UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

Computer Program SPDIKE for Calculation of Self-Potential Anomalies Near Vertical Dikes

by

David V. Fitterman

Open File Report 82-470

1982

Contents

Introduction

This report describes program SPDIKE, which is used to compute self-potential (SP) anomalies produced by a thermoelectrical source in the vicinity of a vertical dike. The theory of the calculations is described in a separate paper (Fitterman, 1982). The SP anomaly on the surface of the half-space can be computed along a specified traverse or on a rectangular grid. In addition to printable output, the user can request the output be written into files suitable for plotting or contouring. The model geometry is shown in Figure 1. The model consists of a dike (region 2) of width w surrounded by two quarter-spaces (regions 1 and 3). Each region has a conductivity σ_i and a thermoelectric coefficient C_i. The source voltages are created by the elevated temperature along the interfaces between the adjacent regions. For simplicity, the temperature is assumed to be constant within a rectangular region on each interface. These regions are buried at a depth d_i, have a vertical extent t_i, and a strike length of l_i. The temperature along the planes y=w/2 and y=-w/2 outside of the source regions, and on the surface z=o are assumed to be zero. The program could be easily modified to allow the temperature inside each source region to vary. This feature has not been implemented. The source intensities are given by $S_1 = (C_2-C_1)F_1$, and $S_2 = (C_3-C_2)F_2$ respectively.

The coordinate system is centered on the dike with the strike direction parallel the x-axis. The center of source 1 is on the y-axis, while the center of source 2 is located at x-offset. Notice that the coordinate system is right handed, and that the positive z-axis points downward.

Site Dependent Modifications

This program has been run on several computer systems including Honeywell MULTICS, IBM 370/165 DOS, and VAX/VMS 11/780. The version presented here is intended for the VAX/VMS system. All site-dependent modifications are noted by comments outlined with pound signs (#). If site dependent statements replace statements which should run on most other systems, the replaced statements have been made into comments to make reconversion easy.

VAX-11 FORTRAN does not presently support NAMELIST input. This input mode must be simulated by other calls (NAM$LIST, NAM$COM, NAM$READ). To allow optimization of the compilation the dummy labelled COMMON called "VAX" had to be included.

In function subprograms G1, G2, and G3 the parameter BIG must be set to largest allowable argument for the exponential function. For proper execution subroutine HANKEL (Anderson, 1982) requires that the results of underflows are set to zero. This is automatically done on the VAX/VMS system.

Logical unit assignments are set in lines 860 and 870 of the main program. Input data are read on LUIN and output results are written on LUOUT. Program created error messages are written to LUOUT and LULOG. Output for plotting goes to LUPLT. The first plot output is written to the initial value of LUPLT plus one. For example, if LUSTD is set to 10 in the DATA statement on line 870, the first plotter output file will be written on logical unit 11. Each data set which produces plotter output increments the value of LUPLT before writing the output.

The user must assign the input file to LUIN external to the program. More will be said about this later when VAX operating information is discussed.

The user may have to modify two subroutines OUTPLOT and OUTSTD, which write output for plotting traverses and contouring gridded data respectively. The version of OUTSTD furnished produces data in a USGS Standard Format suitable for contouring with the Evenden CONTOUR program (Evenden, 1975).

Other Necessary Subroutines

Three subroutines written by W. Anderson are used by this program (Anderson, 1971, Anderson, 1982):

SPLIN1 - computes spline interpolation coefficients

SPOINT - interpolates a single point using coefficients computed by SPLIN1

HANKEL - computes Hankel transform

They are included without further explanation. For more information the user should see the above cited references.

Input Parameters

The input data are read from a file containing title information and model parameters. An unlimited number of input data sets can be contained in an input file. The problem title is input using a formatted read, while the model parameters are input using NAMELIST. Use of NAMELIST allows inputting only parameters which have been changed from the previous model.

Below is a description of the input file read from logical unit LUIN. If allowable ranges exist on the parameters, or default values exist, they are specified.

1. TITLE - 80 character data title.

2. $INPUT - parameter NAMELIST sequence containing some or all of following:

   TOL - Hankel transform tolerance. Should be 3 orders of magnitude smaller than desired relative accuracy.
   Range: ≥ 0 Default: 1.E-7

   ORDER - Gaussian integration order. Computational effort increases as the square of this number. User should experiment to determine value needed for desired accuracy.
   Range: 1-20 Default: 5

   WIDTH - dike width
   Range: >0 Default: 1.0

   OFFSET - relative offset along strike of the center of Source 2 with respect to the center of Source 1. Source 1 is centered on the y-axis (x=0)
   Range: all Default: 0.0

   SIGMA1 - medium 1 conductivity in S/m
   Range: >0 Default: 1.0

   SIGMA2 - medium 2 conductivity in S/m
   Range: >0 Default: 1.0

   SIGMA3 - medium 3 conductivity in S/m
   Range: >0 Default: 1.0

   C1 - medium 1 thermoelectric coefficient in mV/^{o}C
   Range: all Default: 1.0

   C2 - medium 2 thermoelectric coefficient in mV/^{o}/C
   Range: all Default: 1.0

   C3 - medium 3 thermoelectric coefficient in mV/^{o}C
   Range: all Default: 1.0
   N.B. There must be a non-zero difference between any two thermoelectric coefficients.

   F1 - source 1 temperature in ^{o}C for contact between regions 1 and 2.
   Range: all Default: 1000

   F2 - source 2 temperature in ^{o}C for contact between regions 2 and 3.
   Range: all Default: 1000
   N.B. Both source temperatures cannot be zero.

D1 - depth to top of source 1
Range: >0 Default: 1.0

T1 - vertical extent of source 1
Range: >0 Default: 1.0

L1 - strike length of source 1
Range: >0 Default: 1.0

D2 - depth to top of source 2
Range: >0 Default: 1.0

T2 - vertical extent of source 2
Range: >0 Default: 1.0

L2 - strike length of source 2
Range: >0 Default: 1.0

X1 - strike coordinate for first traverse point or lower left corner for grid computation
Range: any Default: 0.0

Y1 - perpendicular coordinate for first traverse point or lower left corner for grid computation
Range: any Default: -2.5

X2 - strike coordinate for last traverse point or upper right corner for grid computation
Range: any Default: 0.0

Y2 - perpendicular coordinate for last traverse point or upper right corner for grid computation
Range: any Default: 2.5

DX - spacing between grid points in x-direction
Range: >0 for ICTYP=3 or 4 Default: 0.0

DY - spacing between grid points in y-direction
Range: >0 for ICTYP=3 or 4 Default: 0.0

DR - spacing between traverse points
Range: >0 for ICTYP=1 or 2 Default: 0.25

ICTYP - computation type:
1=traverse computation
2=traverse plus plottable output file
3=grid computation
4=grid computation plus contourable output file
Range: 1-4 Default: 1

ISTOP - current data set action flag:
0=process current data set and continue
1=process current data set and stop
2=skip current data set and continue
Range: 0-2 Default: 1
N. B. ISTOP=2 will change any NAMELIST parameters which were read.

All NAMELIST input must begin in column 2. The first and last characters must be a $ or &. An example of an allowable input data set is given below.

```
                Column 1
                ↓
1st line¶       EXAMPLE INPUT DATA SET
2nd line¶        $ INPUT  WIDTH=2.0, SIGMA2=10.0,
3rd line¶        DR=0.1, ISTOP=1 $
```

Figure 1 contains a print out of another input data set.

Since the calculations are scale invariant, any system of distance units can be used as long as all distances are in the same units. The solutions depend only on the ratios of conductivities, and not the absolute conductivity values. Thus keeping the ratios SIGMA1/SIGMA2 and SIGMA2/SIGMA3 constant will not change the solution even if the individual conductivities are varied.

Multiple data sets can be included in one input file. Each data set begins with a title line followed by the NAMELIST data. Parameter ISTOP should be equal to 0 or 2 if processing of subseqent data sets is to occur. Remember that if ISTOP=2, the calculations for the current data set will not be performed, but any NAMELIST input variables will be read and modified. The subsequent data set will use the previous values for the parameters in the NAMELIST unless new values are read in.

Error Messages

Error messages can be generated during execution of the program. This is usually the result of an input parameter being out of allowable bounds. A summary error message will be written to LULOG -- usually the interactive terminal -- and in some cases a detailed message will appear in the output

file (LUOUT). The following list contains the possible error messages and an explanation.

1.	INPUT PARAMETER	- some input parameter is not within the allowable range.
2.	HANKEL TRANSFORM TABLE	- The Hankel transform table will exceed the allocated storage space (100) the size of arrays RWORK2 and RWORK2 must be increased, or the ratio of the maximum to minimum distance between the observation points and the source regions must be made smaller.
3.	TRAVERSE LENGTH	- the number of points in the traverse exceeds NMAX(120), DR must be reduced, or the size of arrays X, Y, and POT increased.
4.	GRID X SIZE	- the number of points in the x-direction of the grid exceeds NMAX, DX must be reduced, or the size of arrays X and POT increased.
5.	GRID Y SIZE	- the number of points in the y-direction of the grid exceeds NMAX, DY must be reduced, or the size of arrays Y and POT increased.
6.	HANKEL TRANSFORM	- an error occurred during a call to routine HANKEL. This message can be generated for traverse or grid calculation. Consult Anderson (1982) for explanation of problem.

All errors result in termination of the current data set computations.

Output Description

Sample input and output files are shown in Figures 2 and 3 respectively for a traverse calculation. The title appears at the top followed by the model description. The last part of the output contains the distance between observation points DR, the observation point location and computed potentials.

When calculations are made on a grid, there is an output sheet for each value of X. Figure 4 contains an input data set for such a case. This example produces the three pages of output shown in Figure 5.

VAX/VMS Operating Instructions

Use of program SPDIKE on the VAX/VMS system is relatively easy. First the user creates a file which contains the input data. The input file is assigned

to logical unit LUIN, which is set to 9 in the present version. The output is written to a file called FOR010.DAT. Changing the values of LUOUT will modify this. If plottable output is requested (ICTYP=2 or 4), files named FOR011.DAT, FOR012.DAT, etc. will be created--one for each input data set.

Listed below are the commands which will perform the execution of SPDIKE

```
$ASSIGN  input_file  FOR009
$RUN  DRA2:[FITTERMAN]SPDIKE
$DEASSIGN  FOR009
$RENAME FOR010  output_file
```

The following commands might be used if plottable output is requested.

```
$RENAME  FOR011  plotting_file_1
$RENAME  FOR012  plotting_file_2
```

References

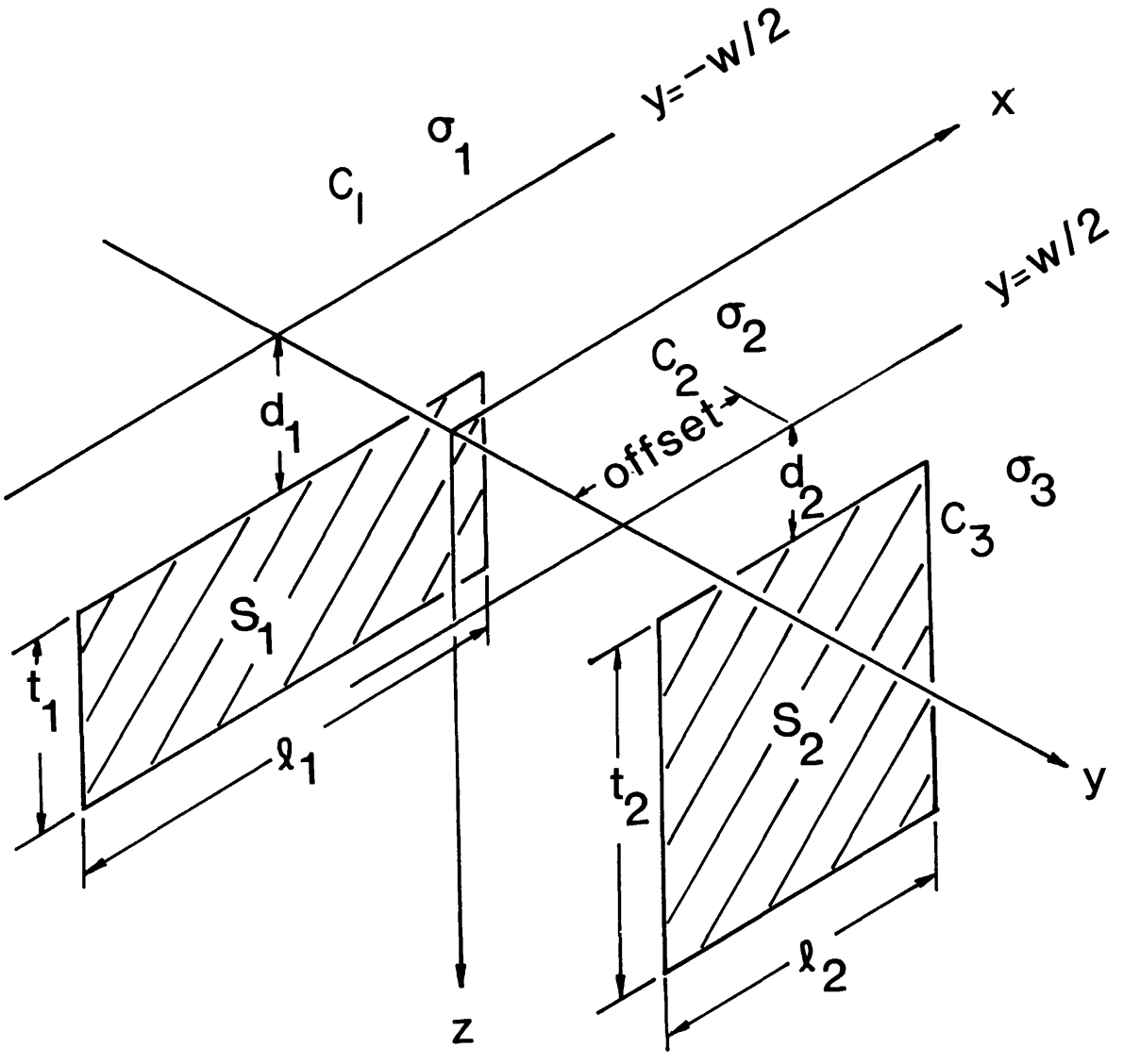
Anderson, W. L., 1971, Application of bicubic spline functions on two-dimensional gridded data: NTIS Report PB-203-579, National Technical Information Service, U.S. Dept. of Commerce, Springfield, VA 22161.

Anderson, W. L., 1982, Fast Hankel transforms using related and lagged convolutions: in press ACM Transactions or Mathematical Software.

Evenden, G. I., 1975, A general purpose contouring system: U.S.G.S. Open-File Report 75-317, 108 p.

Fitterman, D. V., 1982, Modelling of self-potential anomalies near vertical dikes: submitted to Geophysics.

