAY NOW BEING READ...'                        # M00920
      CALL IN1DR (IU,DELC,NROW,I7)                                      # M00930
C                                                                       # M00940
C     COMPUTE YMAX ARRAY                                                # M00950
C                                                                       # M00960
      YMX(NROW)=DELC(NROW)                                              # M00970
      YMN(NROW)=0.0                                                     # M00980
      DO 40 I=2,NROW                                                    # M00990
      II=NROW+1-I                                                       # M01000
      YMN(II)=YMX(II+1)                                                 # M01010
40     YMX(II)=YMX(II+1)+DELC(II)                                       # M01020
C                                                                       # M01030
C  READ VERTICAL COORDINATE DATA                                        # M01040
C                                                                       # M01050
      IF (IGRID.EQ.1) THEN                                              # M01060
      WRITE(I7,5000)                                                    # M01070
      WRITE(I7,*) 'DELZ ARRAY NOW BEING READ...'                        # M01080
      CALL IN1DR (IU,DELZ,NLAY,I7)                                      # M01090
      IF(NCBL.GT.0) THEN                                                # M01100
      WRITE(I7,5000)                                                    # M01110
      WRITE(I7,*) 'DELZCB ARRAY NOW BEING READ...'                      # M01120
      CALL IN1DR (IU,DELZCB,NLAY,I7)                                    # M01130
      ELSE                                                              # M01140
      DO 50 N=1,NLAY                                                    # M01150
50    DELZCB(N)=0.0                                                     # M01160
      END IF                                                            # M01170
      READ(IU,5070) ZBL1                                                # M01180
5070  FORMAT(F10.0)                                                     # M01190
      ZBOT(1)=ZBL1                                                      # M01200
      WRITE(I7,5000)                                                    # M01210
      WRITE(I7,5080) ZBOT(1)                                            # M01220
5080  FORMAT(' BOTTOM ELEVATION OF LAYER 1 IS',E13.5)                   # M01230
      ZTOP(1)= ZBOT(1) + DELZ(1)                                        # M01240
      DO 60 K=2,NLAY                                                    # M01250
      ZTOP(K)= ZBOT(K-1) - DELZCB(K-1)                                  # M01260
      ZBOT(K)= ZTOP(K) - DELZ(K)                                        # M01270
60    CONTINUE                                                          # M01280
      ELSE                                                              # M01290
      DO 70 K=1,NLAY                                                    # M01300
      KP= 1 + (K-1)*NCOL*NROW                                           # M01310
      IF(LAYCON(K).NE.1) THEN                                           # M01320
      WRITE(I7,5000)                                                    # M01330
      WRITE(I7,5090) K                                                  # M01340
5090  FORMAT(' TOP ELEVATION OF LAYER',I4,' NOW BEING READ...')         # M01350
      CALL IN2DR (IU,ZTOP(KP),NCOL,NROW,I7)                             # M01360
      END IF                                                            # M01370
      WRITE(I7,5000)                                                    # M01380
      WRITE(I7,5100) K                                                  # M01390
5100  FORMAT(' BOTTOM ELEVATION OF LAYER',I4,' NOW BEING READ...')      # M01400
      CALL IN2DR (IU,ZBOT(KP),NCOL,NROW,I7)                             # M01410
70    CONTINUE                                                          # M01420
      END IF                                                            # M01430
C                                                                       # M01440
C  IBOUND DATA                                                          # M01450
C                                                                       # M01460
      DO 80 K=1,NLAY                                                    # M01470
      WRITE(I7,5000)                                                    # M01480
```

```
      WRITE(I7,5110) K                                                  # M01490
5110  FORMAT(' IBOUND ARRAY FOR LAYER',I4,' NOW BEING READ...')         # M01500
      CALL IN2DI (IU,IBOUND(1,1,K),NCOL,NROW,I7)                        # M01510
80    CONTINUE                                                          # M01520
C                                                                       # M01530
C  READ HEADS                                                           # M01540
C                                                                       # M01550
      DO 90 K=1,NLAY                                                    # M01560
      DO 90 I=1,NROW                                                    # M01570
      DO 90 J=1,NCOL                                                    # M01580
90    HEAD(J,I,K)=0.                                                    # M01590
      IUHED=IUNIT(8)                                                    # M01600
      IF (IUHED.NE.0) THEN                                              # M01610
100    CONTINUE                                                         # M01620
      READ (IUHED,END=110) KSTP,KPER,PERTIM,TOTIM,TEXT,NC,NR,K          # M01630
      IF (TEXT.NE.'            HEAD') THEN                              # M01640
      WRITE (*,*) 'HEAD FILE DOES NOT CONTAIN HEAD DATA.'               # M01650
      WRITE (I7,*) 'HEAD FILE DOES NOT CONTAIN HEAD DATA.'              # M01660
      STOP                                                              # M01670
      END IF                                                            # M01680
      WRITE(I7,5000)                                                    # M01690
      WRITE(I7,5120) K                                                  # M01700
5120  FORMAT(' HEADS NOW BEING READ FOR LAYER',I4)                      # M01710
      READ(IUHED) ((HEAD(J,I,K),J=1,NCOL),I=1,NROW)                     # M01720
      GO TO 100                                                         # M01730
      END IF                                                            # M01740
110    CONTINUE                                                         # M01750
      WRITE(I7,*) 'HEADS HAVE BEEN READ'                                # M01760
      IF(IGRID.EQ.0) THEN                                               # M01770
      DO 140 K=1,NLAY                                                   # M01780
      DELZ(K)=0.0                                                       # M01790
      COUNT=0.0                                                         # M01800
      DO 120 I=1,NROW                                                   # M01810
      DO 120 J=1,NCOL                                                   # M01820
      IF(IBOUND(J,I,K).EQ.0.OR.HEAD(J,I,K).GT.1.0E+29) GO TO 120        # M01830
      COUNT=COUNT+1.0E+0                                                # M01840
      JIK= (K-1)*NCOL*NROW + (I-1)*NCOL + J                             # M01850
      IF(LAYCON(K).NE.1) THEN                                           # M01860
      DELZ(K)=DELZ(K) + (ZTOP(JIK)-ZBOT(JIK))                           # M01870
      ELSE                                                              # M01880
      DELZ(K)=DELZ(K) + (HEAD(J,I,K)-ZBOT(JIK))                         # M01890
      END IF                                                            # M01900
120    CONTINUE                                                         # M01910
      DELZ(K)=DELZ(K)/COUNT                                             # M01920
      DELZCB(K)=0.0                                                     # M01930
      COUNT=0.0                                                         # M01940
      IF(NCON(K).EQ.0) GO TO 140                                        # M01950
      DO 130 I=1,NROW                                                   # M01960
      DO 130 J=1,NCOL                                                   # M01970
      IF(IBOUND(J,I,K).EQ.0) GO TO 130                                  # M01980
      COUNT=COUNT+1.0E+0                                                # M01990
      JIK= (K-1)*NCOL*NROW + (I-1)*NCOL + J                             # M02000
      JIK1=JIK+(NCOL*NROW)                                              # M02010
      DELZCB(K)= DELZCB(K) + (ZBOT(JIK)-ZTOP(JIK1))                     # M02020
130    CONTINUE                                                         # M02030
      DELZCB(K)=DELZCB(K)/COUNT                                         # M02040
140    CONTINUE                                                         # M02050
      ZMX(NLAY)=DELZ(NLAY)                                              # M02060
      ZMN(NLAY)=0.0                                                     # M02070
      DO 150 K=2,NLAY                                                   # M02080
      KK=NLAY+1-K                                                       # M02090
      ZMN(KK)=ZMX(KK+1)+DELZCB(KK)                                      # M02100
      ZMX(KK)=ZMN(KK)+DELZ(KK)                                          # M02110
150    CONTINUE                                                         # M02120
```