Figure 1 Dike model geometry. Temperature inside the two rectangular source regions are F_1 and F_2 resulting in source intensities of $S_1=(C_2-C_1)F_1$ and $S_2=(C_3-C_2)F_2$ respectively.



```
EXAMPLE OF TRAVERSE CALCULATION
 $INPUT WIDTH=2.0, OFFSET=0.5,
 SIGMA1=1.0, SIGMA2=2.0, SIGMA3=5.0,
 C1=2.0, C2=0.0, C3=2.0, F1=300., F2=250.,
 D1=0.5, T1=1.0, L1=1.5, D2=0.25, T2=1.5, L2=2.5,
 X1=-2.0, Y1=-2.0, X2=2.0, Y2=2.0, DR=0.2,
 ICTYP=1, ISTOP=1 $
```

Figure 2 Example of input data for traverse calculation.

```
EXAMPLE OF TRAVERSE CALCULATION

OFFSET=   0.500  WIDTH=   2.000  TOL=0.1000E-06  ORDER= 5

                    /              /
                   /       X      /
      ------------/-------/------/-------------Y
                  |       |      |                      SOURCE 1     SOURCE 2
                  |       Z      |              D=       0.5000       0.2500
                  |              |              T=       1.0000       1.5000
REGION=     1     S1      2      S2      3       L=       1.5000       2.5000
SIGMA =   1.0000  S1    2.0000   S2    5.0000    F=     300.0000     250.0000
C     =   2.0000  S1    0.0000   S2    2.0000    S=    -600.0000     500.0000
                  |              |
           Y=   -1.0000          1.0000

DR=  0.200
     X          Y          POT
   -2.00      -2.00      -9.4693
   -1.86      -1.86     -10.7892
   -1.72      -1.72     -12.6196
   -1.58      -1.58     -15.2905
   -1.43      -1.43     -19.3826
   -1.29      -1.29     -25.9182
   -1.15      -1.15     -36.6048
   -1.01      -1.01     -53.7725
   -0.87      -0.87     -68.9586
   -0.73      -0.73     -86.1304
   -0.59      -0.59    -103.0787
   -0.44      -0.44    -118.1774
   -0.30      -0.30    -131.4402
   -0.16      -0.16    -143.4252
   -0.02      -0.02    -154.4768
    0.12       0.12    -164.4093
    0.26       0.26    -172.5696
    0.40       0.40    -178.1018
    0.55       0.55    -177.7334
    0.69       0.69    -157.7049
    0.83       0.83    -104.8974
    0.97       0.97     -30.5247
    1.11       1.11      18.8942
    1.25       1.25      38.7396
    1.39       1.39      40.7563
    1.54       1.54      36.8966
    1.68       1.68      31.3037
    1.82       1.82      25.9937
    1.96       1.96      21.5532
```

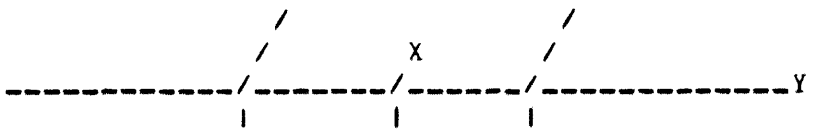
Figure 3 Output from example traverse calculation

```
EXAMPLE OF GRID CALCULATION
 $INPUT WIDTH=2.5, OFFSET=-0.75,
 SIGMA1=3.0, SIGMA2=10.0, SIGMA3=1.0,
 C1=1.0, C2=0.0, C3=1.5, F1=150., F2=200.,
 D1=0.25, T1=0.75, L1=1.25, D2=0.4, T2=1.25, L2=1.6,
 X1=-2.0, Y1=-2.0, X2=2.0, Y2=2.0, DX=2.0, DY=0.2,
 ICTYP=3, ISTOP=1 $
```

Figure 4 Example input for grid calculation

EXAMPLE OF GRID CALCULATION

OFFSET= -0.750 WIDTH= 2.500 TOL=0.1000E-06 ORDER= 5



						SOURCE 1	SOURCE 2
					D=	0.2500	0.4000
					T=	0.7500	1.2500
REGION=	1	S1	2	S2	3	L= 1.2500	1.6000
SIGMA =	3.0000	S1	10.0000	S2	1.0000	F= 150.0000	200.0000
C =	1.0000	S1	0.0000	S2	1.5000	S= -150.0000	300.0000
	Y=	-1.2500		1.2500			

X= -2.00 DY= 0.200

Y	POT
-2.00	0.2534
-1.80	-0.4398
-1.60	-1.3226
-1.40	-2.3617
-1.20	-3.2909
-1.00	-3.6441
-0.80	-3.9915
-0.60	-4.3147
-0.40	-4.6025
-0.20	-4.8477
0.00	-5.0397
0.20	-5.1569
0.40	-5.1580
0.60	-4.9743
0.80	-4.5105
1.00	-3.6800
1.20	-2.5138
1.40	7.1159
1.60	17.1979
1.80	23.4855
2.00	26.4688

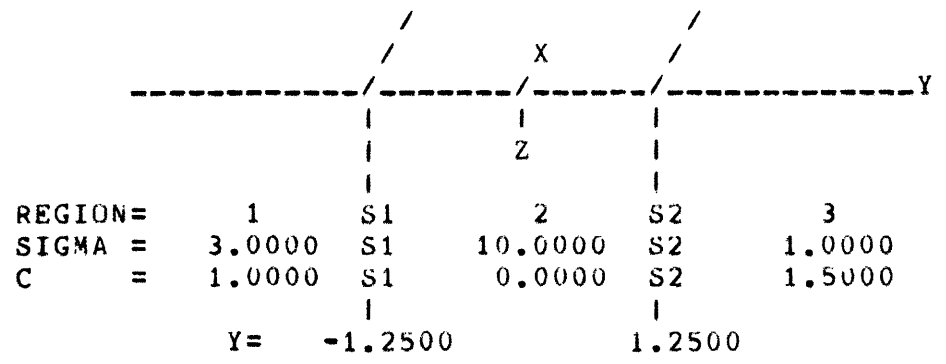
Figure 5a Output from example grid calculation (page 1).

```
EXAMPLE OF GRID CALCULATION

OFFSET=  -0.750  WIDTH=   2.500  TOL=0.1000E-06  ORDER= 5

                    /               /
                   /       X       /
      ------------/-------/-------/-------------Y
                  |       |       |                      SOURCE 1     SOURCE 2
                  |       Z       |              D=       0.2500       0.4000
                  |               |              T=       0.7500       1.2500
REGION=     1     S1       2      S2       3     L=       1.2500       1.6000
SIGMA =   3.0000  S1    10.0000   S2    1.0000   F=     150.0000     200.0000
C     =   1.0000  S1     0.0000   S2    1.5000   S=    -150.0000     300.0000
                  |               |
            Y=   -1.2500          1.2500

X=    0.00    DY=  0.200
      Y          POT
    -2.00      21.8875
    -1.80      26.6577
    -1.60      28.9136
    -1.40      19.3628
    -1.20      -6.4424
    -1.00     -12.9024
    -0.80     -13.5662
    -0.60     -12.5123
    -0.40     -11.3237
    -0.20     -10.4046
     0.00      -9.7984
     0.20      -9.4416
     0.40      -9.2242
     0.60      -8.9764
     0.80      -8.4049
     1.00      -7.0054
     1.20      -4.3291
     1.40      19.1888
     1.60      38.3349
     1.80      45.2896
     2.00      45.6068
```

Figure 5b Output from example grid calculation (page 2).

EXAMPLE OF GRID CALCULATION

OFFSET= -0.750 WIDTH= 2.500 TOL=0.1000E-06 ORDER= 5



	SOURCE 1	SOURCE 2
D=	0.2500	0.4000
T=	0.7500	1.2500
L=	1.2500	1.6000
F=	150.0000	200.0000
S=	-150.0000	300.0000

X= 2.00 DY= 0.200

Y	POT
-2.00	1.0116
-1.80	0.4385
-1.60	-0.3030
-1.40	-1.1766
-1.20	-1.9475
-1.00	-2.2174
-0.80	-2.4506
-0.60	-2.6296
-0.40	-2.7472
-0.20	-2.8040
0.00	-2.8057
0.20	-2.7598
0.40	-2.6741
0.60	-2.5561
0.80	-2.4135
1.00	-2.2545
1.20	-2.0877
1.40	-0.7976
1.60	0.8063
1.80	2.2774
2.00	3.5634

Figure 5c Output from example grid calculation (page 3).

Source Listing

The attached subprograms are listed in the following order:


```
      PROGRAM SPDIKE                                                    00000010
C                                                                       00000020
C     A PROGRAM TO COMPUTE THE SELF-POTENTIAL (SP) ANOMALY              00000030
C     PRODUCED BY A CROSS COUPLING SOURCE IN THE VICINITY OF            00000040
C     AN OUTCROPPING VERTICAL DIKE.  THE SOURCE IS CONSIDERED           00000050
C     TO BE CONSTANT OVER A RECTANGULAR PATCH LOCATED ON                00000060
C     BOTH MATERIAL BOUNDARIES.                                         00000070
C                                                                       00000080
C     WRITTEN BY D. V. FITTERMAN, U.S.G.S., FEBRUARY 1980               00000090
C     MODIFIED  20 APRIL 1982                                           00000100
C                                                                       00000110
C     PARAMETER FILE INPUT SEQUENCE (LUIN)                              00000120
C     1. TITLE        - DATA SET TITLE (80 CHARACTERS)                  00000130
C     2. $INPUT       - INPUT PARAMETER NAMELIST CONTAINING:            00000140
C          TOL        - HANKEL TRANSFORM TOLERANCE, DEFAULT=1.E-7       00000150
C                       SHOULD BE SET 3 ORDERS OF MAGNITUDE SMALLER THAN 00000160
C                       DESIRED RELATIVE ACCURACY.                      00000170
C          ORDER      - GAUSSIAN INTEGRATION ORDER (1-20) DEFAULT=5     00000180
C          WIDTH      - DIKE WIDTH DEFAULT=1.0                          00000190
C          OFFSET     - RELATIVE OFFSET ALONG STRIKE OF THE CENTER      00000200
C                       OF SOURCE PATCH 2 WITH RESPECT TO SOURCE        00000210
C                       PATCH 1.  PATCH 1 CENTERED ON Y-AXIS.           00000220
C                       DEFAULT=0.0                                     00000230
C          SIGMA1     - MEDIUM 1 CONDUCTIVITY  DEFAULT=1.0              00000240
C          SIGMA2     - MEDIUM 2 CONDUCTIVITY  DEFAULT=1.0              00000250
C          SIGMA3     - MEDIUM 3 CONDUCTIVITY  DEFAULT=1.0              00000260
```

```
C          C1          - MEDIUM 1 COUPLING COEFFICIENT  DEFAULT=1.0        00000270
C          C2          - MEDIUM 2 COUPLING COEFFICIENT  DEFAULT=1.0        00000280
C          C3          - MEDIUM 3 COUPLING COEFFICIENT  DEFAULT=1.0        00000290
C          F1          - CONTACT 1-2 DRIVING FORCE INTENSITY               00000300
C                        DEFAULT=1000.                                     00000310
C          F2          - CONTACT 2-3 DRIVING FORCE INTENS                  00000320
C                        DEFAULT=1000.                                     00000330
C          D1          - DEPTH TO TOP OF PATCH 1  DEFAULT=1.0              00000340
C          T1          - VERTICAL EXTENT OF PATCH 1  DEFAULT=1.0           00000350
C          L1          - STRIKE LENGTH OF PATCH 1  DEFAULT=1.0             00000360
C          D2          - DEPTH TO TOP OF PATCH 2  DEFAULT=1.0              00000370
C          T2          - VERTICAL EXTENT OF PATCH 2  DEFAULT=1.0           00000380
C          L2          - STRIKE LENGTH OF PATCH 2  DEFAULT=1.0             00000390
C          X1          - STRIKE COORDINATE OF FIRST POINT  DEFAULT=0.0     00000400
C          Y1          - PERPENDICULAR COORDINATE OF FIRST POINT           00000410
C                        DEFAULT=-2.5                                      00000420
C          X2          - STRIKE COORDINATE OF LAST POINT  DEFAULT=0.0      00000430
C          Y2          - PERPENDICULAR COORDINATE OF LAST POINT            00000440
C                        DEFAULT=2.5                                       00000450
C          DX          - SPACING BETWEEN GRID ROWS  DEFAULT=0.0            00000460
C          DY          - SPACING BETWEEN GRID COLUMNS  DEFAULT=0.0         00000470
C          DR          - SPACING BETWEEN TRAVERSE POINTS  DEFAULT=0.25     00000480
C          ICTYP       - COMPUTATION TYPE, 1=TRAVERSE                      00000490
C                                          2=TRAVERSE PLUS PLOTTING FILE   00000500
C                                          3=GRID                          00000510
C                                          4=GRID PLUS STANDARD FILE       00000520
C                        DEFAULT=1                                         00000530
C          ISTOP       - ACTION FLAG FOR CURRENT DATA SET                  00000540
C                         0=PROCESS CURRENT DATA SET AND CONTINUE          00000550
C                         1=PROCESS CURRENT DATA SET AND STOP              00000560
C                         2=SKIP CURRENT DATA SET AND CONTINUE             00000570
C                         N.B. ISTOP=2 WILL CHANGE ANY NAMELIST PARAMETERS00000580
C                         WHICH WERE READ.  DEFAULT=1                      00000590
C                                                                          00000600
C     PRINTED OUTPUT IS WRITTEN TO LUOUT.  IF ICTYP IS SET TO 3            00000610
C     A STANDARD FORMAT FILE FOR CONTOURING IS WRITTEN TO LUPLT.           00000620
C                                                                          00000630
      INTEGER ORDER,OLDORD                                                 00000640
      REAL L1,L2,K21,K23,K21K23                                            00000650
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,       00000660
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,              00000670
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                     00000680
C                                                                          00000690
C######### VAX/VMS SITE DEPENDENT CODE FOLLOWS ##########################00000700
C                                                                          00000710
C     COMMON BLOCK 'VAX' USED TO ALLOW FORTRAN OPTIMIZATION WITH           00000720
C     NAMELIST SIMULATOR.                                                  00000730
C                                                                          00000740
      COMMON /VAX/ ORDER,X1,Y1,X2,Y2,DR,ICTYP,ISTOP                        00000750
C                                                                          00000760
C##########################################################################00000770
C                                                                          00000780
      DIMENSION X(120),Y(120),ARG(100),XABS(20),                           00000790
     1RWORK1(100),RWORK2(100),WEIGHT(20,20),POT(120,120)                   00000800
      CHARACTER*80 TITLE                                                   00000810
C     NAMELIST /INPUT/ TOL,ORDER,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,        00000820
C    1C1,C2,C3,F1,F2,D1,T1,L1,D2,T2,L2,X1,Y1,X2,Y2,DX,DY,DR,ICTYP,ISTOP   00000830
```

```
      PARAMETER (NBMAX=100,NMAX=120)                                    00000840
      PARAMETER (HALF=0.5,TWO=2.0)                                      00000850
      PARAMETER (LUIN=9,LUOUT=10,LULOG=6)                               00000860
      DATA OLDORD/0/,LUPLT/10/                                          00000870
C                                                                       00000880
C########## VAX/VMS SITE DEPENDENT CODE FOLLOWS ########################00000890
C                                                                       00000900
C     VAX NAMELIST SIMULATOR CALLS                                      00000910
C                                                                       00000920
      CALL NAMSLIST('INPUT','TOL',%DESCR(TOL),'ORDER',%DESCR(ORDER))    00000930
      CALL NAMSCONT('WIDTH',%DESCR(WIDTH),'OFFSET',%DESCR(OFFSET))      00000940
      CALL NAMSCONT('SIGMA1',%DESCR(SIGMA1),'SIGMA2',%DESCR(SIGMA2))    00000950
      CALL NAMSCONT('SIGMA3',%DESCR(SIGMA3),'C1',%DESCR(C1))            00000960
      CALL NAMSCONT('C2',%DESCR(C2),'C3',%DESCR(C3),'F1',%DESCR(F1))    00000970
      CALL NAMSCONT('F2',%DESCR(F2),'D1',%DESCR(D1),'T1',%DESCR(T1))    00000980
      CALL NAMSCONT('L1',%DESCR(L1),'D2',%DESCR(D2),'T2',%DESCR(T2))    00000990
      CALL NAMSCONT('L2',%DESCR(L2),'X1',%DESCR(X1),'Y1',%DESCR(Y1))    00001000
      CALL NAMSCONT('X2',%DESCR(X2),'Y2',%DESCR(Y2),'DX',%DESCR(DX))    00001010
      CALL NAMSCONT('DY',%DESCR(DY),'DR',%DESCR(DR))                    00001020
      CALL NAMSCONT('ICTYP',%DESCR(ICTYP),'ISTOP',%DESCR(ISTOP))        00001030
C                                                                       00001040
C########################################################################00001050
C                                                                       00001060
C                                                                       00001070
C---- SET PARAMETERS                                                    00001080
      TOL=1.0E-7                                                        00001090
      ORDER=5                                                           00001100
      WIDTH=1.0                                                         00001110
      OFFSET=0.0                                                        00001120
      SIGMA1=1.0                                                        00001130
      SIGMA2=1.0                                                        00001140
      SIGMA3=1.0                                                        00001150
      C1=1.0                                                            00001160
      C2=1.0                                                            00001170
      C3=1.0                                                            00001180
      F1=1000.                                                          00001190
      F2=1000.                                                          00001200
      D1=1.0                                                            00001210
      T1=1.0                                                            00001220
      L1=1.0                                                            00001230
      D2=1.0                                                            00001240
      T2=1.0                                                            00001250
      L2=1.0                                                            00001260
      X1=0.0                                                            00001270
      Y1=-2.5                                                           00001280
      X2=0.0                                                            00001290
      Y2=2.5                                                            00001300
      DX=0.0                                                            00001310
      DY=0.0                                                            00001320
      DR=0.25                                                           00001330
      ICTYP=1                                                           00001340
      ISTOP=1                                                           00001350
C                                                                       00001360
C---- INPUT TITLE                                                       00001370
   10 READ(LUIN,1000,END=999) TITLE                                     00001380
 1000 FORMAT(A80)                                                       00001390
C                                                                       00001400
```

```
C---- INPUT PARAMETERS                                                  00001410
C     READ(LUIN,INPUT,END=999)                                          00001420
C                                                                       00001430
C######### VAX/VMS SITE DEPENDENT CODE FOLLOWS ##########################00001440
C                                                                       00001450
C     VAX NAMELIST SIMULATOR CALLS                                      00001460
C                                                                       00001470
      CALL NAMSREAD(LUIN,'INPUT',*999,*999)                             00001480
C                                                                       00001490
C#########################################################################00001500
C                                                                       00001510
C                                                                       00001520
C---- CHECK FOR DATA SET TO BE SKIPPED                                  00001530
      IF(ISTOP .EQ. 2) GO TO 10                                         00001540
C                                                                       00001550
C---- CHECK MODEL PARAMETERS                                            00001560
      CALL CHECK(LUOUT,TITLE,ORDER,DR,DX,DY,ICTYP,ISTOP,IER)            00001570
      IF(IER .EQ. 0) GO TO 20                                           00001580
      WRITE(LULOG,1010) TITLE                                           00001590
 1010 FORMAT('** INPUT PARAMETER ERROR - DETAILS IN OUTPUT FILE'/       00001600
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00001610
     2A80//)                                                            00001620
      GO TO 10                                                          00001630
C                                                                       00001640
C---- INCREMENT LUPLT IF PLOTTING OR STANDARD FILE ARE OUTPUT           00001650
   20 IF(ICTYP .EQ. 2 .OR. ICTYP .EQ. 4) LUPLT=LUPLT+1                  00001660
C                                                                       00001670
C---- BUILD GAUSSIAN INTEGRATOR IF ORDER HAS BEEN CHANGED               00001680
      IF(ORDER .EQ. OLDORD) GO TO 30                                    00001690
      CALL BUILD(XABS,WEIGHT,ORDER)                                     00001700
      OLDORD=ORDER                                                      00001710
C                                                                       00001720
C---- COMPUTE SOURCE INTENSITY FACTORS AND CONDUCTIVITY CONTRASTS       00001730
   30 S1=(C2-C1)*F1                                                     00001740
      S2=(C3-C2)*F2                                                     00001750
      K21=(SIGMA1-SIGMA2)/(SIGMA1+SIGMA2)                               00001760
      K23=(SIGMA3-SIGMA2)/(SIGMA3+SIGMA2)                               00001770
      K21K23=K21*K23                                                    00001780
      WIDTH2=HALF*WIDTH                                                 00001790
      TWOW=TWO*WIDTH                                                    00001800
      HL1=HALF*L1                                                       00001810
      HL2=HALF*L2                                                       00001820
      HT1=HALF*T1                                                       00001830
      HT2=HALF*T2                                                       00001840
      ZMID1=D1+HALF*T1                                                  00001850
      ZMID2=D2+HALF*T2                                                  00001860
C                                                                       00001870
C---- DETERMINE BOUNDS FOR HANKEL TRANSFORM TABLE COMPUTATION           00001880
      XSQMAX=MAX((X2-HL1)**2,(X2+HL1)**2,(X1-HL1)**2,(X1+HL1)**2,       00001890
     1(X2-OFFSET-HL2)**2,(X2-OFFSET+HL2)**2,(X1-OFFSET-HL2)**2,         00001900
     2(X1-OFFSET+HL2)**2)                                               00001910
      ZMAX=MAX(D1+T1,D2+T2)                                             00001920
      BMAX=SQRT(XSQMAX+ZMAX**2)                                         00001930
      BMIN=MIN(D1,D2)                                                   00001940
      NB=INT(5.*ALOG(BMAX/BMIN))+2                                      00001950
C                                                                       00001960
C---- CHECK FOR NB TOO LARGE                                            00001970
```

```
      IF(NB .LE. NBMAX) GO TO 40                                        00001980
C                                                                       00001990
C---- WRITE ERROR MESSAGE                                               00002000
      WRITE(LULOG,1020) TITLE                                           00002010
 1020 FORMAT('** HANKEL TRANSFORM TABLE ERROR - DETAILS IN OUTPUT FILE'/00002020
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00002030
     2A80//)                                                            00002040
      WRITE(LUOUT,1030) TITLE,NB,NBMAX,BMIN,BMAX                        00002050
 1030 FORMAT(1H1,'** HANKEL TRANSFORM TABLE ERROR'/                      00002060
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00002070
     23X,A80//3X,'NB=',I5,' EXCEEDS NBMAX=',I3,' BMIN=',F10.3,          00002080
     3' BMAX=',F10.3)                                                   00002090
      GO TO 10                                                          00002100
C                                                                       00002110
C---- CHECK FOR TYPE OF COMPUTATION                                     00002120
   40 IF(ICTYP .GE. 3) GO TO 50                                         00002130
C                                                                       00002140
C---- DO TRAVERSE CALCULATION                                           00002150
C                                                                       00002160
C---- COMPUTE OBSERVATION POINT SPACING                                 00002170
      CALL TSPACE(X1,Y1,X2,Y2,DR,NMAX,X,Y,NR)                           00002180
C                                                                       00002190
C---- CHECK FOR TOO MANY POINTS                                         00002200
      IF(NR .LE. NMAX) GO TO 60                                         00002210
C                                                                       00002220
C---- WRITE ERROR MESSAGE                                               00002230
      WRITE(LULOG,1040) TITLE                                           00002240
 1040 FORMAT('** TRAVERSE LENGTH ERROR - DETAILS IN OUTPUT FILE'/       00002250
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00002260
     2A80//)                                                            00002270
      WRITE(LUOUT,1050) TITLE,NR,NMAX,X1,Y1,X2,Y2,DR                    00002280
 1050 FORMAT(1H1,'** TRAVERSE LENGTH ERROR'/                            00002290
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00002300
     23X,A80/3X,'TRAVERSE: NR=',I5,' EXCEEDS NMAX=',I3,                 00002310
     3' X1=',F10.3,' Y1=',F10.3,' X2=',F10.3,' Y2=',F10.3,              00002320
     4' DR=',F10.6)                                                     00002330
      GO TO 10                                                          00002340
C                                                                       00002350
C---- COMPUTE POTENTIALS                                                00002360
   60 CALL TRAVERSE(X,Y,NR,XABS,WEIGHT,ORDER,NB,BMAX,ARG,RWORK1,RWORK2, 00002370
     1POT,LUOUT,LULOG,TITLE,IER)                                        00002380
C                                                                       00002390
C---- CHECK FOR ERROR                                                   00002400
      IF(IER .NE. 0) GO TO 90                                           00002410
C                                                                       00002420
C---- WRITE OUTPUT                                                      00002430
      CALL OUTRAV(LUOUT,TITLE,ORDER,X,Y,DR,NR,POT)                      00002440
C                                                                       00002450
C---- WRITE TRAVERSE PLOTTING FILE                                      00002460
      IF(ICTYP .EQ. 2) CALL OUTPLOT(LUPLT,TITLE,ORDER,X,Y,DR,NR,POT)    00002470
      GO TO 90                                                          00002480
C                                                                       00002490
C---- DO GRID CALCULATION                                               00002500
   50 CALL GSPACE(X1,X2,DX,NMAX,X,NX)                                   00002510
      IF(NX .LE. NMAX) GO TO 70                                         00002520
      WRITE(LULOG,1060) TITLE                                           00002530
 1060 FORMAT('** GRID X SIZE ERROR - DETAILS IN OUTPUT FILE'/           00002540
```

```
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//          00002550
     2A80//)                                                         00002560
      WRITE(LUOUT,1070) TITLE,NX,NMAX,X1,X2,DX                       00002570
 1070 FORMAT(1H1,'** GRID X SIZE ERROR'/                             00002580
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//         00002590
     23X,A80//3X,'NX=',I5,' EXCEEDS NMAX=',I3,                      00002600
     3' X1=',F10.3,' X2=',F10.3,' DX=',F10.6)                       00002610
      GO TO 10                                                       00002620
   70 CALL GSPACE(Y1,Y2,DY,NMAX,Y,NY)                                00002630
C                                                                    00002640