```
      END IF                                                            # M02130
      IF (IGRID.EQ.1) THEN                                              # M02140
      IF(LAYCON(1).EQ.1) THEN                                           # M02150
      DELZ(1)=0.0                                                       # M02160
      COUNT=0.0                                                         # M02170
      DO 160 I=1,NROW                                                   # M02180
      DO 160 J=1,NCOL                                                   # M02190
      IF(IBOUND(J,I,1).EQ.0.OR.HEAD(J,I,1).GT.1.0E+29) GO TO 160        # M02200
      COUNT=COUNT+1.0E+0                                                # M02210
      DELZ(1)=DELZ(1) + (HEAD(J,I,1)-ZBOT(1))                           # M02220
160   CONTINUE                                                          # M02230
      DELZ(1)=DELZ(1)/COUNT                                             # M02240
      END IF                                                            # M02250
      ZMX(1)=ZBOT(1) + DELZ(1)                                          # M02260
      ZMN(1)=ZBOT(1)                                                    # M02270
      DO 170 K=2,NLAY                                                   # M02280
      ZMX(K)=ZTOP(K)                                                    # M02290
      ZMN(K)=ZBOT(K)                                                    # M02300
170   CONTINUE                                                          # M02310
      END IF                                                            # M02320
      RETURN                                                            # M02330
      END                                                               # M02340
C-----END OF ROUTINE-------------------------------------------------------# M02350
C                                                                       # N00010
C---GKS Version 1.0    March 1, 1990                                    # N00020
C**********************************************************************# N00030
C                                                                       # N00040
C                              IN2DR                                    # N00050
C                                                                       # N00060
C  THIS SUBROUTINE READS A TWO DIMENSIONAL ARRAY OF REAL NUMBERS.       # N00070
C                                                                       # N00080
C**********************************************************************# N00090
C                                                                       # N00100
      SUBROUTINE IN2DR (IU,X,JJ,II,I7)                                  # N00110
      DIMENSION X(JJ,II)                                                # N00120
      CHARACTER*20 FMT                                                  # N00130
C                                                                       # N00140
      READ(IU,5000) LOCAT,CNSTNT,FMT,IPRN                               # N00150
5000  FORMAT(I10,F10.0,A20,I10)                                         # N00160
      IF(LOCAT.LE.0) THEN                                               # N00170
      DO 10 I=1,II                                                      # N00180
      DO 10 J=1,JJ                                                      # N00190
10    X(J,I)= CNSTNT                                                    # N00200
      WRITE(I7,5010) CNSTNT                                             # N00210
5010  FORMAT('  CONSTANT VALUE OF',E13.5)                              # N00220
      RETURN                                                            # N00230
      END IF                                                            # N00240
C                                                                       # N00250
      IF(FMT.EQ.'                    ') THEN                            # N00260
      DO 20 I=1,II                                                      # N00270
      READ(LOCAT,*) (X(J,I),J=1,JJ)                                     # N00280
20    CONTINUE                                                          # N00290
      ELSE                                                              # N00300
      DO 30 I=1,II                                                      # N00310
      READ(LOCAT,FMT) (X(J,I),J=1,JJ)                                   # N00320
30    CONTINUE                                                          # N00330
      END IF                                                            # N00340
      IF (CNSTNT.GT.0.0.OR.CNSTNT.LT.0.0) THEN                          # N00350
      DO 40 I=1,II                                                      # N00360
      DO 40 J=1,JJ                                                      # N00370
40    X(J,I)= CNSTNT*X(J,I)                                             # N00380
      END IF                                                            # N00390
      IF(IPRN.GT.0) THEN                                                # N00400
      DO 50 I=1,II                                                      # N00410
```

```
      WRITE(I7,5020) (X(J,I),J=1,JJ)                                    # N00420
50    CONTINUE                                                          # N00430
      ELSE                                                              # N00440
      WRITE(I7,*) '  DATA WAS READ BUT NOT PRINTED'                     # N00450
      END IF                                                            # N00460
5020  FORMAT(10E13.5)                                                   # N00470
C                                                                       # N00480
      RETURN                                                            # N00490
      END                                                               # N00500
C-----END OF ROUTINE----------------------------------------------------# N00510
C                                                                       # O00010
C---GKS Version 1.0    March 1, 1990                                    # O00020
C***********************************************************************# O00030
C                                                                       # O00040
C                              IN1DR                                    # O00050
C                                                                       # O00060
C  THIS SUBROUTINE READS A ONE DIMENSIONAL ARRAY OF REAL NUMBERS.       # O00070
C                                                                       # O00080
C***********************************************************************# O00090
C                                                                       # O00100
      SUBROUTINE IN1DR (IU,X,JJ,I7)                                     # O00110
      DIMENSION X(JJ)                                                   # O00120
      CHARACTER*20 FMT                                                  # O00130
C                                                                       # O00140
      READ(IU,5000) LOCAT,CNSTNT,FMT,IPRN                               # O00150
5000  FORMAT(I10,F10.0,A20,I10)                                         # O00160
      IF(LOCAT.LE.0) THEN                                               # O00170
      DO 10 J=1,JJ                                                      # O00180
10    X(J)= CNSTNT                                                      # O00190
      WRITE(I7,5010) CNSTNT                                             # O00200
5010  FORMAT('  CONSTANT VALUE OF',E13.5)                              # O00210
      RETURN                                                            # O00220
      END IF                                                            # O00230
C                                                                       # O00240
      IF(FMT.EQ.'                    ') THEN                            # O00250
      READ(LOCAT,*) (X(J),J=1,JJ)                                       # O00260
      ELSE                                                              # O00270
      READ(LOCAT,FMT) (X(J),J=1,JJ)                                     # O00280
      END IF                                                            # O00290
      IF (CNSTNT.GT.0.0.OR.CNSTNT.LT.0.0) THEN                          # O00300
      DO 20 J=1,JJ                                                      # O00310
20    X(J)= CNSTNT*X(J)                                                 # O00320
      END IF                                                            # O00330
C                                                                       # O00340
      IF(IPRN.GT.0) THEN                                                # O00350
      WRITE(I7,5020) (X(J),J=1,JJ)                                      # O00360
5020  FORMAT(10E13.5)                                                   # O00370
      ELSE                                                              # O00380
      WRITE(I7,*) '  DATA WAS READ BUT NOT PRINTED'                     # O00390
      END IF                                                            # O00400
      RETURN                                                            # O00410
      END                                                               # O00420
C-----END OF ROUTINE----------------------------------------------------# O00430
C                                                                       # P00010
C---GKS Version 1.0    March 1, 1990                                    # P00020
C***********************************************************************# P00030
C                                                                       # P00040
C                              IN2DI                                    # P00050
C                                                                       # P00060
C  THIS SUBROUTINE READS A TWO DIMENSIONAL ARRAY OF INTEGERS.           # P00070
C                                                                       # P00080
C***********************************************************************# P00090
C                                                                       # P00100
      SUBROUTINE IN2DI (IU,N,JJ,II,I7)                                  # P00110
```