C---- TEST FOR TOO MANY POINTS                                       00002650
      IF(NY .LE. NMAX) GO TO 80                                      00002660
      WRITE(LULOG,1080) TITLE                                        00002670
 1080 FORMAT('** GRID Y SIZE ERROR - DETAILS IN OUTPUT FILE'/        00002680
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//         00002690
     2A80//)                                                         00002700
      WRITE(LUOUT,1090) TITLE,NY,NMAX,Y1,Y2,DY                       00002710
 1090 FORMAT(1H1,'** GRID Y SIZE ERROR'/                             00002720
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//         00002730
     23X,A80//3X,'NY=',I5,' EXCEEDS NMAX=',I3,                      00002740
     3' Y1=',F10.3,' Y2=',F10.3,' DY=',F10.6)                       00002750
      GO TO 10                                                       00002760
C                                                                    00002770
C---- COMPUTE POTENTIALS                                             00002780
   80 CALL GRID(X,Y,NX,NY,XABS,WEIGHT,ORDER,NB,BMAX,ARG,RWORK1,RWORK2, 00002790
     1POT,LUOUT,LULOG,TITLE,IER)                                     00002800
C                                                                    00002810
C---- CHECK FOR ERROR                                                00002820
      IF(IER .NE. 0) GO TO 90                                        00002830
C                                                                    00002840
C---- WRITE OUTPUT                                                   00002850
      CALL OUTGRID(LUOUT,TITLE,ORDER,X,Y,DX,DY,NX,NY,POT)            00002860
C                                                                    00002870
C---- WRITE STANDARD FILE FOR PLOTTING IF REQUESTED                  00002880
      IF(ICTYP .EQ. 4) CALL OUTSTD(LUPLT,TITLE,X,Y,DX,DY,NX,NY,POT)  00002890
C                                                                    00002900
C---- TEST FOR STOP FLAG SET                                         00002910
   90 IF(ISTOP .NE. 1) GO TO 10                                      00002920
  999 IF(LUPLT .GT. 10) WRITE(LUOUT,1100) LUPLT                      00002930
 1100 FORMAT(1H1/' STANDARD OUTPUT FILES WRITTEN: FOR011.DAT',       00002940
     1' THROUGH FOR',I3,'.DAT')                                      00002950
      CALL EXIT                                                      00002960
      END                                                            00002970
      SUBROUTINE CHECK(LUOUT,TITLE,ORDER,DR,DX,DY,ICTYP,ISTOP,IER)   00002980
C                                                                    00002990
C---- ROUTINE TO CHECK MODEL PARAMETERS                              00003000
      INTEGER ORDER                                                  00003010
      REAL L1,L2,K21,K23,K21K23                                      00003020
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3, 00003030
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,        00003040
     2TWOW,HL1,HL2,HT1,TH2,ZMID1,ZMID2                               00003050
      CHARACTER*80 TITLE                                             00003060
C                                                                    00003070
C---- CLEAR ERROR FLAG                                               00003080
      IER=0                                                          00003090
C                                                                    00003100
C---- CHECK HANKEL TRANSFORM TOLERANCE                               00003110
```

```
      IF(TOL .GE. 0.0) GO TO 10                                          00003120
      WRITE(LUOUT,1000)                                                  00003130
 1000 FORMAT(1H1)                                                        00003140
      IER=1                                                              00003150
      WRITE(LUOUT,1010)                                                  00003160
 1010 FORMAT(' HANKEL TRANSFORM TOLERANCE NEGATIVE')                     00003170
C                                                                        00003180
C---- CHECK INTEGRATION ORDER                                            00003190
   10 IF(ORDER .GE. 1 .AND. ORDER .LE. 20) GO TO 20                      00003200
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003210
      IER=1                                                              00003220
      WRITE(LUOUT,1020)                                                  00003230
 1020 FORMAT(' INTEGRATION ORDER OUT OF BOUNDS (1-20)')                  00003240
C                                                                        00003250
C---- CHECK DIKE WIDTH                                                   00003260
   20 IF(WIDTH .GT. 0.0) GO TO 30                                        00003270
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003280
      IER=1                                                              00003290
      WRITE(LUOUT,1030)                                                  00003300
 1030 FORMAT(' DIKE WIDTH  NON-POSITIVE')                                00003310
C                                                                        00003320
C---- CHECK CONDUCTIVITIES                                               00003330
   30 IF(SIGMA1 .GE. 0.0 .AND. SIGMA2 .GE. 0.0 .AND. SIGMA3 .GE. 0.0)    00003340
     1GO TO 40                                                           00003350
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003360
      IER=1                                                              00003370
      WRITE(LUOUT,1040)                                                  00003380
 1040 FORMAT(' SOME CONDUCTIVITY NON-NEGATIVE')                          00003390
C                                                                        00003400
C---- CHECK CROSS COUPLING CONTRASTS                                     00003410
   40 IF(C1 .NE. C2 .OR. C2 .NE. C3) GO TO 50                            00003420
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003430
      IER=1                                                              00003440
      WRITE(LUOUT,1050)                                                  00003450
 1050 FORMAT(' NO CROSS COUPLING COEFFICIENT CONTRAST')                  00003460
C                                                                        00003470
C---- CHECK DRIVING FORCE                                                00003480
   50 IF(F1 .NE. 0.0 .OR. F2 .NE. 0.0) GO TO 60                          00003490
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003500
      IER=1                                                              00003510
      WRITE(LUOUT,1060)                                                  00003520
 1060 FORMAT(' BOTH DRIVING FORCES ZERO')                                00003530
C                                                                        00003540
C---- CHECK PATCH PARAMETERS                                             00003550
   60 IF(D1 .GT. 0.0 .AND. T1 .GT. 0.0 .AND. L1 .GT. 0.0 .AND.           00003560
     1D2 .GT. 0.0 .AND. T2 .GT. 0.0 .AND. L2 .GT. 0.0) GO TO 70          00003570
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003580
      IER=1                                                              00003590
      WRITE(LUOUT,1070)                                                  00003600
 1070 FORMAT(' SOME PATCH PARAMETER NON-POSITIVE')                       00003610
C                                                                        00003620
C---- CHECK TRAVERSE SPACING                                             00003630
   70 IF(ICTYP .NE. 1 .OR. ICTYP .NE. 2) GO TO 80                        00003640
      IF(DR .GT. 0.0) GO TO 80                                           00003650
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                   00003660
      IER=1                                                              00003670
      WRITE(LUOUT,1080)                                                  00003680
```

```
 1080 FORMAT(' TRAVERSE POINT SPACING NON-POSITIVE')                    00003690
C                                                                       00003700
C---- CHECK GRID SPACING                                                00003710
   80 IF(ICTYP .NE. 3 .OR. ICTYP .NE. 4) GO TO 90                       00003720
      IF(DX .GT. 0.0 .AND. DY .GT. 0.0) GO TO 90                        00003730
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                  00003740
      IER=1                                                             00003750
      WRITE(LUOUT,1090)                                                 00003760
 1090 FORMAT(' SOME GRID POINT SPACING NON-POSITIVE')                   00003770
C                                                                       00003780
C---- CHECK COMPUTATION TYPE                                            00003790
   90 IF(ICTYP .GE. 1 .AND. ICTYP .LE. 4) GO TO 100                     00003800
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                  00003810
      IER=1                                                             00003820
      WRITE(LUOUT,1100)                                                 00003830
 1100 FORMAT(' NON-ALLOWED COMPUTATION TYPE')                           00003840
C                                                                       00003850
C---- CHECK ACTION FLAG                                                 00003860
  100 IF(ISTOP .GE. 0 .AND. ISTOP .LE. 2) GO TO 110                     00003870
      IF(IER .EQ. 0) WRITE(LUOUT,1000)                                  00003880
      IER=1                                                             00003890
      WRITE(LUOUT,1110)                                                 00003900
 1110 FORMAT(' NON-ALLOWED ACTION FLAG')                                00003910
  110 IF(IER .EQ. 0) RETURN                                             00003920
      WRITE(LUOUT,1120) TITLE,TOL,ORDER,WIDTH,OFFSET,SIGMA1,SIGMA2,     00003930
     1SIGMA3,C1,C2,C3,D1,D2,T1,T2,L1,L2,F1,F2,DR,DX,DY,ICTYP,ISTOP      00003940
 1120 FORMAT(/1X,A80/                                                   00003950
     1' TOL=',E10.4,' ORDER=',I5,' WIDTH=',F10.4,' OFFSET=',            00003960
     2F10.4/' SIGMA1=',F10.4,' SIGMA2=',F10.4,' SIGMA3=',F10.4/         00003970
     3' C1    =',F10.4,' C2    =',F10.4,' C3    =',F10.4/             00003980
     4' D1=',F10.4,' D2=',F10.4/' T1=',F10.4,' T2=',F10.4/              00003990
     5' L1=',F10.4,' L2=',F10.4/' F1=',F10.4,' F2=',F10.4/              00004000
     6' DR=',F10.4,' DX=',F10.4,' DY=',F10.4/                           00004010
     7' ICTYP=',I5,' ISTOP=',I5)                                        00004020
      RETURN                                                            00004030
      END                                                               00004040
      SUBROUTINE BUILD(XABS,WEIGHT,ORDER)                               00004050
C                                                                       00004060
C---- ROUTINE TO BUILD INTEGRATOR WEIGHT MATRIX AND ABSCISSA VECTOR     00004070
C                                                                       00004080
      INTEGER ORDER,ORDER2                                              00004090
      DIMENSION XABS(1),WEIGHT(ORDER,ORDER)                             00004100
      DOUBLE PRECISION GX(20),GW(20)                                    00004110
C                                                                       00004120
C---- COMPUTE 1-D GAUSSIAN COEFFICIENTS AND ABSCISSAS                   00004130
      CALL GAUSS(GX,GW,ORDER)                                           00004140
C                                                                       00004150
C---- FORM 2-D INTEGRATOR                                               00004160
      ORDER2=(ORDER+1)/2                                                00004170
      DO 10 I=1,ORDER2                                                  00004180
      XABS(I)=-GX(I)                                                    00004190
      XABS(ORDER-I+1)=GX(I)                                             00004200
      DO 10 J=1,ORDER2                                                  00004210
      WEIGHT(I,J)=GW(I)*GW(J)                                           00004220
      WEIGHT(I,ORDER-J+1)=WEIGHT(I,J)                                   00004230
      WEIGHT(ORDER-I+1,J)=WEIGHT(I,J)                                   00004240
   10 WEIGHT(ORDER-I+1,ORDER-J+1)=WEIGHT(I,J)                           00004250
```

```
      RETURN                                                            00004260
      END                                                               00004270
      SUBROUTINE GSPACE(X1,X2,DX,NMAX,X,NX)                             00004280
C                                                                       00004290
C---- SUBROUTINE TO GENERATE EQUALLY SPACED DATA POINTS PARALLEL AXIS    00004300
C                                                                       00004310
      DIMENSION X(1)                                                    00004320
C                                                                       00004330
C---- REVERSE ORDER OF COORDINATES IF NECESSARY                          00004340
      IF(X1 .LT. X2) GO TO 10                                           00004350
      T=X1                                                              00004360
      X1=X2                                                             00004370
      X2=T                                                              00004380
   10 I=1                                                               00004390
      X(1)=X1                                                           00004400
C                                                                       00004410
C---- TEST FOR TOO MANY DATA POINTS                                     00004420
   20 IF(I .GE. NMAX+1) GO TO 30                                        00004430
      T=X(I)+DX                                                         00004440
C                                                                       00004450
C---- TEST FOR COMPLETION                                               00004460
      IF(T-X2 .GT. 1.0E-5*(X2-X1)) GO TO 30                             00004470
      I=I+1                                                             00004480
      X(I)=T                                                            00004490
      GO TO 20                                                          00004500
   30 NX=I                                                              00004510
      RETURN                                                            00004520
      END                                                               00004530
      SUBROUTINE TSPACE(X1,Y1,X2,Y2,DR,NMAX,X,Y,N)                      00004540
C                                                                       00004550
C---- SUBROUTINE TO COMPUTE EQUALLY SPACED COORDINATES ALONG A LINE     00004560
C                                                                       00004570
      DIMENSION X(1),Y(1)                                               00004580
C                                                                       00004590
C---- COMPUTE TOTAL DISTANCE AND ANGLE                                  00004600
      R2=SQRT((X2-X1)**2+(Y2-Y1)**2)                                    00004610
      R=0.0                                                             00004620
      THETA=ATAN2((Y2-Y1),(X2-X1))                                      00004630
      S=SIN(THETA)                                                      00004640
      C=COS(THETA)                                                      00004650
      X(1)=X1                                                           00004660
      Y(1)=Y1                                                           00004670
      I=1                                                               00004680
C                                                                       00004690
C---- TEST FOR TOO MANY DATA POINTS                                     00004700
   10 IF(I .GE. NMAX+1) GO TO 20                                        00004710
C                                                                       00004720
C---- TEST FOR ENOUGH POINTS                                            00004730
      R=R+DR                                                            00004740
      IF(R-R2 .GT. 1.0E-5*R2) GO TO 20                                  00004750
      I=I+1                                                             00004760
      X(I)=X1+R*C                                                       00004770
      Y(I)=Y1+R*S                                                       00004780
      GO TO 10                                                          00004790
   20 N=I                                                               00004800
      RETURN                                                            00004810
      END                                                               00004820
```

```
      SUBROUTINE TRAVERSE(X,Y,NR,XABS,WEIGHT,ORDER,NB,BMAX,ARG,          00004830
     1RWORK1,RWORK2,POT,LUOUT,LULOG,TITLE,IER)                           00004840
C                                                                        00004850
C---- ROUTINE TO COMPUTE POTENTIALS ALONG TRAVERSE POINTS                00004860
C                                                                        00004870
      INTEGER ORDER                                                      00004880
      REAL L1,L2,K21,K23,K21K23                                          00004890
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,     00004900
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,            00004910
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                   00004920
      DIMENSION X(1),Y(1),XABS(1),WEIGHT(ORDER,ORDER),ARG(1),RWORK1(1),  00004930
     1RWORK2(1),POT(1),AA1(100),BB1(100),CC1(100),AA2(100),              00004940
     2BB2(100),CC2(100)                                                  00004950
      CHARACTER*80 TITLE                                                 00004960
      PARAMETER (EIGHT_PI=25.13274132)                                   00004970
C                                                                        00004980
C---- LOOP OVER OBSERVATION POINTS                                       00004990
      DO 10 I=1,NR                                                       00005000
      XO=X(I)                                                            00005010
      YO=Y(I)                                                            00005020
C                                                                        00005030
C---- FORM GREEN'S FUNCTION TABLE                                        00005040
      CALL GTABLE(NB,BMAX,ARG,RWORK1,RWORK2,AA1,BB1,                     00005050
     1CC1,AA2,BB2,CC2,IER)                                               00005060
C                                                                        00005070
C---- CHECK FOR ERROR                                                    00005080
      IF(IER .EQ. 0) GO TO 20                                            00005090
      WRITE(LULOG,1000) TITLE                                            00005100
 1000 FORMAT('** TRAVERSE HANKEL TRANSFORM ERROR - ',                    00005110
     1'SEE OUTPUT FILE FOR DETAILS'/                                     00005120
     23X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//             00005130
     3A80//)                                                             00005140
      WRITE(LUOUT,1010) TITLE,IER,I,YO                                   00005150
 1010 FORMAT(1H1,'** TRAVERSE HANKEL TRANSFORM ERROR'/                   00005160
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//             00005170
     23X,A80//3X,'ERROR ON CALL TO HANKEL: IER=',I2,' I=',I6,' YO=',     00005180
     3E10.4)                                                             00005190
      RETURN                                                             00005200
C                                                                        00005210
C---- INTEGRATE OVER SOURCE PLANE                                        00005220
   20 CALL INTEGRATE(XO,ORDER,WEIGHT,NB,XABS,ARG,RWORK1,                 00005230
     1RWORK2,AA1,BB1,CC1,AA2,BB2,CC2,SUM1,SUM2)                          00005240
   10 POT(I)=(L1*T1*SUM1+L2*T2*SUM2)/EIGHT_PI                            00005250
      RETURN                                                             00005260
      END                                                                00005270
      SUBROUTINE GRID(X,Y,NX,NY,XABS,WEIGHT,ORDER,NB,BMAX,ARG,           00005280
     1RWORK1,RWORK2,POT,LUOUT,LULOG,TITLE,IER)                           00005290
C                                                                        00005300
C---- ROUTINE TO COMPUTE POTENTIALS OVER GRID POINTS                     00005310
C                                                                        00005320
      INTEGER ORDER                                                      00005330
      REAL L1,L2,K21,K23,K21K23                                          00005340
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,     00005350
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,            00005360
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                   00005370
      DIMENSION X(1),Y(1),XABS(1),WEIGHT(ORDER,ORDER),ARG(1),RWORK1(1),  00005380
     1RWORK2(1),POT(NX,NY),AA1(100),BB1(100),CC1(100),AA2(100),          00005390
```

```
     2BB2(100),CC2(100)                                                 00005400
      CHARACTER*80 TITLE                                                00005410
      PARAMETER (EIGHT_PI=25.13274132)                                  00005420
C                                                                       00005430
C---- LOOP OVER Y OBSERVATION POINTS                                    00005440
      DO 10 I=1,NY                                                      00005450
      YO=Y(I)                                                           00005460
C                                                                       00005470
C---- FORM GREEN'S FUNCTION TABLE                                       00005480
      CALL GTABLE(NB,BMAX,ARG,RWORK1,RWORK2,AA1,BB1,                    00005490
     1CC1,AA2,BB2,CC2,IER)                                              00005500
C                                                                       00005510
C---- CHECK FOR ERROR                                                   00005520
      IF(IER .EQ. 0) GO TO 20                                           00005530
      WRITE(LULOG,1000) TITLE                                           00005540
 1000 FORMAT('** GRID HANKEL TRANSFORM ERROR - ',                       00005550
     1'DETAILS IN OUTPUT FILE'/                                         00005560
     23X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00005570
     3A80//)                                                            00005580
      WRITE(LUOUT,1010) TITLE,IER,I,YO                                  00005590
 1010 FORMAT(1H1,'** GRID HANKEL TRANSFORM ERROR'/                      00005600
     13X,'EXECUTION TERMINATED FOR THE FOLLOWING DATA SET'//            00005610
     23X,A80/3X,'ERROR ON CALL TO HANKEL: IER=',I2,' I=',I6,' YO=',     00005620
     3E10.4)                                                            00005630
      RETURN                                                            00005640
C                                                                       00005650
C---- LOOP OVER X OBSERVATION POINTS                                    00005660
   20 DO 10 J=1,NX                                                      00005670
      XO=X(J)                                                           00005680
C                                                                       00005690
C---- INTEGRATE OVER SOURCE PLANE                                       00005700
      CALL INTEGRATE(XO,ORDER,WEIGHT,NB,XABS,ARG,RWORK1,                00005710
     1RWORK2,AA1,BB1,CC1,AA2,BB2,CC2,SUM1,SUM2)                         00005720
   10 POT(J,I)=(L1*T1*SUM1+L2*T2*SUM2)/EIGHT_PI                         00005730
      RETURN                                                            00005740
      END                                                               00005750
      SUBROUTINE GTABLE(NB,BMAX,ARG,RWORK1,RWORK2,                      00005760
     1AA1,BB1,CC1,AA2,BB2,CC2,IER)                                      00005770
C                                                                       00005780
C---- ROUTINE TO COMPUTE GREEN'S FUNCTION TABLE FOR INTEGRATION         00005790
C                                                                       00005800
      REAL L1,L2,K21,K23,K21K23                                         00005810
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00005820
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00005830
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00005840
      DIMENSION ARG(1),RWORK1(1),RWORK2(1),AA1(1),BB1(1),CC1(1),        00005850
     1AA2(1),BB2(1),CC2(1),P(100),S(100),D(2)                           00005860
      COMPLEX ZANS(100),ZWORK(283)                                      00005870
      EXTERNAL G1,G2,G3                                                 00005880
      PARAMETER (NREL=1,NTOL=1,NORD=0)                                  00005890
      DATA D/2*0.0/                                                     00005900
C                                                                       00005910
C---- (XO,YO) IN REGION 1                                               00005920
      IF(YO .LT. -WIDTH2) CALL HANKEL(BMAX,NB,NREL,TOL,NTOL,NORD,G1,    00005930
     1IJREL,ZWORK,ZANS,ARG,NOFUN1,IER)                                  00005940
C                                                                       00005950
C---- (XO,YO) IN REGION 2                                               00005960
```

```
      IF(YO .GE. -WIDTH2 .AND. YO .LE. WIDTH2) CALL HANKEL(BMAX,NB,NREL,00005970
     1TOL,NTOL,NORD,G2,IJREL,ZWORK,ZANS,ARG,NOFUN1,IER)                 00005980
C                                                                       00005990
C---- (XO,YO) IN REGION 3                                              00006000
      IF(YO .GT. WIDTH2) CALL HANKEL(BMAX,NB,NREL,TOL,NTOL,NORD,G3,     00006010
     1IJREL,ZWORK,ZANS,ARG,NOFUN1,IER)                                  00006020
C                                                                       00006030
C---- CHECK FOR ERRORS                                                 00006040
      IF(IER .NE. 0) RETURN                                             00006050
C                                                                       00006060
C---- SEPARATE OUT GREEN'S FUNCTIONS                                   00006070
      DO 10 J=1,NB                                                      00006080
      RWORK1(J)=REAL(ZANS(J))                                           00006090
      RWORK2(J)=AIMAG(ZANS(J))                                          00006100
C                                                                       00006110
C---- CONVERT ARGUMENTS TO LOGARITHMIC SPACING                         00006120
   10 ARG(J)=ALOG(ARG(J))                                               00006130
C                                                                       00006140
C---- COMPUTE INTERPOLATION COEFFICIENTS                               00006150
      IF(S1 .NE. 0.0) CALL SPLIN1(NB,0,ARG,RWORK1,AA1,BB1,CC1,0,D,P,S)  00006160
      IF(S2 .NE. 0.0) CALL SPLIN1(NB,0,ARG,RWORK2,AA2,BB2,CC2,0,D,P,S)  00006170
      RETURN                                                            00006180
      END                                                               00006190
      SUBROUTINE INTEGRATE(XO,ORDER,WEIGHT,NB,XABS,ARG,RWORK1,          00006200
     1RWORK2,AA1,BB1,CC1,AA2,BB2,CC2,SUM1,SUM2)                         00006210
C                                                                       00006220
C---- ROUTINE TO DO GAUSSIAN QUADRATURE OVER SOURCE PLANE              00006230
C                                                                       00006240
      INTEGER ORDER                                                     00006250
      REAL L1,L2,K21,K23,K21K23                                         00006260
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00006270
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00006280
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00006290
      DIMENSION WEIGHT(ORDER,ORDER),XABS(1),ARG(1),RWORK1(1),           00006300
     1RWORK2(1),AA1(1),BB1(1),CC1(1),AA2(1),BB2(1),CC2(1)               00006310
      PARAMETER (ZERO=0.0)                                              00006320
C                                                                       00006330
      SUM1=ZERO                                                         00006340
      SUM2=ZERO                                                         00006350
      GI1=ZERO                                                          00006360
      GI2=ZERO                                                          00006370
C                                                                       00006380
C---- LOOP OVER XP                                                     00006390
      DO 10 K=1,ORDER                                                   00006400
      XP1=HL1*XABS(K)                                                   00006410
      XP2=HL2*XABS(K)+OFFSET                                            00006420
      XT1=(XO-XP1)**2                                                   00006430
      XT2=(XO-XP2)**2                                                   00006440
C                                                                       00006450
C---- LOOP OVER ZP                                                     00006460
      DO 10 L=1,ORDER                                                   00006470
      IF(S1 .EQ. 0.0) GO TO 20                                          00006480
      ZT1=(HT1*XABS(L)+ZMID1)**2                                        00006490
      R=ALOG(SQRT(XT1+ZT1))                                             00006500
C                                                                       00006510
C---- INTERPOLATE SOURCE PLANE 1 GREEN'S FUNCTION                      00006520
      CALL SPOINT(NB,ARG,RWORK1,AA1,BB1,CC1,R,GI1)                      00006530
```

```
      SUM1=SUM1+S1*GI1*WEIGHT(K,L)                                      00006540
C                                                                       00006550
   20 IF(S2 .EQ. 0.0) GO TO 10                                          00006560
      ZT2=(HT2*XABS(L)+ZMID2)**2                                        00006570
      R=ALOG(SQRT(XT2+ZT2))                                             00006580
C                                                                       00006590
C---- INTERPOLATE SOURCE PLANE 2 GREEN'S FUNCTION                       00006600
      CALL SPOINT(NB,ARG,RWORK2,AA2,BB2,CC2,R,GI2)                      00006610
      SUM2=SUM2+S2*GI2*WEIGHT(K,L)                                      00006620
   10 CONTINUE                                                          00006630
      RETURN                                                            00006640
      END                                                               00006650
      COMPLEX FUNCTION G1(LBD)                                          00006660
C                                                                       00006670
C---- GREEN'S FUNCTION INTEGRAND FOR OBSERVATION POINT IN REGION 1      00006680
C     YO .LT. -WIDTH/2                                                  00006690
C                                                                       00006700
      REAL L1,L2,K21,K23,K21K23,LBD                                     00006710
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00006720
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00006730
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00006740
      PARAMETER (ZERO=0.0,ONE=1.0,TWO=2.0)                              00006750
C-----------------------------------------------------------------------00006760
C                                                                       00006770
C     SET PARAMETER BIG TO THE LARGEST ALLOWABLE ARGUMENT FOR THE       00006780
C     EXPONENTIAL FUNCTION.                                             00006790
C                                                                       00006800
      PARAMETER (BIG=88.028)                                            00006810
C                                                                       00006820
C-----------------------------------------------------------------------00006830
C                                                                       00006840
C----- SET DEFAULT VALUES FOR NO SOURCE                                 00006850
C                                                                       00006860
      F11=ZERO                                                          00006870
      F12=ZERO                                                          00006880
C                                                                       00006890
C---- CHECK FOR LARGE ARGUMENT                                          00006900
      IF(TWOW*LBD .GE. BIG) GO TO 30                                    00006910
C                                                                       00006920
C---- G11 SOURCE PLANE 1 TERM (YP=-WIDTH/2)                             00006930
      R1=EXP(TWOW*LBD)                                                  00006940
      R2=K21K23-R1                                                      00006950
      R3=R1/R2                                                          00006960
      IF(S1 .EQ. 0.0) GO TO 10                                          00006970
      B1=(K23-TWO*K21K23)/R2+K21*R3                                     00006980
      DB1=TWO*LBD*(K23-K21K23)/R2                                       00006990
      F11=EXP(LBD*(YO+WIDTH2))*(-LBD*(ONE+B1)+DB1)                      00007000
C                                                                       00007010
C---- G12 SOURCE PLANE 2 TERM (YP=WIDTH/2)                              00007020
   10 IF(S2 .EQ. 0.0) GO TO 20                                          00007030
      B1=R3*(K21+K23-K21K23)-K21K23/R2                                  00007040
      DB1=TWO*LBD*(K23-K21K23)*R3                                       00007050
      F12=EXP(LBD*(YO-WIDTH2))*(DB1-LBD*(ONE+B1))                       00007060
   20 G1=CMPLX(F11,F12)                                                 00007070
      RETURN                                                            00007080
C                                                                       00007090
C---- LARGE ARGUMENT CALCULATION                                        00007100
```

```
   30 IF(S1 .NE. ZERO) F11=-LBD*(ONE-K21)*EXP(LBD*(YO+WIDTH2))          00007110
      IF(S2 .NE. ZERO) F13=-LBD*(ONE-K21)*(ONE+K23)*EXP(LBD*(YO-WIDTH2))00007120
      G1=CMPLX(F11,F12)                                                 00007130
      END                                                               00007140
      COMPLEX FUNCTION G2(LBD)                                          00007150
C                                                                       00007160
C---- GREEN'S FUNCTION INTEGRAND FOR OBSERVATION POINT IN REGION 2      00007170
C     -WIDTH/2 .LE. YO .LE. WIDTH/2                                     00007180
C                                                                       00007190
      REAL L1,L2,K21,K23,K21K23,LBD                                     00007200
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00007210
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00007220
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00007230
      PARAMETER (ZERO=0.0,ONE=1.0,TWO=2.0)                              00007240
C                                                                       00007250
C-----------------------------------------------------------------------00007260
C                                                                       00007270
C     SET PARAMETER BIG TO LARGEST ALLOWABLE ARGUMENT FOR THE           00007280
C     EXPONENTIAL FUNCTION.                                             00007290
C                                                                       00007300
      PARAMETER (BIG=88.028)                                            00007310
C                                                                       00007320
C-----------------------------------------------------------------------00007330
C                                                                       00007340
C---- SET DEFAULT VALUES FOR NO SOURCE                                  00007350
      F21=ZERO                                                          00007360
      F22=ZERO                                                          00007370
C                                                                       00007380
C                                                                       00007390
C---- CHECK FOR LARGE ARGUMENT                                          00007400
      IF(TWOW*LBD .GE. BIG) GOTO 30                                     00007410
C                                                                       00007420
C---- G21 SOURCE PLANE 1 TERM (YP=-WIDTH/2)                             00007430
      R1=EXP(TWOW*LBD)                                                  00007440
      R2=K21K23-R1                                                      00007450
      R3=R1/R2                                                          00007460
      IF(S1 .EQ. 0.0) GO TO 10                                          00007470
      A2=K21*R3-K21K23/R2                                               00007480
      B2=(K23-K21K23)/R2                                                00007490
      DA2=-TWO*LBD*K21*R3                                               00007500
      DB2=TWO*LBD*K23/R2                                                00007510
      F21=EXP(-LBD*(YO+WIDTH2))*(LBD*(ONE+A2)+DA2)                      00007520
     1+EXP(LBD*(YO+WIDTH2))*(DB2-LBD*B2)                                00007530
   10 IF(S2 .EQ. 0.0) GOTO 20                                           00007540
C                                                                       00007550
C---- G22 SOURCE PLANE TERM (YP=WIDTH/2)                                00007560
      A2=(K21-K21K23)/R2                                                00007570
      B2=K23*R3-K21K23/R2                                               00007580
      DA2=-TWO*LBD*K21/R2                                               00007590
      DB2=TWO*LBD*K23*R3                                                00007600
      F22=-EXP(LBD*(YO-WIDTH2))*(LBD*(ONE+B2)-DB2)                      00007610
     1+EXP(-LBD*(YO-WIDTH2))*(LBD*A2+DA2)                               00007620
   20 G2=CMPLX(F21,F22)                                                 00007630
      RETURN                                                            00007640
C                                                                       00007650
C---- LARGE ARGUMENT CALCULATION                                        00007660
   30 IF(S1 .NE. ZERO) F21=LBD*(ONE+K21)*EXP(-LBD*(YO+WIDTH2))          00007670
```

```
      IF(S2 .NE. ZERO) F22=-LBD*(ONE+K23)*EXP(LBD*(YO-WIDTH2))          00007680
      G2=CMPLX(F21,F22)                                                 00007690
      RETURN                                                            00007700
      END                                                               00007710
      COMPLEX FUNCTION G3(LBD)                                          00007720
C                                                                       00007730
C---- GREEN'S FUNCTION INTEGRAND FOR OBSERVATION POINT IN REGION 3      00007740
C     YO .GT. WIDTH/2                                                   00007750
C                                                                       00007760
      REAL L1,L2,K21,K23,K21K23,LBD                                     00007770
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00007780
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00007790
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00007800
      PARAMETER (ZERO=0.0,ONE=1.0,TWO=2.0)                              00007810
C                                                                       00007820
                                                                        00007830
C-----------------------------------------------------------------------00007840
C                                                                       00007850
C     SET PARAMETER BIG TO LARGEST ALLOWABLE ARGUMENT FOR THE           00007860
C     EXPONENTIAL FUNCTION.                                             00007870
C                                                                       00007880
      PARAMETER (BIG=88.028)                                            00007890
C                                                                       00007900
C-----------------------------------------------------------------------00007910
C                                                                       00007920
C---- SET DEFAULT VALUES FOR NO SOURCE                                  00007930
      F31=ZERO                                                          00007940
      F32=ZERO                                                          00007950
C                                                                       00007960
C---- CHECK FOR LARGE ARGUMENT                                          00007970
      IF(TWOW*LBD .GE. BIG) GOTO 30                                     00007980
C                                                                       00007990
C---- G31 SOURCE PLANE 1 TERM (YP=-WIDTH/2)                             00008000
      R1=EXP(TWOW*LBD)                                                  00008010
      R2=K21K23-R1                                                      00008020
      R3=R1/R2                                                          00008030