```
      DIMENSION N(JJ,II)                                                # P00120
      CHARACTER*20 FMT                                                  # P00130
C                                                                       # P00140
      READ(IU,5000) LOCAT,ICON,FMT,IPRN                                 # P00150
5000  FORMAT(2I10,A20,I10)                                              # P00160
      IF(LOCAT.LE.0) THEN                                               # P00170
      DO 10 I=1,II                                                      # P00180
      DO 10 J=1,JJ                                                      # P00190
10    N(J,I)= ICON                                                      # P00200
      WRITE(I7,5010) ICON                                               # P00210
5010  FORMAT('  CONSTANT VALUE OF',I10)                                 # P00220
      RETURN                                                            # P00230
      END IF                                                            # P00240
C                                                                       # P00250
      IF(FMT.EQ.'                    ') THEN                            # P00260
      DO 20 I=1,II                                                      # P00270
      READ(LOCAT,*) (N(J,I),J=1,JJ)                                     # P00280
20    CONTINUE                                                          # P00290
      ELSE                                                              # P00300
      DO 30 I=1,II                                                      # P00310
      READ(LOCAT,FMT) (N(J,I),J=1,JJ)                                   # P00320
30    CONTINUE                                                          # P00330
      END IF                                                            # P00340
      IF (ICON.NE.0) THEN                                               # P00350
      DO 40 I=1,II                                                      # P00360
      DO 40 J=1,JJ                                                      # P00370
40    N(J,I)= ICON*N(J,I)                                               # P00380
      END IF                                                            # P00390
      IF(IPRN.GT.0) THEN                                                # P00400
      DO 50 I=1,II                                                      # P00410
      WRITE(I7,5020) (N(J,I),J=1,JJ)                                    # P00420
50    CONTINUE                                                          # P00430
      ELSE                                                              # P00440
      WRITE(I7,*) '  DATA WAS READ BUT NOT PRINTED'                     # P00450
      END IF                                                            # P00460
5020  FORMAT(25I5)                                                      # P00470
C                                                                       # P00480
      RETURN                                                            # P00490
      END                                                               # P00500
C-----END OF ROUTINE----------------------------------------------------# P00510
C                                                                       # Q00010
C---GKS Version 1.0    March 1, 1990                                    # Q00020
C*************************************************************************# Q00030
C                                                                       # Q00040
C                              IN1DI                                    # Q00050
C                                                                       # Q00060
C  THIS SUBROUTINE READS A ONE DIMENSIONAL ARRAY OF INTEGERS.           # Q00070
C                                                                       # Q00080
C*************************************************************************# Q00090
C                                                                       # Q00100
      SUBROUTINE IN1DI (IU,N,JJ,I7)                                     # Q00110
      DIMENSION N(JJ)                                                   # Q00120
      CHARACTER*20 FMT                                                  # Q00130
C                                                                       # Q00140
      READ(IU,5000) LOCAT,ICON,FMT,IPRN                                 # Q00150
5000  FORMAT(2I10,A20,I10)                                              # Q00160
      IF(LOCAT.LE.0) THEN                                               # Q00170
      DO 10 J=1,JJ                                                      # Q00180
10    N(J)= ICON                                                        # Q00190
      WRITE(I7,5010) ICON                                               # Q00200
5010  FORMAT('  CONSTANT VALUE OF',I10)                                 # Q00210
      RETURN                                                            # Q00220
      END IF                                                            # Q00230
C                                                                       # Q00240
```

```
      IF(FMT.EQ.'                    ') THEN                              # Q00250
      READ(LOCAT,*) (N(J),J=1,JJ)                                         # Q00260
      ELSE                                                                # Q00270
      READ(LOCAT,FMT) (N(J),J=1,JJ)                                       # Q00280
      END IF                                                              # Q00290
      IF (ICON.NE.0) THEN                                                 # Q00300
      DO 20 J=1,JJ                                                        # Q00310
20    N(J)= ICON*N(J)                                                     # Q00320
      END IF                                                              # Q00330
      IF(IPRN.GT.0) THEN                                                  # Q00340
      WRITE(I7,5020) (N(J),J=1,JJ)                                        # Q00350
      ELSE                                                                # Q00360
      WRITE(I7,*) '  DATA WAS READ BUT NOT PRINTED'                       # Q00370
      END IF                                                              # Q00380
5020  FORMAT(25I5)                                                        # Q00390
C                                                                         # Q00400
      RETURN                                                              # Q00410
      END                                                                 # Q00420
C-----END OF ROUTINE-----------------------------------------------------# Q00430
C                                                                         # R00010
C---GKS Version 1.0    March 1, 1990                                      # R00020
C*************************************************************************# R00030
C                                                                         # R00040
C                           NEWIB                                         # R00050
C                                                                         # R00060
C  THIS ROUTINE FILLS A NEW TWO DIMENSIONAL "IBOUND" ARRAY FOR CROSS      # R00070
C  SECTIONAL SLICE THROUGH THE 3-D GRID                                   # R00080
C                                                                         # R00090
C*************************************************************************# R00100
C                                                                         # R00110
      SUBROUTINE NEWIB (IVIEW,IBOUND,IB,NCOL,NROW,NLAY,NH,L,HEAD)         # R00120
      DIMENSION IBOUND(NCOL,NROW,NLAY),IB(NH,NLAY),                       # R00130
     1 HEAD(NCOL,NROW,NLAY)                                               # R00140
C                                                                         # R00150
      IF(IVIEW.EQ.2) THEN                                                 # R00160
      DO 10 K=1,NLAY                                                      # R00170
      DO 10 I=1,NROW                                                      # R00180
      IB(I,K)= IBOUND(L,I,K)                                              # R00190
      IF(HEAD(L,I,K).GT.1.0E+29) IB(I,K)=0                                # R00200
10    CONTINUE                                                            # R00210
      ELSE IF(IVIEW.EQ.3) THEN                                            # R00220
      DO 20 K=1,NLAY                                                      # R00230
      DO 20 J=1,NCOL                                                      # R00240
      IB(J,K)=IBOUND(J,L,K)                                               # R00250
      IF(HEAD(J,L,K).GT.1.0E+29) IB(J,K)=0                                # R00260
20    CONTINUE                                                            # R00270
      END IF                                                              # R00280
      RETURN                                                              # R00290
      END                                                                 # R00300
C-----END OF ROUTINE-----------------------------------------------------# R00310
C                                                                         # S00010
C---GKS Version 1.0    March 1, 1990                                      # S00020
C*************************************************************************# S00030
C                                                                         # S00040
C                           PCBEDS                                        # S00050
C                                                                         # S00060
C  THIS ROUTINE DRAWS CONFINING LAYERS                                    # S00070
C                                                                         # S00080
C*************************************************************************# S00090
C                                                                         # S00100
      SUBROUTINE PCBEDS (NCON,NLAY,ZMN,ZMX,JMIN,JMAX,XMN,XMX,NH,IB,KMIN,# S00110
     1 KMAX)                                                              # S00120
      DIMENSION ZMN(NLAY),ZMX(NLAY),NCON(NLAY),XMN(NH),XMX(NH),           # S00130
     1 IB(NH,NLAY)                                                        # S00140
```

```
      DIMENSION X(2),Y(2)                                              # S00150
C                                                                      # S00160
      DO 20 K=KMIN,KMAX                                                # S00170
      IF(NCON(K).EQ.0) GO TO 20                                        # S00180
      DO 10 J=JMIN,JMAX                                                # S00190
      IF(IB(J,K).NE.0) THEN                                            # S00200
      X(1)=XMN(J)                                                      # S00210
      X(2)=XMX(J)                                                      # S00220
      Y(1)=ZMN(K)                                                      # S00230
      Y(2)=ZMN(K)                                                      # S00240
      CALL GPL(2,X,Y)                                                  # S00250
      END IF                                                           # S00260
      IF(K.EQ.NLAY) GO TO 20                                           # S00270
      IF(IB(J,K+1).NE.0) THEN                                          # S00280
      X(1)=XMN(J)                                                      # S00290
      X(2)=XMX(J)                                                      # S00300
      Y(1)=ZMX(K+1)                                                    # S00310
      Y(2)=ZMX(K+1)                                                    # S00320
      CALL GPL(2,X,Y)                                                  # S00330
      END IF                                                           # S00340
10    CONTINUE                                                         # S00350
20    CONTINUE                                                         # S00360
      RETURN                                                           # S00370
      END                                                              # S00380
C-----END OF ROUTINE-------------------------------------------------- # S00390
C                                                                      # T00010
C---GKS Version 1.0    March 1, 1990                                   # T00020
C********************************************************************* # T00030
C                                                                      # T00040
C                            NOTATE                                    # T00050
C                                                                      # T00060
C  THIS ROUTINE DRAWS THE SCALE BAR AND WRITES PLOT ANNOTATION         # T00070
C                                                                      # T00080
C********************************************************************* # T00090
C                                                                      # T00100
      SUBROUTINE NOTATE (XMIN,XMAX,YMIN,YMAX,PX,PY,JUNITS,TITLE,VX,    # T00110
     1 IVIEW)                                                          # T00120
      DIMENSION XA(3),YA(3)                                            # T00130
      CHARACTER*5 JU1                                                  # T00140
      CHARACTER*7 JU2                                                  # T00150
      CHARACTER*80 TITLE                                               # T00160
      CHARACTER*25 EXAG                                                # T00170
      CHARACTER*20 STRING                                              # T00180
      CHARACTER*40 CONCAT                                              # T00190
C                                                                      # T00200
      JU1= ' FEET'                                                     # T00210
      JU2= ' METERS'                                                   # T00220
      EXAG= 'VERTICAL EXAGGERATION IS '                                # T00230
      YPY= (YMAX-YMIN)/PY                                              # T00240
      XPX= (XMAX-XMIN)/PX                                              # T00250
      HGHT= 0.015*(XMAX-XMIN)                                          # T00260
      CALL GSCHSP(0.3)                                                 # T00270
      CALL GSCHH(HGHT)                                                 # T00280
      CALL GQCHW (IERR,CWNOW)                                          # T00290
      CWNEW= (XMAX-XMIN)/80.0                                          # T00300
      IF (CWNEW.LT.CWNOW) THEN                                         # T00310
      HGHT= CWNEW*HGHT/CWNOW                                           # T00320
      CALL GSCHH(HGHT)                                                 # T00330
      END IF                                                           # T00340
      XMID= XMIN + 0.6*(XMAX-XMIN)                                     # T00350
      X= 0.2*(XMAX-XMIN)                                               # T00360
      XL10= ALOG10(X)                                                  # T00370
      IXL10= IFIX(XL10)                                                # T00380
      XX= FLOAT(IXL10)                                                 # T00390
```

```
      XMAN= XL10-XX                                                    # T00400
      XXX= 10.0**(XMAN)                                                # T00410
      IXXX= IFIX(XXX)                                                  # T00420
      XINC= 10.0**(IXL10)                                              # T00430
      NINC= IXXX                                                       # T00440
      YP5= YMIN-(YPY*0.20*PY)                                          # T00450
      YP4= YMIN-(YPY*0.05*PY)                                          # T00460
      YP3= YMIN-(YPY*0.058*PY)                                         # T00470
      YP2= YMIN-(YPY*0.083*PY)                                         # T00480
      YP1= YMIN-(YPY*0.133*PY)                                         # T00490
      CALL GTX(XMIN,YP5,TITLE)                                         # T00500
      XA(1)= XMID                                                      # T00510
      XA(2)=XMID                                                       # T00520
      XA(3)=XMID                                                       # T00530
      YA(1)=YP3                                                        # T00540
      YA(2)=YP2                                                        # T00550
      YA(3)=YP2                                                        # T00560
      CALL GPL(3,XA,YA)                                                # T00570
      XA(1)=XMID                                                       # T00580
      YA(1)=YP2                                                        # T00590
      YA(2)=YP2                                                        # T00600
      YA(3)=YP3                                                        # T00610
      DO 10  N=1,NINC                                                  # T00620
      XA(2)=XMID + (XINC*N)                                            # T00630
      XA(3)=XA(2)                                                      # T00640
      CALL GPL(3,XA,YA)                                                # T00650
      XA(1)=XA(2)                                                      # T00660
10    CONTINUE                                                         # T00670
      INUM=0                                                           # T00680
      ZTMP=INUM                                                        # T00690
      CALL NUMCNV(ZTMP,-1,STRING,NCHARS)                               # T00700
      CALL GTX(XMID,YP4,STRING(1:NCHARS))                              # T00710
      XVAL= XMID + XINC                                                # T00720
      CALL NUMCNV(XINC,-1,STRING,NCHARS)                               # T00730
      CALL GTX(XVAL,YP4,STRING(1:NCHARS))                              # T00740
      XVAL= XINC*NINC                                                  # T00750
      CALL NUMCNV(XVAL,-1,STRING,NCHARS)                               # T00760
      XVAL=XVAL + XMID                                                 # T00770
      IF(JUNITS.EQ.1) THEN                                             # T00780
      STRING=STRING(1:NCHARS)//JU1                                     # T00790
      NCHARS=NCHARS+5                                                  # T00800
      CALL GTX(XVAL,YP4,STRING(1:NCHARS))                              # T00810
      ELSE                                                             # T00820
      CONCAT=STRING(1:NCHARS)//JU2                                     # T00830
      NCHARS=NCHARS+7                                                  # T00840
      CALL GTX(XVAL,YP4,CONCAT(1:NCHARS))                              # T00850
      END IF                                                           # T00860
      IF (IVIEW.NE.1) THEN                                             # T00870
      IF(VX.GE.1.0) CALL NUMCNV(VX,1,STRING,NCHARS)                    # T00880
      IF(VX.LT.1.0) CALL NUMCNV(VX,4,STRING,NCHARS)                    # T00890
      CONCAT=EXAG//STRING(1:NCHARS)                                    # T00900
      NCHARS=25+NCHARS                                                 # T00910
      CALL GTX(XMID,YP1,CONCAT(1:NCHARS))                              # T00920
      END IF                                                           # T00930
      YP3S= YP2 + (YP3-YP2)/2.0                                        # T00940
      IF(NINC.LE.3) THEN                                               # T00950
      XINC=XINC/5.0                                                    # T00960
      XA(1)=XMID                                                       # T00970
      YA(1)= YP2                                                       # T00980
      YA(2)=YP2                                                        # T00990
      YA(3)= YP3S                                                      # T01000
      DO 20  N=1,4                                                     # T01010
      XA(2)= XMID + (XINC*N)                                           # T01020
      XA(3)= XA(2)                                                     # T01030
```

```
      CALL GPL(3,XA,YA)                                                 # T01040
      XA(1)=XA(2)                                                       # T01050
20    CONTINUE                                                          # T01060
      END IF                                                            # T01070
      RETURN                                                            # T01080
      END                                                               # T01090
C-----END OF ROUTINE----------------------------------------------------# T01100
C                                                                       # U00010
C---GKS Version 1.0    March 1, 1990                                    # U00020
C***********************************************************************# U00030
C                                                                       # U00040
C                          PTIMS                                        # U00050
C                                                                       # U00060
C  THIS ROUTINE PLOTS TIMESERIES DATA                                   # U00070
C                                                                       # U00080
C***********************************************************************# U00090
C                                                                       # U00100
      SUBROUTINE PTIMS (IU,XMIN,XMAX,YMIN,YMAX,ZMN,ZMX,NLAY,           # U00110
     1 IVIEW,NPSTPS,ISPLOT,NISP,IPROJ,NSEC,NCLR)                        # U00120
      DIMENSION ZMN(NLAY),ZMX(NLAY),ISPLOT(NISP)                        # U00130
C                                                                       # U00140
      NN=1                                                              # U00150
      IF(NPSTPS.LT.0) THEN                                              # U00160
      KSTP=0                                                            # U00170
      NEXT=1                                                            # U00180
      ELSE                                                              # U00190
      KSTP=ISPLOT(NN)                                                   # U00200
      NEXT= ISPLOT(NN+1)                                                # U00210
      END IF                                                            # U00220
      CALL GSMK(1)                                                      # U00230
      CALL GSMKSC(.1)                                                   # U00240
      IF(NCLR.GE.0) THEN                                                # U00250
      KCLR=NCLR                                                         # U00260
      IF(KCLR.EQ.0) KCLR=1                                              # U00270
      CALL PKCLR (KCLR)                                                 # U00280
      END IF                                                            # U00290
10    CONTINUE                                                          # U00300
      IF (IVIEW.EQ.1) THEN                                              # U00310
      READ(IU,*,END=20) KNT,N,J,I,K,XPT,YPT,ZL,ZLL,TIME                 # U00320
      LSEC=K                                                            # U00330
      ELSE IF (IVIEW.EQ.2) THEN                                         # U00340
      READ(IU,*,END=20) KNT,N,J,I,K,XX,XPT,ZL,ZLL,TIME                  # U00350
      LSEC=J                                                            # U00360
      IF(ZLL.GE.0.0) YPT= (1.0-ZLL)*ZMN(K) + ZLL*ZMX(K)                 # U00370
      IF(ZLL.LT.0.0) YPT= (1.0E+0+ZLL)*ZMN(K) - ZLL*ZMX(K+1)            # U00380
      ELSE IF (IVIEW.EQ.3) THEN                                         # U00390
      READ(IU,*,END=20) KNT,N,J,I,K,XPT,YY,ZL,ZLL,TIME                  # U00400
      LSEC=I                                                            # U00410
      IF(ZLL.GE.0.0) YPT= (1.0-ZLL)*ZMN(K) + ZLL*ZMX(K)                 # U00420
      IF(ZLL.LT.0.0) YPT= (1.0E+0+ZLL)*ZMN(K) - ZLL*ZMX(K+1)            # U00430
      END IF                                                            # U00440
      ICK=0                                                             # U00450
      IF(XPT.GT.XMAX.OR.XPT.LT.XMIN) ICK=1                              # U00460
      IF(YPT.GT.YMAX.OR.YPT.LT.YMIN) ICK=1                              # U00470
      IF(IPROJ.EQ.1.AND.LSEC.NE.NSEC) ICK=1                             # U00480
      IF (KNT.EQ.KSTP) THEN                                             # U00490
      IF(ICK.EQ.0) CALL GPM(1,XPT,YPT)                                  # U00500
      GO TO 10                                                          # U00510
      ELSE IF (KNT.EQ.NEXT.AND.KNT.NE.0) THEN                           # U00520
      KSTP=NEXT                                                         # U00530
      IF(NCLR.EQ.0) THEN                                                # U00540
      KCLR=KCLR+1                                                       # U00550
      CALL PKCLR (KCLR)                                                 # U00560
      END IF                                                            # U00570
```