      IF(S1 .EQ. 0.0) GO TO 10                                          00008040
      A3=R3*(K21+K23-K21K23)-K21K23/R2                                  00008050
      DA3=-TWO*LBD*(K21-K21K23)*R3                                      00008060
      F31=EXP(-LBD*(YO+WIDTH2))*(LBD*(ONE+A3)+DA3)                      00008070
C                                                                       00008080
C---- G32 SOURCE PLANE 2 TERM (YP=WIDTH/2)                              00008090
   10 IF(S2 .EQ. 0.0) GO TO 20                                          00008100
      A3=(K21-TWO*K21K23)/R2+K23*R3                                     00008110
      DA3=-TWO*LBD*(K21-K21K23)/R2                                      00008120
      F32=EXP(-LBD*(YO-WIDTH2))*(LBD*(ONE+A3)+DA3)                      00008130
   20 G3=CMPLX(F31,F32)                                                 00008140
      RETURN                                                            00008150
C                                                                       00008160
C---- LARGE ARGUMENT CALCULATION                                        00008170
   30 IF(S1 .NE. ZERO) F31=LBD*(ONE+K21)*(ONE-K23)*EXP(-LBD*(YO+WIDTH2))00008180
      IF(S2 .NE. ZERO) F32=LBD*(ONE-K23)*EXP(-LBD*(YO-WIDTH2))          00008190
      G3=CMPLX(F31,F32)                                                 00008200
      RETURN                                                            00008210
      END                                                               00008220
      SUBROUTINE OUTRAV(LU,TITLE,ORDER,X,Y,DR,NR,POT)                   00008230
C                                                                       00008240
```

```
C---- ROUTINE TO OUTPUT TRAVERSE CALCULATION RESULTS                    00008250
C                                                                       00008260
      INTEGER ORDER                                                     00008270
      REAL L1,L2,K21,K23,K21K23                                         00008280
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00008290
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00008300
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00008310
      CHARACTER*80 TITLE                                                00008320
      CHARACTER*32 COLABL                                               00008330
      DIMENSION X(1),Y(1),POT(1)                                        00008340
      PARAMETER (COLABL='       X          Y         POT     ')         00008350
C                                                                       00008360
C---- PRINT HEADER                                                      00008370
      CALL HEADER(LU,TITLE,ORDER)                                       00008380
C                                                                       00008390
C---- PRINT DATA                                                        00008400
      WRITE(LU,1000) DR                                                 00008410
 1000 FORMAT(' DR=',F7.3)                                               00008420
C                                                                       00008430
C---- COMPUTE NUMBER OF COLUMNS AND REMAINDER                           00008440
      ICOL=NR/30                                                        00008450
      IREM=NR-30*ICOL                                                   00008460
      IF(IREM .NE. 0) ICOL=ICOL+1                                       00008470
C                                                                       00008480
C---- PRINT COLUMN LABEL                                                00008490
      WRITE(LU,1010) (COLABL,I=1,ICOL)                                  00008500
 1010 FORMAT(4A32)                                                      00008510
C                                                                       00008520
C---- PRINT DATA                                                        00008530
      I=0                                                               00008540
   10 I=I+1                                                             00008550
      WRITE(LU,1020) (X(I+J),Y(I+J),POT(I+J),J=0,30*(ICOL-1),30)        00008560
 1020 FORMAT(4(1X,F8.2,2X,F8.2,2X,F9.4,2X))                             00008570
C                                                                       00008580
C---- CHECK FOR ALL REMAINDER COLUMNS PRINTED                           00008590
      IF(I .EQ. IREM) ICOL=ICOL-1                                       00008600
C                                                                       00008610
C---- CHECK FOR ALL DATA PRINTED                                        00008620
      IF(I .LT. 30 .AND. I .LT. NR) GO TO 10                            00008630
      RETURN                                                            00008640
      END                                                               00008650
      SUBROUTINE OUTPLOT(LU,TITLE,ORDER,X,Y,DR,NR,POT)                  00008660
C                                                                       00008670
C---- ROUTINE TO OUTPUT DATA FOR TRAVERSE PLOTTING                      00008680
      INTEGER ORDER                                                     00008690
      REAL L1,L2,K21,K23,K21K23                                         00008700
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00008710
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00008720
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00008730
      DIMENSION X(1),Y(1),POT(1)                                        00008740
      CHARACTER*80 TITLE                                                00008750
C                                                                       00008760
C---- WRITE OUTPUT HEADER                                               00008770
      WRITE(LU,1000) TITLE                                              00008780
 1000 FORMAT(20A4)                                                      00008790
      WRITE(LU,1010) SIGMA1,SIGMA2,SIGMA3                               00008800
 1010 FORMAT(' SIGMA1=',F7.3,' SIGMA2=',F7.3,' SIGMA3=',F7.3,11X)       00008810
```

```
      WRITE(LU,1020) C1,C2,C3                                           00008820
 1020 FORMAT(' C1    =',F7.3,' C2    =',F7.3,' C3    =',F7.3,11X)      00008830
      WRITE(LU,1030) D1,T1,L1,F1,S1                                     00008840
 1030 FORMAT(' D1=',F7.3,' T1=',F7.3,' L1=',F7.3,' F1=',F7.3,' S1=',    00008850
     1F8.3)                                                             00008860
      WRITE(LU,1040) D2,T2,L2,F2,S2                                     00008870
 1040 FORMAT(' D2=',F7.3,' T2=',F7.3,' L2=',F7.3,' F2=',F7.3,' S2=',    00008880
     1F8.3)                                                             00008890
      WRITE(LU,1050) WIDTH,OFFSET,TOL,ORDER                             00008900
 1050 FORMAT(' WIDTH=',F7.3,' OFFSET=',F8.3,' TOL=',E10.4,' ORDER=',I2, 00008910
     1X)                                                                00008920
C                                                                       00008930
C---- WRITE NUMBER OF DATA POINTS AND SPACING ALONG TRAVERSE            00008940
      WRITE(LU,1060) NR,DR                                              00008950
 1060 FORMAT(I6,1X,E12.6)                                               00008960
C                                                                       00008970
C---- WRITE DATA                                                        00008980
      WRITE(LU,1070) (X(I),Y(I),POT(I),I=1,NR)                          00008990
 1070 FORMAT(E12.5,1X,E12.5,1X,E12.5)                                   00009000
      RETURN                                                            00009010
      END                                                               00009020
      SUBROUTINE OUTGRID(LU,TITLE,ORDER,X,Y,DX,DY,NX,NY,POT)            00009030
C                                                                       00009040
C---- ROUTINE TO OUTPUT RESULTS FROM GRID CALCULATION                   00009050
C                                                                       00009060
      INTEGER ORDER                                                     00009070
      REAL L1,L2,K21,K23,K21K23                                         00009080
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,    00009090
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,           00009100
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                  00009110
      CHARACTER*80 TITLE                                                00009120
      CHARACTER*24 COLABL                                               00009130
      DIMENSION X(1),Y(1),POT(NX,NY)                                    00009140
      PARAMETER (COLABL='       Y         POT       ')                  00009150
C                                                                       00009160
C---- LOOP OVER XO VALUES                                               00009170
      DO 10 I=1,NX                                                      00009180
C                                                                       00009190
C---- PRINT HEADER                                                      00009200
      CALL HEADER(LU,TITLE,ORDER)                                       00009210
C                                                                       00009220
C---- PRINT DATA                                                        00009230
      WRITE(LU,1000) X(I),DY                                            00009240
 1000 FORMAT(' X=',F8.2,2X,' DY=',F7.3)                                 00009250
C                                                                       00009260
C---- COMPUTE NUMBER OF COLUMNS AND REMAINDER                           00009270
      ICOL=NY/30                                                        00009280
      IREM=NY-30*ICOL                                                   00009290
      IF(IREM .NE. 0) ICOL=ICOL+1                                       00009300
C                                                                       00009310
C---- PRINT COLUMN LABEL                                                00009320
      WRITE(LU,1010) (COLABL,K=1,ICOL)                                  00009330
 1010 FORMAT(4A24)                                                      00009340
C                                                                       00009350
C---- PRINT DATA                                                        00009360
      J=0                                                               00009370
   20 J=J+1                                                             00009380
```

```
      WRITE(LU,1020) (Y(J+K),POT(I,J+K),K=0,30*(ICOL-1),30)              00009390
 1020 FORMAT(4(1X,F8.2,2X,F9.4,4X))                                      00009400
C                                                                        00009410
C---- CHECK FOR ALL REMAINDER COLUMNS PRINTED                            00009420
      IF(J .EQ. IREM) ICOL=ICOL-1                                        00009430
C                                                                        00009440
C---- CHECK FOR ALL DATA PRINTED                                         00009450
      IF(J .LT. 30 .AND. J .LT. NY) GO TO 20                             00009460
   10 CONTINUE                                                           00009470
      RETURN                                                             00009480
      END                                                                00009490
      SUBROUTINE HEADER(LU,TITLE,ORDER)                                  00009500
C                                                                        00009510
C---- ROUTINE TO PRINT OUTPUT HEADER                                     00009520
C                                                                        00009530
      INTEGER ORDER                                                      00009540
      REAL L1,L2,K21,K23,K21K23                                          00009550
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,     00009560
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,            00009570
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                   00009580
      CHARACTER*4 TITLE(20)                                              00009590
C                                                                        00009600
      UW2=-WIDTH2                                                        00009610
      WRITE(LU,1000) TITLE,OFFSET,WIDTH,TOL,ORDER,D1,D2,T1,T2,L1,L2      00009620
 1000 FORMAT(1H1,20A4//' OFFSET=',F8.3,'  WIDTH=',F8.3,'  TOL=',E10.4,   00009630
     1'  ORDER=',I2//                                                    00009640
     221X,'/',14X,'/'/                                                   00009650
     320X,'/',7X,'X',6X,'/'/                                             00009660
     47X,'------------','/','-------','/','------','/',                 00009670
     5'-------------Y'/                                                 00009680
     619X,'I',7X,'I',6X,'I',23X,'SOURCE 1',4X,'SOURCE 2'/                00009690
     719X,'I',7X,'Z',6X,'I',18X,'D=',F10.4,2X,F10.4/                     00009700
     819X,'I',14X,'I',18X,'T=',F10.4,2X,F10.4/                           00009710
     9' REGION=',5X,'1',5X,'S1',7X,'2',5X,'S2',7X,'3',9X,'L=',           00009720
     1F10.4,2X,F10.4)                                                    00009730
      WRITE(LU,1010) SIGMA1,SIGMA2,SIGMA3,F1,F2,C1,C2,C3,S1,S2           00009740
 1010 FORMAT(' SIGMA =',F9.4,'  S1  ',F9.4,'  S2  ',F9.4,                00009750
     16X,'F=',F10.4,2X,F10.4/                                            00009760
     2' C     =',F9.4,'  S1  ',F9.4,'  S2  ',F9.4,                       00009770
     36X,'S=',F10.4,2X,F10.4/                                            00009780
     419X,'I',14X,'I')                                                   00009790
      WRITE(LU,1020) UW2,WIDTH2                                          00009800
 1020 FORMAT(12X,'Y=',F10.4,5X,F10.4/)                                   00009810
      RETURN                                                             00009820
      END                                                                00009830
      SUBROUTINE OUTSTD(LU,TITLE,X,Y,DX,DY,NX,NY,POT)                    00009840
C                                                                        00009850
C---- ROUTINE TO OUTPUT STANDARD FILE FOR CONTOURING                     00009860
C                                                                        00009870
      INTEGER ORDER                                                      00009880
      REAL L1,L2,K21,K23,K21K23                                          00009890
      COMMON /MODEL/ TOL,WIDTH,OFFSET,SIGMA1,SIGMA2,SIGMA3,C1,C2,C3,     00009900
     1F1,F2,S1,S2,D1,T1,L1,D2,T2,L2,K21,K23,K21K23,YO,WIDTH2,            00009910
     2TWOW,HL1,HL2,HT1,HT2,ZMID1,ZMID2                                   00009920
      CHARACTER*80 TITLE                                                 00009930
      CHARACTER*8 PGM                                                    00009940
      DIMENSION X(1),Y(1),POT(NX,NY)                                     00009950
```

```
      PARAMETER (PGM='SPDIKE  ',NZ=1)                                   00009960
C                                                                       00009970
C---- WRITE HEADER RECORD                                               00009980
      WRITE(LU) TITLE(1:56),PGM,NY,NX,NZ,X(1),DX,Y(1),DY                00009990
C                                                                       00010000
C---- LOOP OVER ROWS (X VALUES)                                         00010010
      DO 10 I=1,NX                                                      00010020
   10 WRITE(LU) Y(I),(POT(I,J),J=1,NY)                                  00010030
      RETURN                                                            00010040
      END                                                               00010050
      SUBROUTINE GAUSS(X,W,NORDER)                                      00010060
C                                                                       00010070
C---- ROUTINE TO COMPUTE GAUSSIAN QUADRATURE ABSCISSAS AND WEIGHTS      00010080
C     FOR ORDER NORDER. THE APPROXIMATE ZERO OF THE LEGENDRE            00010090
C     POLYNOMIAL AND THE DERIVATIVE AT THE ZERO IS COMPUTED.            00010100
C     IF THE ABSOLUTE VALUE OF THE POLYNOMIAL AT THE ZERO IS GREATER    00010110
C     THAN 1.0D-15 AND THE ABSOLUTE VALUE OF THE NEWTON'S RULE          00010120
C     CORRECTION TO THE ZERO IS GREATER THAN 1.0D-15, THEN THE          00010130
C     CORRECTION IS APPLIED.  THIS PROCEDURE IS REPEATED FOR A MAXIMUM  00010140
C     OF 10 ITERATIONS OR UNTIL THE ABOVE CRITERIA ARE MET.  THEN THE   00010150
C     WEIGHT IS COMPUTED USING THE FINAL POLYNOMIAL DERIVATIVE VALUES.  00010160
C                                                                       00010170
C     THE SUBROUTINE ONLY COMPUTES NORDER/2 WEIGHTS AND ABSCISSAS.  THE 00010180
C     WEIGHTS ARE THE SAME AT MINUS THE ABSCISSA VALUES.  FOR AN ODD    00010190
C     ORDER THERE IS AN ABSCISSA AT ZERO.                               00010200
C                                                                       00010210
C     WRITTEN BY D. V. FITTERMAN, U.S.G.S., NOVEMBER 1979               00010220
C     MODIFIED 20 JANUARY 1982                                          00010230
      DOUBLE PRECISION X(1),W(1),P(20),DP,PI                            00010240
      PARAMETER (PI=3.1415926539D0)                                     00010250
C                                                                       00010260
C---- LIMIT ORDER TO SIZE OF ARRAY P                                    00010270
      IF(NORDER .GT. 20) NORDER=20                                      00010280
      NLOOP=(NORDER+1)/2                                                00010290
C                                                                       00010300
C---- LOOP OVER ZEROES                                                  00010310
      DO 10 I=1,NLOOP                                                   00010320
C                                                                       00010330
C---- APPROXIMATE ZERO                                                  00010340
      X(I)=(1.0D0-0.125D0/NORDER**2+0.125D0/NORDER**3)*COS((4.0D0*I     00010350
     1-1.0D0)*PI/(4.0D0*NORDER+2.0D0))                                  00010360
      J=1                                                               00010370
C                                                                       00010380
C---- COMPUTE LEGENDRE POLYNOMIAL AT ZERO                               00010390
   20 CALL LEGENDRE(X(I),P,NORDER)                                      00010400
C                                                                       00010410
C---- COMPUTE DERIVATIVE OF LEGENDRE POLYNOMIAL                         00010420
      DP=(-NORDER*X(I)*P(NORDER+1)+NORDER*P(NORDER))/(1.0-X(I)**2)      00010430
C                                                                       00010440
C---- TEST FOR CONVERGENCE                                              00010450
      IF(DABS(P(NORDER+1)) .LE. 1.0D-15 .OR. DABS(P(NORDER+1)/DP) .LE.  00010460
     1 1.0D-15 .OR. J .GE. 10) GO TO 10                                 00010470
C                                                                       00010480
C---- APPLY NEWTON'S RULE CORRECTION TO ZERO                            00010490
      X(I)=X(I)-P(NORDER+1)/DP                                          00010500
      J=J+1                                                             00010510
      GO TO 20                                                          00010520
```

```
C                                                                       00010530
C---- COMPUTE QUADRATURE WEIGHT                                        00010540
   10 W(I)=2.0D0/(1.0D0-X(I)**2)/DP**2                                  00010550
      RETURN                                                            00010560
      END                                                               00010570
      SUBROUTINE LEGENDRE(X,P,NORDER)                                   00010580
C                                                                       00010590
C---- ROUTINE TO EVALUATE THE FIRST NORDER LEGENDRE POLYNOMIALS         00010600
C     FOR ARGUMENT X.                                                   00010610
C                                                                       00010620
C     WRITTEN BY D. V. FITTERMAN, U.S.G.S., NOVEMBER 1979               00010630
C     MODIFIED 13 JANUARY 1982                                          00010640
C                                                                       00010650
      DOUBLE PRECISION X,P(100)                                         00010660
C                                                                       00010670
C---- SET ZEROTH ORDER POLYNOMIAL                                       00010680
      P(1)=1.0D0                                                        00010690
      IF(NORDER .EQ. 0) RETURN                                          00010700