```
      IF(NPSTPS.LT.0) THEN                                              # U00580
      NEXT=KSTP+1                                                       # U00590
      IF(ICK.EQ.0) CALL GPM (1,XPT,YPT)                                 # U00600
      ELSE                                                              # U00610
      NN=NN+1                                                           # U00620
      NEXT= ISPLOT(NN+1)                                                # U00630
      IF(ICK.EQ.0) CALL GPM(1,XPT,YPT)                                  # U00640
      END IF                                                            # U00650
      ELSE IF (KNT.GT.KSTP.AND.NEXT.EQ.0) THEN                          # U00660
      GO TO 20                                                          # U00670
      END IF                                                            # U00680
      GO TO 10                                                          # U00690
20    CONTINUE                                                          # U00700
      RETURN                                                            # U00710
      END                                                               # U00720
C-----END OF ROUTINE----------------------------------------------------# U00730
C                                                                       # V00010
C---GKS Version 1.0    March 1, 1990                                    # V00020
C***********************************************************************# V00030
C                                                                       # V00040
C                           PKCLR                                       # V00050
C                                                                       # V00060
C  THIS ROUTINE SELECTS A PEN COLOR                                     # V00070
C                                                                       # V00080
C***********************************************************************# V00090
C                                                                       # V00100
      SUBROUTINE PKCLR (KCLR)                                           # V00110
      IF(KCLR.LE.4) GO TO 20                                            # V00120
10    KCLR=KCLR-4                                                       # V00130
      IF(KCLR.LE.4) GO TO 20                                            # V00140
      GO TO 10                                                          # V00150
20    CONTINUE                                                          # V00160
      CALL GSTXCI(KCLR)                                                 # V00170
      CALL GSPLCI(KCLR)                                                 # V00180
      CALL GSPMCI(KCLR)                                                 # V00190
      RETURN                                                            # V00200
      END                                                               # V00210
C-----END OF ROUTINE----------------------------------------------------# V00220
C                                                                       # W00010
C---GKS Version 1.0    March 1, 1990                                    # W00020
C***********************************************************************# W00030
C                                                                       # W00040
C                           PKPAT                                       # W00050
C                                                                       # W00060
C  THIS ROUTINE SELECTS A PEN PATTERN                                   # W00070
C                                                                       # W00080
C***********************************************************************# W00090
C                                                                       # W00100
      SUBROUTINE PKPAT (IPAT)                                           # W00110
      IF(IPAT.LE.3) GO TO 20                                            # W00120
10    IPAT=IPAT-3                                                       # W00130
      IF(IPAT.LE.3) GO TO 20                                            # W00140
      GO TO 10                                                          # W00150
20    CONTINUE                                                          # W00160
      IF (IPAT.EQ.1) THEN                                               # W00170
      CALL GSLN(1)                                                      # W00180
      ELSE IF (IPAT.EQ.2) THEN                                          # W00190
      CALL GSLN(2)                                                      # W00200
      ELSE IF (IPAT.EQ.3) THEN                                          # W00210
      CALL GSLN(3)                                                      # W00220
      END IF                                                            # W00230
      RETURN                                                            # W00240
      END                                                               # W00250
C-----END OF ROUTINE----------------------------------------------------# W00260
```

```
C                                                                       # X00010
C---GKS Version 1.0    March 1, 1990                                    # X00020
C***********************************************************************# X00030
C                                                                       # X00040
C                          FILES                                        # X00050
C                                                                       # X00060
C  THIS ROUTINE OPENS FILES                                             # X00070
C                                                                       # X00080
C***********************************************************************# X00090
C                                                                       # X00100
      SUBROUTINE FILES (I0,I1,I7)                                       # X00110
      CHARACTER*80 FNAME                                                # X00120
C                                                                       # X00130
C  CREATE AND OPEN STANDARD OUTPUT AND SCRATCH FILES                    # X00140
C                                                                       # X00150
      FNAME= 'SUMMARY.PLT'                                              # X00160
      CALL OPNFIL (I7,FNAME,4,I7,0,3)                                   # X00170
C                                                                       # X00180
C  OPEN FILE CONTAINING FILE NAMES AND FORTRAN UNITS                    # X00190
C                                                                       # X00200
      WRITE  (*,*) 'ENTER NAME OF FILE CONTAINING NAMES AND UNITS OF DAT# X00210
     1A FILES:'                                                         # X00220
      READ  (*,5000) FNAME                                              # X00230
5000   FORMAT (A)                                                       # X00240
C                                                                       # X00250
      CALL OPNFIL (I0,FNAME,1,I7,0,1)                                   # X00260
C                                                                       # X00270
      KFIRST=0                                                          # X00280
10    READ (I0,*,END=20 ) IU,FNAME                                      # X00290
      IF (KFIRST.EQ.0) I1=IU                                            # X00300
      KFIRST=1                                                          # X00310
C                                                                       # X00320
C  OPEN DATA FILES                                                      # X00330
C                                                                       # X00340
      CALL OPNFIL (IU,FNAME,1,I7,0,1)                                   # X00350
C                                                                       # X00360
      WRITE(I7,5010) IU,FNAME                                           # X00370
5010  FORMAT(' FORTRAN UNIT ',I3,' = ',A60)                             # X00380
      GO TO 10                                                          # X00390
20    CONTINUE                                                          # X00400
      RETURN                                                            # X00410
      END                                                               # X00420
C-----END OF ROUTINE--------------------------------------------------#  X00430
C                                                                       # Y00010
C---GKS Version 1.0    March 1, 1990                                    # Y00020
C*********************************************************************** Y00030
C                                                                       # Y00040
C                           COUNTP                                      # Y00050
C                                                                       # Y00060
C  THIS ROUTINE READS THE ENDPOINT FILE AND COUNTS THE NUMBER OF        # Y00070
C  PARTICLES RECORDED IN THE FILE. IT REWINDS THE FILE WHEN THE         # Y00080
C  COUNT IS FINISHED                                                    # Y00090
C                                                                       # Y00100
C***********************************************************************# Y00110
C                                                                       # Y00120
      SUBROUTINE COUNTP (IU,NPART)                                      # Y00130
      N=0                                                               # Y00140
10    READ (IU,5000,END=20 )                                            # Y00150
5000   FORMAT(I10)                                                      # Y00160
      N=N+1                                                             # Y00170
      GO TO 10                                                          # Y00180
20    CONTINUE                                                          # Y00190
      REWIND (IU)                                                       # Y00200
      IF (N.GT.NPART) THEN                                              # Y00210
```

```
      WRITE (*,5010) N,NPART                                                # Y00220
5010   FORMAT (I6,' PARTICLES IN ENDPOINT FILE (MAXIMUM ALLOWED IS',I6,     # Y00230
     1')')                                                                  # Y00240
      WRITE (*,5020) NPART                                                  # Y00250
5020  FORMAT(' RERUN PROGRAM AND SPECIFY NUMBER OF PARTICLES >= ',I6)       # Y00260
      STOP                                                                  # Y00270
      END IF                                                                # Y00280
      RETURN                                                                # Y00290
      END                                                                   # Y00300
C-----END OF ROUTINE----------------------------------------------------    # Y00310
C                                                                           # Z00010
C---GKS Version 1.0    March 1, 1990                                        # Z00020
C*************************************************************************# Z00030
C                                                                           # Z00040
C                        PDRAIN                                             # Z00050
C                                                                           # Z00060
C  THIS ROUTINE READS DRAIN DATA AND PLOTS DRAINS                           # Z00070
C                                                                           # Z00080
C*************************************************************************# Z00090
C                                                                           # Z00100
      SUBROUTINE PDRAIN (IMIN,IMAX,JMIN,JMAX,KMIN,KMAX,XMN,XMX,            # Z00110
     1YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUDRAI,NCLR,I7,IPROJ,ISLICE)     # Z00120
      DIMENSION XMN(NCOL),XMX(NCOL),YMN(NROW),YMX(NROW),ZMN(NLAY),          # Z00130
     1ZMX(NLAY)                                                             # Z00140
C                                                                           # Z00150
      IV= IVIEW*IPROJ                                                       # Z00160
      IF(NCLR.GE.0) THEN                                                    # Z00170
      CALL PKCLR (2)                                                        # Z00180
      END IF                                                                # Z00190
      WRITE(I7,5000)                                                        # Z00200
5000  FORMAT(' ----------------------------------------------------------# Z00210
     1----------------------')                                              # Z00220
      WRITE(I7,*) 'DRAIN DATA BEING READ...'                                # Z00230
      READ(IUDRAI,5010) MXW,ID                                              # Z00240
      READ(IUDRAI,5010) NDRAI                                               # Z00250
5010   FORMAT(2I10)                                                         # Z00260
      DO 10 N=1,NDRAI                                                       # Z00270
      READ(IUDRAI,5020) K,I,J,H,C,IFACE                                     # Z00280
      IF(IFACE.NE.0) GO TO 10                                               # Z00290
      IF(I.LT.IMIN.OR.I.GT.IMAX) GO TO 10                                   # Z00300
      IF(J.LT.JMIN.OR.J.GT.JMAX) GO TO 10                                   # Z00310
      IF(K.LT.KMIN.OR.K.GT.KMAX) GO TO 10                                   # Z00320
      IF (IV.EQ.1.AND.K.NE.ISLICE) GO TO 10                                 # Z00330
      IF (IVIEW.EQ.2.AND.J.NE.ISLICE) GO TO 10                              # Z00340
      IF (IVIEW.EQ.3.AND.I.NE.ISLICE) GO TO 10                              # Z00350
      IF(IVIEW.EQ.1) THEN                                                   # Z00360
      X=(XMX(J)+XMN(J))/2.0                                                 # Z00370
      Y=(YMX(I)+YMN(I))/2.0                                                 # Z00380
      DX=XMX(J)-XMN(J)                                                      # Z00390
      DY=YMX(I)-YMN(I)                                                      # Z00400
      ELSE IF(IVIEW.EQ.2) THEN                                              # Z00410
      X=(YMX(I)+YMN(I))/2.0                                                 # Z00420
      Y=(ZMX(K)+ZMN(K))/2.0                                                 # Z00430
      DX=YMX(I)-YMN(I)                                                      # Z00440
      DY=ZMX(K)-ZMN(K)                                                      # Z00450
      ELSE IF(IVIEW.EQ.3) THEN                                              # Z00460
      X=(XMX(J)+XMN(J))/2.0                                                 # Z00470
      Y=(ZMX(K)+ZMN(K))/2.0                                                 # Z00480
      DX=XMX(J)-XMN(J)                                                      # Z00490
      DY=ZMX(K)-ZMN(K)                                                      # Z00500
      END IF                                                                # Z00510
      IF (IVIEW.EQ.1) THEN                                                  # Z00520
      CALL SYMBL (X,Y,DX,DY,1)                                              # Z00530
      ELSE                                                                  # Z00540
```

```
      CALL SYMBL (X,Y,DX,DY,0)                                          # Z00550
      END IF                                                            # Z00560
10    CONTINUE                                                          # Z00570
5020   FORMAT(3I10,2F10.0,I10)                                          # Z00580
      WRITE(I7,*) '  DRAIN DATA HAS BEEN READ...'                      # Z00590
      IF (NCLR.GE.0) CALL PKCLR (1)                                     # Z00600
      RETURN                                                            # Z00610
      END                                                               # Z00620
C-----END OF ROUTINE-------------------------------------------------- # Z00630
C                                                                       #AA00010
C---GKS Version 1.0    March 1, 1990                                    #AA00020
C***********************************************************************#AA00030
C                                                                       #AA00040
C                                  PGHB                                 #AA00050
C                                                                       #AA00060
C  THIS ROUTINE READS GHB DATA AND PLOTS GENERAL HEAD BOUNDARIES        #AA00070
C                                                                       #AA00080
C***********************************************************************#AA00090
C                                                                       #AA00100
      SUBROUTINE PGHB (IMIN,IMAX,JMIN,JMAX,KMIN,KMAX,XMN,XMX,          #AA00110
     1YMN,YMX,ZMN,ZMX,NCOL,NROW,NLAY,IVIEW,IUGHB,NCLR,I7,IPROJ,ISLICE)  #AA00120
      DIMENSION XMN(NCOL),XMX(NCOL),YMN(NROW),YMX(NROW),ZMN(NLAY),      #AA00130
     1ZMX(NLAY)                                                         #AA00140
C                                                                       #AA00150
      IV= IVIEW*IPROJ                                                   #AA00160
      IF(NCLR.GE.0) THEN                                                #AA00170
      CALL PKCLR (2)                                                    #AA00180
      END IF                                                            #AA00190
      WRITE(I7,5000)                                                    #AA00200
5000  FORMAT(' ------------------------------------------------------#AA00210
     1----------------------')                                          #AA00220
      WRITE(I7,*) 'GHB DATA BEING READ...'                              #AA00230
      READ(IUGHB,5010) MXW,ID                                           #AA00240
      READ(IUGHB,5010) NGHB                                             #AA00250
5010   FORMAT(2I10)                                                     #AA00260
      DO 10 N=1,NGHB                                                    #AA00270
      READ(IUGHB,5020) K,I,J,H,C,IFACE                                  #AA00280
      IF(IFACE.NE.0) GO TO 10                                           #AA00290
      IF(I.LT.IMIN.OR.I.GT.IMAX) GO TO 10                               #AA00300
      IF(J.LT.JMIN.OR.J.GT.JMAX) GO TO 10                               #AA00310
      IF(K.LT.KMIN.OR.K.GT.KMAX) GO TO 10                               #AA00320
      IF (IV.EQ.1.AND.K.NE.ISLICE) GO TO 10                             #AA00330
      IF (IVIEW.EQ.2.AND.J.NE.ISLICE) GO TO 10                          #AA00340
      IF (IVIEW.EQ.3.AND.I.NE.ISLICE) GO TO 10                          #AA00350
      IF(IVIEW.EQ.1) THEN                                               #AA00360
      X=(XMX(J)+XMN(J))/2.0                                             #AA00370
      Y=(YMX(I)+YMN(I))/2.0                                             #AA00380
      DX=XMX(J)-XMN(J)                                                  #AA00390
      DY=YMX(I)-YMN(I)                                                  #AA00400
      ELSE IF(IVIEW.EQ.2) THEN                                          #AA00410
      X=(YMX(I)+YMN(I))/2.0                                             #AA00420
      Y=(ZMX(K)+ZMN(K))/2.0                                             #AA00430
      DX=YMX(I)-YMN(I)                                                  #AA00440
      DY=ZMX(K)-ZMN(K)                                                  #AA00450
      ELSE IF(IVIEW.EQ.3) THEN                                          #AA00460
      X=(XMX(J)+XMN(J))/2.0                                             #AA00470
      Y=(ZMX(K)+ZMN(K))/2.0                                             #AA00480
      DX=XMX(J)-XMN(J)                                                  #AA00490
      DY=ZMX(K)-ZMN(K)                                                  #AA00500
      END IF                                                            #AA00510
      IF (IVIEW.EQ.1) THEN                                              #AA00520
      CALL SYMBL (X,Y,DX,DY,1)                                          #AA00530
      ELSE                                                              #AA00540
      CALL SYMBL (X,Y,DX,DY,0)                                          #AA00550
```