C                                                                       00010710
C---- SET FIRST ORDER POLYNOMIAL                                        00010720
      P(2)=X                                                            00010730
      IF(NORDER .EQ. 1) RETURN                                          00010740
C                                                                       00010750
C---- LOOP OVER ORDERS                                                  00010760
      I=2                                                               00010770
   10 I=I+1                                                             00010780
      P(I)=((2.0D0*(I-2)+1.0D0)*X*P(I-1)-(I-2)*P(I-2))/(I-1)            00010790
      IF(I .LE. NORDER) GO TO 10                                        00010800
      RETURN                                                            00010810
      END                                                               00010820
      SUBROUTINE SPLIN1(M,H,X,Y,A,B,C,IT,D,P,S)                         00010830
C--ONE DIMENSIONAL CUBIC SPLINE COEFFICIENT DETERMINATION.              00010840
C                                                                       00010850
C        BY  W.L.ANDERSON, U.S. GEOLOGICAL SURVEY, DENVER, COLORADO     00010860
C                                                                       00010870
C  PARMS--- M= NUMBER OF DATA POINTS .GT. 2                             00010880
C            H= EQUAL INTERVAL OPTION WHEN H.GT.0. (USE DUMMY X HERE),  00010890
C               UNEQUAL INTERVALS IF H=0. (X REQUIRED STORAGE)          00010900
C            X= INDEP.VAR WHEN H=0. (DIM .GE. M).                       00010910
C            Y= DEPENDENT VARIABLE  (DIM .GE. M).                       00010920
C            A,B,C=COEFF.ARRAYS (EACH DIM .GE. M)                       00010930
C                  RESULTS ARE RETURNED IN 1ST(M-1) ELEMENTS OF A,B,&C.  00010940
C                  ALSO USED AS WORK ARRAYS DURING EXECUTION.            00010950
C            IT= TYPE OF BOUNDARY CONDITION SUPPLIED IN D ARRAY. USE     00010960
C               IT=1 IF 1ST DERIVATIVES GIVEN AT END POINTS, OR         00010970
C               IT=0 IF 2ND DERIVATIVES GIVEN AT END POINTS.            00010980
C            D= BOUNDARY ARRAY (DIM 2) AT POINT 1 AND M RESPECTIVELY.   00010990
C            P,S= WORK ARRAYS (EACH DIM=M).                             00011000
C--ERROR RETURN WITH M=-(ABS(M)) IF ANY PARM OUT OF RANGE.              00011010
C  THE RESULTING CUBIC SPLINE IS OF THE FORM:                           00011020
C     Y=Y(I)+A(I)*(X-X(I))+B(I)*(X-X(I))**2+C(I)*(X-X(I))**3            00011030
C       FOR I=1,2,...,M-1                                               00011040
C                                                                       00011050
C                                                                       00011060
      REAL*4  X(1),Y(1),A(1),B(1),C(1),D(2),P(1),S(1),MUL               00011070
      IF(IT.LT.0.OR.IT.GT.1.OR.H.LT.0..OR.M.LT.3) GO TO 999             00011080
      N=M-1                                                             00011090
```

```
      IF(IT.EQ.0) GO TO 20                                              00011100
C--1ST DERIVATIVE BOUNDARIES GIVEN                                      00011110
      NE=N-1                                                            00011120
      IF(H) 999,11,1                                                    00011130
C--EQUAL SPACING H .GT. 0. AND IT=1                                     00011140
    1 HH=3.0/H                                                          00011150
      DO 2 I=1,NE                                                       00011160
      B(I)=4.0                                                          00011170
      C(I)=1.0                                                          00011180
      A(I)=1.0                                                          00011190
    2 P(I)=HH*(Y(I+2)-Y(I))                                             00011200
      P(1)=P(1)-D(1)                                                    00011210
      P(NE)=P(NE)-D(2)                                                  00011220
C--SOLUTION OF TRIDIAGONAL MATRIX EQ. OF ORDER NE                       00011230
    3 C(1)=C(1)/B(1)                                                    00011240
      P(1)=P(1)/B(1)                                                    00011250
      DO 4 I=2,NE                                                       00011260
      MUL=1.0/(B(I)-A(I)*C(I-1))                                        00011270
      C(I)=MUL*C(I)                                                     00011280
    4 P(I)=MUL*(P(I)-A(I)*P(I-1))                                       00011290
C--OBTAIN SPLINE COEFFICIENTS                                           00011300
      A(NE+IT)=P(NE)                                                    00011310
      I=NE-1                                                            00011320
    5 A(I+IT)=P(I)-C(I)*A(I+IT+1)                                       00011330
      I=I-1                                                             00011340
       IF(I.GE.1) GO TO 5                                               00011350
      IF(IT.EQ.0) GO TO 6                                               00011360
      A(1)=D(1)                                                         00011370
      A(M)=D(2)                                                         00011380
    6 IF(H.EQ.0.) GO TO 14                                              00011390
      HH=1.0/H                                                          00011400
      DO 7 I=1,N                                                        00011410
      MUL=HH*(Y(I+1)-Y(I))                                              00011420
      B(I)=HH*(3.0*MUL-(A(I+1)+2.0*A(I)))                               00011430
    7 C(I)=HH*HH*(-2.0*MUL+A(I+1)+A(I))                                 00011440
      RETURN                                                            00011450
C--UNEQUAL SPACING H=0.. AND IT=1                                       00011460
   11 DO 12 I=1,N                                                       00011470
   12 S(I+1)=X(I+1)-X(I)                                                00011480
      DO 13 I=1,NE                                                      00011490
      B(I)=2.0*(S(I+1)+S(I+2))                                          00011500
      C(I)=S(I+1)                                                       00011510
      A(I)=S(I+2)                                                       00011520
   13 P(I)=3.0*(S(I+1)**2*(Y(I+2)-Y(I+1))+S(I+2)**2*(Y(I+1)-Y(I)))/     00011530
     $ (S(I+1)*S(I+2))                                                  00011540
      P(1)=P(1)-S(3)*D(1)                                               00011550
      P(NE)=P(NE)-S(N)*D(2)                                             00011560
      GO TO 3                                                           00011570
   14 DO 15 I=1,N                                                       00011580
      HH=1.0/S(I+1)                                                     00011590
      MUL=(Y(I+1)-Y(I))*HH**2                                           00011600
      B(I)=3.0*MUL-(A(I+1)+2.0*A(I))*HH                                 00011610
   15 C(I)=-2.0*MUL*HH+(A(I+1)+A(I))*HH**2                              00011620
      RETURN                                                            00011630
C--2ND DERIVATIVE BOUNDARIES GIVEN                                      00011640
   20 NE=N+1                                                            00011650
      IF(H) 999,31,21                                                   00011660
```

```
C--EQUAL SPACING H .GT. 0 AND IT=0                                      00011670
   21 HH=3.0/H                                                          00011680
      DO 22 I=2,N                                                       00011690
      B(I)=4.0                                                          00011700
      C(I)=1.0                                                          00011710
      A(I)=1.0                                                          00011720
   22 P(I)=HH*(Y(I+1)-Y(I-1))                                           00011730
      B(1)=2.0                                                          00011740
      B(NE)=2.0                                                         00011750
      C(1)=1.0                                                          00011760
      C(NE)=1.0                                                         00011770
      A(NE)=1.0                                                         00011780
      P(1)=HH*(Y(2)-Y(1))-0.5*H*D(1)                                    00011790
      P(NE)=HH*(Y(M)-Y(N))+0.5*H*D(2)                                   00011800
      GO TO 3                                                           00011810
C--UNEQUAL SPACING H=0 AND IT=0                                         00011820
   31 DO 32 I=1,N                                                       00011830
   32 S(I+1)=X(I+1)-X(I)                                                00011840
      N1=N-1                                                            00011850
      DO 33 I=1,N1                                                      00011860
      B(I+1)=2.0*(S(I+1)+S(I+2))                                        00011870
      C(I+1)=S(I+1)                                                     00011880
      A(I+1)=S(I+2)                                                     00011890
   33 P(I+1)=3.0*(S(I+1)**2*(Y(I+2)-Y(I+1))+S(I+2)**2*(Y(I+1)-Y(I)))/   00011900
     *     (S(I+1)*S(I+2))                                              00011910
      B(1)=2.0                                                          00011920
      B(NE)=2.0                                                         00011930
      C(1)=1.0                                                          00011940
      C(NE)=1.0                                                         00011950
      A(NE)=1.0                                                         00011960
      P(1)=3.0*(Y(2)-Y(1))/S(2)-0.5*S(2)*D(1)                           00011970
      P(NE)=3.0*(Y(M)-Y(N))/S(M)+0.5*S(M)*D(2)                          00011980
      GO TO 3                                                           00011990
  999 M=-IABS(M)                                                        00012000
      RETURN                                                            00012010
      END                                                               00012020
      SUBROUTINE SPOINT(M,X,Y,A,B,C,XX,YY)                              00012030
C--GIVEN CUBIC SPLINE COEFF'S A,B,C,AND M OBS.DATA ARRAYS X,Y           00012040
C  SPOINT EVALUATES THE PIECEWISE CUBIC SPLINE ORDINATE YY AT THE       00012050
C  ABSCISSA XX, WHERE XX IS IN THE CLOSED INTERVAL (X(1),X(M)).         00012060
C  NOTE: IF COMPUTING OVER EQUAL INTERVALS, USE THE SUBR 'CUBIC'        00012070
C  WHICH REQUIRES ONLY ONE CALL.                                        00012080
C                                                                       00012090
      DIMENSION X(1),Y(1),A(1),B(1),C(1)                                00012100
      IF(XX.LT.X(1).OR.XX.GT.X(M)) GO TO 9                              00012110
      M1=M-1                                                            00012120
      DO 1 I=1,M1                                                       00012130
      J=I                                                               00012140
      IF(XX.LE.X(1+1)) GO TO 2                                          00012150
    1 CONTINUE                                                          00012160
    9 WRITE(6,60) XX,X(1),X(M)                                          00012170
   60 FORMAT('0ERROR IN SPOINT CALL--XX=',E16.8,' NOT IN CLOSED INTERVAL00012180
     * (',E16.8,',',E16.8,')')                                         00012190
      RETURN                                                            00012200
    2 Z=XX-X(J)                                                         00012210
      YY=Y(J)+((C(J)*Z+B(J))*Z+A(J))*Z                                  00012220
      RETURN                                                            00012230
```

```
      END                                                               00012240
      SUBROUTINE HANKEL(BMAX,NB,NREL,TOL,NTOL,NORD,FUN1,IJREL,ZWORK,    00012250
     * ZANS,ARG,NOFUN1,IERR)                                            00012260
      COMPLEX ZANS(NB,NREL),ZWORK(283,NREL),C,CMAX,ZSUM,ZERO,FUN1       00012270
      INTEGER KEY(283),NORD(NREL),IJREL(2,NREL)                         00012280
      DOUBLE PRECISION E,ER,Y1,Y,ABSCIS                                 00012290
      DIMENSION ARG(NB),T(2),TMAX(2)                                    00012300
      DIMENSION WT0(283),WT1(283)                                       00012310
      EQUIVALENCE (C,T(1)),(CMAX,TMAX(1))                               00012320
      DATA ZERO/(0.0,0.0)/                                              00012330
      DATA ABSCIS/0.73588526614797944600/                               00012340
      DATA E,ER/1.22140275816016983400,0.81873075307798185900/          00012350
      DATA                                                              00012360
     * WT0(  1),WT0(  2),WT0(  3),WT0(  4),                            00012370
     * WT0(  5),WT0(  6),WT0(  7),WT0(  8),                            00012380
     * WT0(  9),WT0( 10),WT0( 11),WT0( 12),                            00012390
     * WT0( 13),WT0( 14),WT0( 15),WT0( 16),                            00012400
     * WT0( 17),WT0( 18),WT0( 19),WT0( 20),                            00012410
     * WT0( 21),WT0( 22),WT0( 23),WT0( 24),                            00012420
     * WT0( 25),WT0( 26),WT0( 27),WT0( 28),                            00012430
     * WT0( 29),WT0( 30),WT0( 31),WT0( 32),                            00012440
     * WT0( 33),WT0( 34),WT0( 35),WT0( 36)/                            00012450
     * 2.1969101E-11, 4.1201161E-09,-6.1322980E-09, 7.2479291E-09,     00012460
     *-7.9821627E-09, 8.5778983E-09,-9.1157294E-09, 9.6615250E-09,     00012470
     *-1.0207546E-08, 1.0796633E-08,-1.1393033E-08, 1.2049873E-08,     00012480
     *-1.2708789E-08, 1.3446466E-08,-1.4174300E-08, 1.5005577E-08,     00012490
     *-1.5807160E-08, 1.6747136E-08,-1.7625961E-08, 1.8693427E-08,     00012500
     *-1.9650840E-08, 2.0869789E-08,-2.1903555E-08, 2.3305308E-08,     00012510
     *-2.4407377E-08, 2.6033678E-08,-2.7186773E-08, 2.9094334E-08,     00012520
     *-3.0266804E-08, 3.2534013E-08,-3.3672072E-08, 3.6408936E-08,     00012530
     *-3.7425022E-08, 4.0787921E-08,-4.1543242E-08, 4.5756842E-08/     00012540
      DATA                                                              00012550
     * WT0( 37),WT0( 38),WT0( 39),WT0( 40),                            00012560
     * WT0( 41),WT0( 42),WT0( 43),WT0( 44),                            00012570
     * WT0( 45),WT0( 46),WT0( 47),WT0( 48),                            00012580
     * WT0( 49),WT0( 50),WT0( 51),WT0( 52),                            00012590
     * WT0( 53),WT0( 54),WT0( 55),WT0( 56),                            00012600
     * WT0( 57),WT0( 58),WT0( 59),WT0( 60),                            00012610
     * WT0( 61),WT0( 62),WT0( 63),WT0( 64),                            00012620
     * WT0( 65),WT0( 66),WT0( 67),WT0( 68),                            00012630
     * WT0( 69),WT0( 70),WT0( 71),WT0( 72)/                            00012640
     *-4.6035233E-08, 5.1425075E-08,-5.0893896E-08, 5.7934897E-08,     00012650
     *-5.6086570E-08, 6.5475248E-08,-6.1539913E-08, 7.4301996E-08,     00012660
     *-6.7117043E-08, 8.4767837E-08,-7.2583120E-08, 9.7366568E-08,     00012670
     *-7.7553611E-08, 1.1279873E-07,-8.1416723E-08, 1.3206914E-07,     00012680
     *-8.3217217E-08, 1.5663185E-07,-8.1482581E-08, 1.8860593E-07,     00012690
     *-7.3963141E-08, 2.3109673E-07,-5.7243707E-08, 2.8867452E-07,     00012700
     *-2.6163525E-08, 3.6808773E-07, 2.7049871E-08, 4.7932617E-07,     00012710
     * 1.1407365E-07, 6.3720626E-07, 2.5241961E-07, 8.6373487E-07,     00012720
     * 4.6831433E-07, 1.1916346E-06, 8.0099716E-07, 1.6696015E-06/     00012730
      DATA                                                              00012740
     * WT0( 73),WT0( 74),WT0( 75),WT0( 76),                            00012750
     * WT0( 77),WT0( 78),WT0( 79),WT0( 80),                            00012760
     * WT0( 81),WT0( 82),WT0( 83),WT0( 84),                            00012770
     * WT0( 85),WT0( 86),WT0( 87),WT0( 88),                            00012780
     * WT0( 89),WT0( 90),WT0( 91),WT0( 92),                            00012790
     * WT0( 93),WT0( 94),WT0( 95),WT0( 96),                            00012800
```

```
     * WT0( 97),WT0( 98),WT0( 99),WT0(100),                                  00012810
     * WT0(101),WT0(102),WT0(103),WT0(104),                                  00012820
     * WT0(105),WT0(106),WT0(107),WT0(108)/                                  00012830
     * 1.3091334E-06, 2.3701475E-06, 2.0803829E-06, 3.4012978E-06,           00012840
     * 3.2456774E-06, 4.9240402E-06, 5.0005198E-06, 7.1783540E-06,           00012850
     * 7.6367633E-06, 1.0522038E-05, 1.1590021E-05, 1.5488635E-05,           00012860
     * 1.7510398E-05, 2.2873836E-05, 2.6368006E-05, 3.3864387E-05,           00012870
     * 3.9610390E-05, 5.0230379E-05, 5.9397373E-05, 7.4612122E-05,           00012880
     * 8.8951409E-05, 1.1094809E-04, 1.3308026E-04, 1.6511335E-04,           00012890
     * 1.9895671E-04, 2.4587195E-04, 2.9728181E-04, 3.6629770E-04,           00012900
     * 4.4402013E-04, 5.4589361E-04, 6.6298832E-04, 8.1375348E-04,           00012910
     * 9.8971624E-04, 1.2132772E-03, 1.4772052E-03, 1.8092022E-03/           00012920
      DATA                                                                  00012930
     * WT0(109),WT0(110),WT0(111),WT0(112),                                  00012940
     * WT0(113),WT0(114),WT0(115),WT0(116),                                  00012950
     * WT0(117),WT0(118),WT0(119),WT0(120),                                  00012960
     * WT0(121),WT0(122),WT0(123),WT0(124),                                  00012970
     * WT0(125),WT0(126),WT0(127),WT0(128),                                  00012980
     * WT0(129),WT0(130),WT0(131),WT0(132),                                  00012990
     * WT0(133),WT0(134),WT0(135),WT0(136),                                  00013000
     * WT0(137),WT0(138),WT0(139),WT0(140),                                  00013010
     * WT0(141),WT0(142),WT0(143),WT0(144)/                                  00013020
     * 2.2045122E-03, 2.6980811E-03, 3.2895354E-03, 4.0238764E-03,           00013030
     * 4.9080203E-03, 6.0010999E-03, 7.3216878E-03, 8.9489225E-03,           00013040
     * 1.0919448E-02, 1.3340696E-02, 1.6276399E-02, 1.9873311E-02,           00013050
     * 2.4233627E-02, 2.9555699E-02, 3.5990069E-02, 4.3791529E-02,           00013060
     * 5.3150319E-02, 6.4341372E-02, 7.7506720E-02, 9.2749987E-02,           00013070
     * 1.0980561E-01, 1.2791555E-01, 1.4525830E-01, 1.5820085E-01,           00013080
     * 1.6058576E-01, 1.4196085E-01, 8.9781222E-02,-1.0238278E-02,           00013090
     *-1.5083434E-01,-2.9059573E-01,-2.9105437E-01,-3.7973244E-02,           00013100
     * 3.8273717E-01, 2.2014118E-01,-4.7342635E-01, 1.9331133E-01/           00013110
      DATA                                                                  00013120
     * WT0(145),WT0(146),WT0(147),WT0(148),                                  00013130
     * WT0(149),WT0(150),WT0(151),WT0(152),                                  00013140
     * WT0(153),WT0(154),WT0(155),WT0(156),                                  00013150
     * WT0(157),WT0(158),WT0(159),WT0(160),                                  00013160
     * WT0(161),WT0(162),WT0(163),WT0(164),                                  00013170
     * WT0(165),WT0(166),WT0(167),WT0(168),                                  00013180
     * WT0(169),WT0(170),WT0(171),WT0(172),                                  00013190
     * WT0(173),WT0(174),WT0(175),WT0(176),                                  00013200
     * WT0(177),WT0(178),WT0(179),WT0(180)/                                  00013210
     * 5.3839527E-02,-1.1909845E-01, 9.9317051E-02,-6.6152628E-02,           00013220
     * 4.0703241E-02,-2.4358316E-02, 1.4476533E-02,-8.6198067E-03,           00013230
     * 5.1597053E-03,-3.1074602E-03, 1.8822342E-03,-1.1456545E-03,           00013240
     * 7.0004347E-04,-4.2904226E-04, 2.6354444E-04,-1.6215439E-04,           00013250
     * 9.9891279E-05,-6.1589037E-05, 3.7996921E-05,-2.3452250E-05,           00013260
     * 1.4479572E-05,-8.9417427E-06, 5.5227518E-06,-3.4114252E-06,           00013270
     * 2.1074101E-06,-1.3019229E-06, 8.0433617E-07,-4.9693681E-07,           00013280
     * 3.0702417E-07,-1.8969219E-07, 1.1720069E-07,-7.2412496E-08,           00013290
     * 4.4740283E-08,-2.7643004E-08, 1.7079403E-08,-1.0552634E-08/           00013300
      DATA                                                                  00013310
     * WT0(181),WT0(182),WT0(183),WT0(184),                                  00013320
     * WT0(185),WT0(186),WT0(187),WT0(188),                                  00013330