```
      END IF                                                            #AA00560
10    CONTINUE                                                          #AA00570
5020   FORMAT(3I10,2F10.0,I10)                                          #AA00580
      WRITE(I7,*) '  GHB DATA HAS BEEN READ...'                        #AA00590
      IF (NCLR.GE.0) CALL PKCLR (1)                                     #AA00600
      RETURN                                                            #AA00610
      END                                                               #AA00620
C-----END OF ROUTINE----------------------------------------------------#AA00630
C                                                                       #AB00010
C---GKS Version 1.0    March 1, 1990                                    #AB00020
C***********************************************************************#AB00030
C                                                                       #AB00040
C                          OPNFIL                                       #AB00050
C                                                                       #AB00060
C  THIS ROUTINE OPENS A SINGLE FILE.                                    #AB00070
C                                                                       #AB00080
C***********************************************************************#AB00090
C                                                                       #AB00100
      SUBROUTINE OPNFIL (IU,FILNAM,NSTAT,IOUT,IBATCH,NACT)              #AB00110
      CHARACTER*(*) FILNAM                                              #AB00120
      CHARACTER*80 FNAME,FMT,FM                                         #AB00130
      LOGICAL*4 EX                                                      #AB00140
C                                                                       #AB00150
C                    VARIABLES                                          #AB00160
C                                                                       #AB00170
C  IU = FORTRAN UNIT NUMBER OF FILE                                     #AB00180
C  FILNAM = FILE NAME                                                   #AB00190
C  NSTAT = STATUS OF FILE                                               #AB00200
C          1 => OLD FILE                                                #AB00210
C          2 => NEW FILE                                                #AB00220
C          3 => SCRATCH FILE (DELETED AUTOMATICALLY WHEN RUN ENDS)      #AB00230
C          4 => UNDETERMINED STATUS (MAY OR MAY NOT EXIST)              #AB00240
C                 IF IT DOES NOT EXIST, IT IS CREATED BY OPEN STATEMENT #AB00250
C                 IF IT DOES EXIST, IT IS OPENED AS 'OLD' FILE          #AB00260
C                                                                       #AB00270
C  IOUT = UNIT NUMBER FOR OUTPUT FILE TO WRITE ERROR MESSAGE TO         #AB00280
C         IF NECESSARY                                                  #AB00290
C                                                                       #AB00300
C  IBATCH = FLAG INDICATING IF THERE IS INTERACTIVE DIALOGUE AT         #AB00310
C           TERMINAL                                                    #AB00320
C                                                                       #AB00330
C           0 => THERE IS INTERACTIVE DIALOGUE                          #AB00340
C           1 => THERE IS NOT INTERACTIVE DIALOGUE (BATCH MODE)         #AB00350
C                                                                       #AB00360
C  NACT = VARIABLE DENOTING IF A FILE IS READ ONLY, WRITE ONLY, OR      #AB00370
C         READ AND WRITE.                                               #AB00380
C                                                                       #AB00390
C           1 => READ ONLY                                              #AB00400
C           2 => WRITE ONLY                                             #AB00410
C           3 => READ AND WRITE                                         #AB00420
C                                                                       #AB00430
C  "NACT" IS NOT USED IN THIS SUBROUTINE. ALL FILES ARE OPENED FOR      #AB00440
C  READING AND WRITING. OPENING "READ ONLY" AND "WRITE ONLY" FILES      #AB00450
C  IS MACHINE DEPENDENT. THE VARIABLE "NACT" IS PASSED TO THIS          #AB00460
C  ROUTINE TO MAKE IT EASY TO MODIFY THE OPEN STATEMENTS TO ALLOW       #AB00470
C  FOR READ AND WRITE ONLY FILES ON ANY GIVEN MACHINE. THE CALLS TO     #AB00480
C  THIS SUBROUTINE ARE CURRENTLY SET UP SO THAT ANY FILE THAT IS        #AB00490
C  WRITTEN TO IS GIVEN "NACT = 3" AND ANY FILE THAT IS READ FROM BUT    #AB00500
C  NOT WRITTEN TO IS GIVEN "NACT = 1". NO FILES ARE GIVEN "WRITE ONLY"  #AB00510
C  STATUS. "READ ONLY" STATUS IS USEFUL IF A NUMBER OF USERS WANT       #AB00520
C  TO SIMULTANEOUSLY SHARE INPUT FILES.                                 #AB00530
C                                                                       #AB00540
C  IF THE FILE DOES NOT CURRENTLY EXIST, A NEGATIVE UNIT NUMBER IS      #AB00550
C  USED AS A FLAG TO INDICATED THAT A NEW FILE SHOULD BE CREATED AS     #AB00560
```

```
C  AN UNFORMATTED (BINARY) FILE.                                        #AB00570
C                                                                       #AB00580
      IBIN=0                                                            #AB00590
      IF (IU.LT.0) THEN                                                 #AB00600
      IU= -IU                                                           #AB00610
      IBIN=1                                                            #AB00620
      END IF                                                            #AB00630
C                                                                       #AB00640
C  CHECK TO SEE IF A FILE IS OPENED TO UNIT=IOUT SO THAT ERROR          #AB00650
C  MESSAGES CAN BE WRITTEN.                                             #AB00660
      INQUIRE (UNIT=IOUT,OPENED=EX)                                     #AB00670
      IO=0                                                              #AB00680
      IF (EX) IO=1                                                      #AB00690
C                                                                       #AB00700
      FNAME=FILNAM                                                      #AB00710
C                                                                       #AB00720
C  OPEN AN EXISTING FILE                                                #AB00730
C                                                                       #AB00740
      IF (NSTAT.EQ.1) THEN                                              #AB00750
10    INQUIRE (FILE=FNAME,EXIST=EX,UNFORMATTED=FM)                      #AB00760
C  CHECK TO SEE IF FILE EXISTS                                          #AB00770
      IF (EX) GO TO 20                                                  #AB00780
      IF (IBATCH.EQ.0) THEN                                             #AB00790
      WRITE (*,*) 'FILE DOES NOT EXIST.'                                #AB00800
      WRITE (*,*) 'ENTER THE NAME OF AN EXISTING FILE (<CR>=QUIT):'     #AB00810
      READ (*,'(A)') FNAME                                              #AB00820
      IF (FNAME.EQ.' ') STOP                                            #AB00830
      GO TO 10                                                          #AB00840
      ELSE                                                              #AB00850
      IF (IO.EQ.1) WRITE (IOUT,*) 'FILE DOES NOT EXIST:'                #AB00860
      IF (IO.EQ.1) WRITE (IOUT,'(A)') FNAME                             #AB00870
      STOP                                                              #AB00880
      END IF                                                            #AB00890
20    CONTINUE                                                          #AB00900
      FMT='FORMATTED'                                                   #AB00910
      IF (IBIN.EQ.1.OR.FM.EQ.'YES') FMT='UNFORMATTED'                   #AB00920
      OPEN (IU,FILE=FNAME,STATUS='OLD',FORM=FMT,IOSTAT=IERR)            #AB00930
      IF (IERR.GT.0) GO TO 40                                           #AB00940
      RETURN                                                            #AB00950
      END IF                                                            #AB00960
C                                                                       #AB00970
C  OPEN A NEW FILE                                                      #AB00980
C                                                                       #AB00990
      IF (NSTAT.EQ.2) THEN                                              #AB01000
30    INQUIRE (FILE=FNAME,EXIST=EX)                                     #AB01010
      IF (EX) THEN                                                      #AB01020
      IF (IBATCH.EQ.0) THEN                                             #AB01030
      WRITE (*,*) 'FILE ALREADY EXISTS.'                                #AB01040
      WRITE (*,*) 'ENTER THE NAME OF A NEW FILE (<CR>=QUIT):'           #AB01050
      READ (*,'(A)') FNAME                                              #AB01060
      IF (FNAME.EQ.' ') STOP                                            #AB01070
      GO TO 30                                                          #AB01080
      ELSE                                                              #AB01090
      IF (IO.EQ.1) WRITE (IOUT,*) 'FILE ALREADY EXISTS:'                #AB01100
      IF (IO.EQ.1) WRITE (IOUT,'(A)') FNAME                             #AB01110
      STOP                                                              #AB01120
      END IF                                                            #AB01130
      END IF                                                            #AB01140
      FMT='FORMATTED'                                                   #AB01150
      IF (IBIN.EQ.1) FMT='UNFORMATTED'                                  #AB01160
      OPEN (IU,FILE=FNAME,STATUS='NEW',FORM=FMT,IOSTAT=IERR)            #AB01170
      IF (IERR.GT.0) GO TO 40                                           #AB01180
      RETURN                                                            #AB01190
      END IF                                                            #AB01200
```

```
C                                                                       #AB01210
C  OPEN A SCRATCH FILE                                                  #AB01220
C                                                                       #AB01230
      IF (NSTAT.EQ.3) THEN                                              #AB01240
      FMT='FORMATTED'                                                   #AB01250
      IF (IBIN.EQ.1) FMT='UNFORMATTED'                                  #AB01260
      OPEN (IU,STATUS='SCRATCH',FORM=FMT,IOSTAT=IERR)                   #AB01270
C  FOR MICROSOFT FORTRAN USE:                                           #AB01280
C      OPEN (IU,FORM=FMT,IOSTAT=IERR)                                   #AB01290
      IF (IERR.GT.0) GO TO 40                                           #AB01300
      RETURN                                                            #AB01310
      END IF                                                            #AB01320
C                                                                       #AB01330
C  OPEN A FILE OF UNKNOWN STATUS                                        #AB01340
C                                                                       #AB01350
      IF (NSTAT.EQ.4) THEN                                              #AB01360
      INQUIRE (FILE=FNAME,EXIST=EX,UNFORMATTED=FM)                      #AB01370
      IF (EX) THEN                                                      #AB01380
      FMT='FORMATTED'                                                   #AB01390
      IF (IBIN.EQ.1.OR.FM.EQ.'YES') FMT='UNFORMATTED'                   #AB01400
      OPEN (IU,FILE=FNAME,STATUS='OLD',FORM=FMT,IOSTAT=IERR)            #AB01410
      IF (IERR.GT.0) GO TO 40                                           #AB01420
      ELSE                                                              #AB01430
      FMT='FORMATTED'                                                   #AB01440
      IF (IBIN.EQ.1) FMT='UNFORMATTED'                                  #AB01450
      OPEN (IU,FILE=FNAME,STATUS='NEW',FORM=FMT,IOSTAT=IERR)            #AB01460
      IF (IERR.GT.0) GO TO 40                                           #AB01470
      END IF                                                            #AB01480
      RETURN                                                            #AB01490
      END IF                                                            #AB01500
C                                                                       #AB01510
C  WRITE MESSAGE INDICATING PROBLEM OPENING FILE                        #AB01520
C                                                                       #AB01530
40    IF (IBATCH.EQ.0) WRITE (*,5000) IU                                #AB01540
      IF (IO.EQ.1) WRITE (IOUT,5000) IU                                 #AB01550
5000  FORMAT (' ERROR OPENING FILE TO UNIT ',I3)                        #AB01560
      STOP                                                              #AB01570
      END                                                               #AB01580
C-----END OF ROUTINE----------------------------------------------------#AB01590
C                                                                       #AC00010
C---GKS Version 1.0    March 1, 1990                                    #AC00020
C***********************************************************************#AC00030
C                                                                       #AC00040
C                           NUMCNV                                      #AC00050
C                                                                       #AC00060
C  THIS ROUTINE OUTPUTS A REAL NUMBER AS TEXT.                          #AC00070
C                                                                       #AC00080
C***********************************************************************#AC00090
C                                                                       #AC00100
      SUBROUTINE NUMCNV(Z,NDEC,STRING,NCHAR)                            #AC00110
C                                                                       #AC00120
      CHARACTER*20 STRING                                               #AC00130
      CHARACTER*20 BUF                                                  #AC00140
      CHARACTER*7 FMT                                                   #AC00150
      CHARACTER*10 DEC                                                  #AC00160
      DATA FMT/'(F20.X)'/                                               #AC00170
      DATA DEC/'0123456789'/                                            #AC00180
C                                                                       #AC00190
      ND=NDEC+1                                                         #AC00200
      IF(ND.GT.10) ND=10                                                #AC00210
      IF(ND.LT.1) ND=1                                                  #AC00220
      FMT(6:6)=DEC(ND:ND)                                               #AC00230
      WRITE(BUF,FMT) Z                                                  #AC00240
C                                                                       #AC00250
```

```
      DO 10 ISTART=1,20                                                 #AC00260
      IF(BUF(ISTART:ISTART).NE.' ') GO TO 20                            #AC00270
10    CONTINUE                                                          #AC00280
C                                                                       #AC00290
20    ISTOP=20                                                          #AC00300
      IF(NDEC.LT.0) ISTOP=19                                            #AC00310
      NCHAR=ISTOP-ISTART+1                                              #AC00320
      STRING(1:NCHAR)=BUF(ISTART:ISTOP)                                 #AC00330
      RETURN                                                            #AC00340
      END                                                               #AC00350
C-----END OF ROUTINE----------------------------------------------------#AC00360
C                                                                       #AD00010
C---Version 1.0   July 21, 1989                                         #AD00020
C***********************************************************************#AD00030
C                                                                       #AD00040
C                          DEVICE                                       #AD00050
C                                                                       #AD00060
C  THIS ROUTINE SELECTS THE GRAPHICS OUTPUT DEVICE                      #AD00070
C                                                                       #AD00080
C***********************************************************************#AD00090
C                                                                       #AD00100
      SUBROUTINE DEVICE (MODEL,KIND,NWAIT,IWAIT,IU,I7)                  #AD00110
      DIMENSION IDEV(10),ITP(10)                                        #AD00120
      CHARACTER*60 LABEL(10)                                            #AD00130
C  OPEN GRAPHICS DEVICE DATA FILE                                       #AD00140
      CALL OPNFIL (IU,'DEVICE.DAT',1,I7,0,3)                            #AD00150
C                                                                       #AD00160
      READ(IU,*,END=20,ERR=20) MD,MTP,LABEL(1)                          #AD00170
      NDEV=1                                                            #AD00180
      IDEV(1)=MD                                                        #AD00190
      ITP(1)=MTP                                                        #AD00200
      DO 30 N=2,10                                                      #AD00210
      READ(IU,*,END=40,ERR=20) MD,MTP,LABEL(N)                          #AD00220
      NDEV=N                                                            #AD00230
      IDEV(NDEV)=MD                                                     #AD00240
      ITP(NDEV)=MTP                                                     #AD00250
30    CONTINUE                                                          #AD00260
40    CONTINUE                                                          #AD00270
      IF(NDEV.EQ.1) THEN                                                #AD00280
      IM=1                                                              #AD00290
      ELSE                                                              #AD00300
      WRITE(*,*) 'ENTER THE TYPE OF GRAPHICS OUTPUT DEVICE:'            #AD00310
      DO 45 N=1,NDEV                                                    #AD00320
      WRITE(*,5000) N,LABEL(N)                                          #AD00330
45    CONTINUE                                                          #AD00340
50    READ(*,*) IM                                                      #AD00350
      IF (IM.LT.1.OR.IM.GT.NDEV) THEN                                   #AD00360
      WRITE(*,*) 'THE NUMBER IS OUT OF RANGE. CHOOSE AGAIN.'            #AD00370
      GO TO 50                                                          #AD00380
      END IF                                                            #AD00390
      END IF                                                            #AD00400
      MODEL = IDEV(IM)                                                  #AD00410
      IWAIT=0                                                           #AD00420
      IF (ITP(IM).EQ.0) IWAIT=NWAIT                                     #AD00430
      KIND=ITP(IM)                                                      #AD00440
      RETURN                                                            #AD00450
20    WRITE(*,*) 'THE FILE DEVICE.DAT IS EMPTY OR HAS BAD DATA'         #AD00460
      STOP                                                              #AD00470
5000  FORMAT(1X,I5,' = ',A60)                                           #AD00480
      END                                                               #AD00490
C-----END OF ROUTINE----------------------------------------------------#AD00500
```