     * WT0(189),WT0(190),WT0(191),WT0(192),                                  00013340
     * WT0(193),WT0(194),WT0(195),WT0(196),                                  00013350
     * WT0(197),WT0(198),WT0(199),WT0(200),                                  00013360
     * WT0(201),WT0(202),WT0(203),WT0(204),                                  00013370
```

```
     * WT0(205),WT0(206),WT0(207),WT0(208),                              00013380
     * WT0(209),WT0(210),WT0(211),WT0(212),                              00013390
     * WT0(213),WT0(214),WT0(215),WT0(216)/                              00013400
     * 6.5200311E-09,-4.0284597E-09, 2.4890232E-09,-1.5378695E-09,       00013410
     * 9.5019040E-10,-5.8708696E-10, 3.6273937E-10,-2.2412348E-10,       00013420
     * 1.3847792E-10,-8.5560821E-11, 5.2865474E-11,-3.2664392E-11,       00013430
     * 2.0182948E-11,-1.2470979E-11, 7.7057678E-12,-4.7611713E-12,       00013440
     * 2.9415274E-12,-1.8170081E-12, 1.1221034E-12,-6.9271067E-13,       00013450
     * 4.2739744E-13,-2.6344388E-13, 1.6197105E-13,-9.9147443E-14,       00013460
     * 6.0487998E-14,-3.6973097E-14, 2.2817964E-14,-1.4315547E-14,       00013470
     * 9.1574735E-15,-5.9567236E-15, 3.9209969E-15,-2.5911739E-15,       00013480
     * 1.6406939E-15,-8.8248590E-16, 3.0195409E-16, 2.2622634E-17/       00013490
      DATA                                                              00013500
     * WT0(217),WT0(218),WT0(219),WT0(220),                              00013510
     * WT0(221),WT0(222),WT0(223),WT0(224),                              00013520
     * WT0(225),WT0(226),WT0(227),WT0(228),                              00013530
     * WT0(229),WT0(230),WT0(231),WT0(232),                              00013540
     * WT0(233),WT0(234),WT0(235),WT0(236),                              00013550
     * WT0(237),WT0(238),WT0(239),WT0(240),                              00013560
     * WT0(241),WT0(242),WT0(243),WT0(244),                              00013570
     * WT0(245),WT0(246),WT0(247),WT0(248),                              00013580
     * WT0(249),WT0(250),WT0(251),WT0(252)/                              00013590
     *-8.0942556E-17,-3.7172363E-17, 1.9299542E-16,-3.3388160E-16,       00013600
     * 4.6174116E-16,-5.8627358E-16, 7.2227767E-16,-8.7972941E-16,       00013610
     * 1.0211793E-15,-1.0940039E-15, 1.0789555E-15,-9.7089714E-16,       00013620
     * 7.4110927E-16,-4.1700094E-16, 8.5977184E-17, 1.3396469E-16,       00013630
     *-1.7838410E-16, 4.8975421E-17, 1.9398153E-16,-5.0046989E-16,       00013640
     * 8.3280985E-16,-1.1544640E-15, 1.4401527E-15,-1.6637066E-15,       00013650
     * 1.7777129E-15,-1.7322187E-15, 1.5247247E-15,-1.1771155E-15,       00013660
     * 6.9747910E-16,-1.2088956E-16,-4.8382957E-16, 1.0408292E-15,       00013670
     *-1.5220450E-15, 1.9541597E-15,-2.4107448E-15, 2.9241438E-15/       00013680
      DATA                                                              00013690
     * WT0(253),WT0(254),WT0(255),WT0(256),                              00013700
     * WT0(257),WT0(258),WT0(259),WT0(260),                              00013710
     * WT0(261),WT0(262),WT0(263),WT0(264),                              00013720
     * WT0(265),WT0(266),WT0(267),WT0(268),                              00013730
     * WT0(269),WT0(270),WT0(271),WT0(272),                              00013740
     * WT0(273),WT0(274),WT0(275),WT0(276),                              00013750
     * WT0(277),WT0(278),WT0(279),WT0(280),                              00013760
     * WT0(281),WT0(282),WT0(283)/                                       00013770
     *-3.5176475E-15, 4.2276125E-15,-5.0977851E-15, 6.1428456E-15,       00013780
     *-7.3949962E-15, 8.8597601E-15,-1.0515959E-14, 1.2264584E-14,       00013790
     *-1.3949870E-14, 1.5332490E-14,-1.6146782E-14, 1.6084121E-14,       00013800
     *-1.4962523E-14, 1.2794804E-14,-9.9286701E-15, 6.8825809E-15,       00013810
     *-4.0056107E-15, 1.5965079E-15,-7.2732961E-18,-4.0433218E-16,       00013820
     *-6.5679655E-16, 3.3011866E-15,-7.3545910E-15, 1.2394851E-14,       00013830
     *-1.7947697E-14, 2.3774303E-14,-3.0279168E-14, 3.9252831E-14,       00013840
     *-5.5510504E-14, 9.0505371E-14,-1.7064873E-13/                      00013850
      DATA                                                              00013860
     * WT1(  1),WT1(  2),WT1(  3),WT1(  4),                              00013870
     * WT1(  5),WT1(  6),WT1(  7),WT1(  8),                              00013880
     * WT1(  9),WT1( 10),WT1( 11),WT1( 12),                              00013890
     * WT1( 13),WT1( 14),WT1( 15),WT1( 16),                              00013900
     * WT1( 17),WT1( 18),WT1( 19),WT1( 20),                              00013910
     * WT1( 21),WT1( 22),WT1( 23),WT1( 24),                              00013920
     * WT1( 25),WT1( 26),WT1( 27),WT1( 28),                              00013930
     * WT1( 29),WT1( 30),WT1( 31),WT1( 32),                              00013940
```

```
     * WT1( 33),WT1( 34),WT1( 35),WT1( 36)/                                  00013950
     *-4.2129715E-16, 5.3667031E-15,-7.1183962E-15, 8.9478500E-15,          00013960
     *-1.0767891E-14, 1.2362265E-14,-1.3371129E-14, 1.3284178E-14,          00013970
     *-1.1714302E-14, 8.4134738E-15,-3.7726725E-15,-1.4263879E-15,          00013980
     * 6.1279163E-15,-9.1102765E-15, 9.9696405E-15,-9.3649955E-15,          00013990
     * 8.6009018E-15,-8.9749846E-15, 1.1153987E-14,-1.4914821E-14,          00014000
     * 1.9314024E-14,-2.3172388E-14, 2.5605477E-14,-2.6217555E-14,          00014010
     * 2.5057768E-14,-2.2485539E-14, 1.9022752E-14,-1.5198084E-14,          00014020
     * 1.1422464E-14,-7.9323958E-15, 4.8421406E-15,-2.1875032E-15,          00014030
     *-3.2177842E-17, 1.8637565E-15,-3.3683643E-15, 4.6132219E-15/          00014040
      DATA                                                                 00014050
     * WT1( 37),WT1( 38),WT1( 39),WT1( 40),                                 00014060
     * WT1( 41),WT1( 42),WT1( 43),WT1( 44),                                 00014070
     * WT1( 45),WT1( 46),WT1( 47),WT1( 48),                                 00014080
     * WT1( 49),WT1( 50),WT1( 51),WT1( 52),                                 00014090
     * WT1( 53),WT1( 54),WT1( 55),WT1( 56),                                 00014100
     * WT1( 57),WT1( 58),WT1( 59),WT1( 60),                                 00014110
     * WT1( 61),WT1( 62),WT1( 63),WT1( 64),                                 00014120
     * WT1( 65),WT1( 66),WT1( 67),WT1( 68),                                 00014130
     * WT1( 69),WT1( 70),WT1( 71),WT1( 72)/                                 00014140
     *-5.6209538E-15, 6.4192841E-15,-6.8959928E-15, 6.9895792E-15,          00014150
     *-6.5355935E-15, 5.6125163E-15,-4.1453931E-15, 2.6358827E-15,          00014160
     *-9.5104370E-16, 1.4600474E-16, 5.6166519E-16, 8.2899246E-17,          00014170
     * 5.0032100E-16, 4.3752205E-16, 2.1052293E-15,-9.5451973E-16,          00014180
     * 6.4004437E-15,-2.1926177E-15, 1.1651003E-14, 5.8415433E-16,          00014190
     * 1.8044664E-14, 1.0755745E-14, 3.0159022E-14, 3.3506138E-14,          00014200
     * 5.8709354E-14, 8.1475200E-14, 1.2530006E-13, 1.8519112E-13,          00014210
     * 2.7641786E-13, 4.1330823E-13, 6.1506209E-13, 9.1921659E-13,          00014220
     * 1.3698462E-12, 2.0447427E-12, 3.0494477E-12, 4.5501001E-12/          00014230
      DATA                                                                 00014240
     * WT1( 73),WT1( 74),WT1( 75),WT1( 76),                                 00014250
     * WT1( 77),WT1( 78),WT1( 79),WT1( 80),                                 00014260
     * WT1( 81),WT1( 82),WT1( 83),WT1( 84),                                 00014270
     * WT1( 85),WT1( 86),WT1( 87),WT1( 88),                                 00014280
     * WT1( 89),WT1( 90),WT1( 91),WT1( 92),                                 00014290
     * WT1( 93),WT1( 94),WT1( 95),WT1( 96),                                 00014300
     * WT1( 97),WT1( 98),WT1( 99),WT1(100),                                 00014310
     * WT1(101),WT1(102),WT1(103),WT1(104),                                 00014320
     * WT1(105),WT1(106),WT1(107),WT1(108)/                                 00014330
     * 6.7870250E-12, 1.0126237E-11, 1.5104976E-11, 2.2536053E-11,          00014340
     * 3.3617368E-11, 5.0153839E-11, 7.4818173E-11, 1.1161804E-10,          00014350
     * 1.6651222E-10, 2.4840923E-10, 3.7058109E-10, 5.5284353E-10,          00014360
     * 8.2474468E-10, 1.2303750E-09, 1.8355034E-09, 2.7382502E-09,          00014370
     * 4.0849867E-09, 6.0940898E-09, 9.0913020E-09, 1.3562651E-08,          00014380
     * 2.0233058E-08, 3.0184244E-08, 4.5029477E-08, 6.7176304E-08,          00014390
     * 1.0021488E-07, 1.4950371E-07, 2.2303208E-07, 3.3272689E-07,          00014400
     * 4.9636623E-07, 7.4049804E-07, 1.1046805E-06, 1.6480103E-06,          00014410
     * 2.4585014E-06, 3.6677163E-06, 5.4714550E-06, 8.1626422E-06/          00014420
      DATA                                                                 00014430
     * WT1(109),WT1(110),WT1(111),WT1(112),                                 00014440
     * WT1(113),WT1(114),WT1(115),WT1(116),                                 00014450
     * WT1(117),WT1(118),WT1(119),WT1(120),                                 00014460
     * WT1(121),WT1(122),WT1(123),WT1(124),                                 00014470
     * WT1(125),WT1(126),WT1(127),WT1(128),                                 00014480
     * WT1(129),WT1(130),WT1(131),WT1(132),                                 00014490
     * WT1(133),WT1(134),WT1(135),WT1(136),                                 00014500
     * WT1(137),WT1(138),WT1(139),WT1(140),                                 00014510
```

```
     * WT1(141),WT1(142),WT1(143),WT1(144)/                                  00014520
     * 1.2176782E-05, 1.8166179E-05, 2.7099223E-05, 4.0428804E-05,           00014530
     * 6.0307294E-05, 8.9971508E-05, 1.3420195E-04, 2.0021123E-04,           00014540
     * 2.9860417E-04, 4.4545291E-04, 6.6423156E-04, 9.9073275E-04,           00014550
     * 1.4767050E-03, 2.2016806E-03, 3.2788147E-03, 4.8837292E-03,           00014560
     * 7.2596811E-03, 1.0788355E-02, 1.5973323E-02, 2.3612041E-02,           00014570
     * 3.4655327E-02, 5.0608141E-02, 7.2827752E-02, 1.0337889E-01,           00014580
     * 1.4207357E-01, 1.8821315E-01, 2.2996815E-01, 2.5088500E-01,           00014590
     * 2.0334626E-01, 6.0665451E-02,-2.0275683E-01,-3.5772336E-01,           00014600
     *-1.8280529E-01, 4.7014634E-01, 7.2991233E-03,-3.0614594E-01/           00014610
      DATA                                                                  00014620
     * WT1(145),WT1(146),WT1(147),WT1(148),                                  00014630
     * WT1(149),WT1(150),WT1(151),WT1(152),                                  00014640
     * WT1(153),WT1(154),WT1(155),WT1(156),                                  00014650
     * WT1(157),WT1(158),WT1(159),WT1(160),                                  00014660
     * WT1(161),WT1(162),WT1(163),WT1(164),                                  00014670
     * WT1(165),WT1(166),WT1(167),WT1(168),                                  00014680
     * WT1(169),WT1(170),WT1(171),WT1(172),                                  00014690
     * WT1(173),WT1(174),WT1(175),WT1(176),                                  00014700
     * WT1(177),WT1(178),WT1(179),WT1(180)/                                  00014710
     * 2.4781735E-01,-1.1149185E-01, 2.5985386E-02, 1.0850279E-02,           00014720
     *-2.2830217E-02, 2.4644647E-02,-2.2895284E-02, 2.0197032E-02,           00014730
     *-1.7488968E-02, 1.5057670E-02,-1.2953923E-02, 1.1153254E-02,           00014740
     *-9.6138436E-03, 8.2952090E-03,-7.1628361E-03, 6.1882910E-03,           00014750
     *-5.3482055E-03, 4.6232056E-03,-3.9970542E-03, 3.4560118E-03,           00014760
     *-2.9883670E-03, 2.5840801E-03,-2.2345428E-03, 1.9323046E-03,           00014770
     *-1.6709583E-03, 1.4449655E-03,-1.2495408E-03, 1.0805480E-03,           00014780
     *-9.3441130E-04, 8.0803899E-04,-6.9875784E-04, 6.0425624E-04,           00014790
     *-5.2253532E-04, 4.5186652E-04,-3.9075515E-04, 3.3790861E-04/           00014800
      DATA                                                                  00014810
     * WT1(181),WT1(182),WT1(183),WT1(184),                                  00014820
     * WT1(185),WT1(186),WT1(187),WT1(188),                                  00014830
     * WT1(189),WT1(190),WT1(191),WT1(192),                                  00014840
     * WT1(193),WT1(194),WT1(195),WT1(196),                                  00014850
     * WT1(197),WT1(198),WT1(199),WT1(200),                                  00014860
     * WT1(201),WT1(202),WT1(203),WT1(204),                                  00014870
     * WT1(205),WT1(206),WT1(207),WT1(208),                                  00014880
     * WT1(209),WT1(210),WT1(211),WT1(212),                                  00014890
     * WT1(213),WT1(214),WT1(215),WT1(216)/                                  00014900
     *-2.9220916E-04, 2.5269019E-04,-2.1851585E-04, 1.8896332E-04,           00014910
     *-1.6340753E-04, 1.4130796E-04,-1.2219719E-04, 1.0567099E-04,           00014920
     *-9.1379828E-05, 7.9021432E-05,-6.8334412E-05, 5.9092726E-05,           00014930
     *-5.1100905E-05, 4.4189914E-05,-3.8213580E-05, 3.3045496E-05,           00014940
     *-2.8576356E-05, 2.4711631E-05,-2.1369580E-05, 1.8479514E-05,           00014950
     *-1.5980307E-05, 1.3819097E-05,-1.1950174E-05, 1.0334008E-05,           00014960
     *-8.9364160E-06, 7.7278366E-06,-6.6827083E-06, 5.7789251E-06,           00014970
     *-4.9973715E-06, 4.3215167E-06,-3.7370660E-06, 3.2316575E-06,           00014980
     *-2.7946015E-06, 2.4166539E-06,-2.0898207E-06, 1.8071890E-06/           00014990
      DATA                                                                  00015000
     * WT1(217),WT1(218),WT1(219),WT1(220),                                  00015010
     * WT1(221),WT1(222),WT1(223),WT1(224),                                  00015020
     * WT1(225),WT1(226),WT1(227),WT1(228),                                  00015030
     * WT1(229),WT1(230),WT1(231),WT1(232),                                  00015040
     * WT1(233),WT1(234),WT1(235),WT1(236),                                  00015050
     * WT1(237),WT1(238),WT1(239),WT1(240),                                  00015060
     * WT1(241),WT1(242),WT1(243),WT1(244),                                  00015070
     * WT1(245),WT1(246),WT1(247),WT1(248),                                  00015080
```

```
     * WT1(249),WT1(250),WT1(251),WT1(252)/                                  00015090
     *-1.5627811E-06, 1.3514274E-06,-1.1686576E-06, 1.0106059E-06,          00015100
     *-8.7392952E-07, 7.5573750E-07,-6.5353002E-07, 5.6514528E-07,          00015110
     *-4.8871388E-07, 4.2261921E-07,-3.6546333E-07, 3.1603732E-07,          00015120
     *-2.7329579E-07, 2.3633470E-07,-2.0437231E-07, 1.7673258E-07,          00015130
     *-1.5283091E-07, 1.3216174E-07,-1.1428792E-07, 9.8831386E-08,          00015140
     *-8.5465227E-08, 7.3906734E-08,-6.3911437E-08, 5.5267923E-08,          00015150
     *-4.7793376E-08, 4.1329702E-08,-3.5740189E-08, 3.0906612E-08,          00015160
     *-2.6726739E-08, 2.3112160E-08,-1.9986424E-08, 1.7283419E-08,          00015170
     *-1.4945974E-08, 1.2924650E-08,-1.1176694E-08, 9.6651347E-09/          00015180
      DATA                                                                  00015190
     * WT1(253),WT1(254),WT1(255),WT1(256),                                 00015200
     * WT1(257),WT1(258),WT1(259),WT1(260),                                 00015210
     * WT1(261),WT1(262),WT1(263),WT1(264),                                 00015220
     * WT1(265),WT1(266),WT1(267),WT1(268),                                 00015230
     * WT1(269),WT1(270),WT1(271),WT1(272),                                 00015240
     * WT1(273),WT1(274),WT1(275),WT1(276),                                 00015250
     * WT1(277),WT1(278),WT1(279),WT1(280),                                 00015260
     * WT1(281),WT1(282),WT1(283)/                                          00015270
     *-8.3580023E-09, 7.2276490E-09,-6.2501673E-09, 5.4048822E-09,          00015280
     *-4.6739154E-09, 4.0418061E-09,-3.4951847E-09, 3.0224895E-09,          00015290
     *-2.6137226E-09, 2.2602382E-09,-1.9545596E-09, 1.6902214E-09,          00015300
     *-1.4616324E-09, 1.2639577E-09,-1.0930164E-09, 9.4519327E-10,          00015310
     *-8.1736202E-10, 7.0681930E-10,-6.1122713E-10, 5.2856342E-10,          00015320
     *-4.5707937E-10, 3.9526267E-10,-3.4180569E-10, 2.9557785E-10,          00015330
     *-2.5560176E-10, 2.2103233E-10,-1.9113891E-10, 1.6528994E-10,          00015340
     *-1.4294012E-10, 1.2361991E-10,-8.2740936E-11/                         00015350
      NOFUN1=0                                                              00015360
      IF(NB.LT.1.OR.NREL.LT.1.OR.BMAX.LE.0.0) GO TO 9999                    00015370
      Y=DBLE(BMAX)*ER**(NB-1)                                               00015380
      IF(Y.LE.0.0D0) GO TO 9999                                             00015390
      IERR=0                                                                00015400
      DO 10 I=1,283                                                         00015410
10      KEY(I)=0                                                            00015420
      NB1=NB+1                                                              00015430
      LAG=-1                                                                00015440
      Y1=ABSCIS/DBLE(BMAX)                                                  00015450
      DO 1010 ILAG=1,NB                                                     00015460
        LAG=LAG+1                                                           00015470
        ISTORE=NB1-ILAG                                                     00015480
        IF(LAG.GT.0) Y1=Y1*E                                                00015490
        ARG(ISTORE)=ABSCIS/Y1                                               00015500
        DO 1000 JREL=1,NREL                                                 00015510
          NONE=0                                                            00015520
          ITOL=NTOL                                                         00015530
          ZSUM=ZERO                                                         00015540
          CMAX=ZERO                                                         00015550
          Y=Y1                                                              00015560
          ASSIGN 20 TO M                                                    00015570
          I=131                                                             00015580
          Y=Y*E                                                             00015590
          GO TO 100                                                         00015600
20        TMAX(1)=AMAX1(ABS(T(1)),TMAX(1))                                  00015610
          TMAX(2)=AMAX1(ABS(T(2)),TMAX(2))                                  00015620
          I=I+1                                                             00015630
          Y=Y*E                                                             00015640
          IF(I.LE.149) GO TO 100                                            00015650
```

```
      IF(TMAX(1).EQ.0.0.AND.TMAX(2).EQ.0.0) NONE=1                      00015660
      CMAX=TOL*CMAX                                                     00015670
      ASSIGN 30 TO M                                                    00015680
      GO TO 100                                                         00015690
30    IF(ABS(T(1)).LE.TMAX(1).AND.ABS(T(2)).LE.TMAX(2)) GO TO 50        00015700
      ITOL=NTOL                                                         00015710
40    I=I+1                                                             00015720
      Y=Y*E                                                             00015730
      IF(I.LE.283) GO TO 100                                            00015740
50    ITOL=ITOL-1                                                       00015750
      IF(ITOL.GT.0.AND.I.LT.283) GO TO 40                               00015760
      ITOL=NTOL                                                         00015770
      Y=Y1                                                              00015780
      ASSIGN 60 TO M                                                    00015790
      I=130                                                             00015800
      GO TO 100                                                         00015810
60    IF(ABS(T(1)).LE.TMAX(1).AND.ABS(T(2)).LE.TMAX(2).AND.             00015820
     *  NONE.EQ.0) GO TO 80                                            00015830
      ITOL=NTOL                                                         00015840
70    I=I-1                                                             00015850
      Y=Y*ER                                                            00015860
      IF(I.GT.0) GO TO 100                                              00015870
80    ITOL=ITOL-1                                                       00015880
      IF(ITOL.GT.0.AND.I.GT.1) GO TO 70                                 00015890
      ZANS(ISTORE,JREL)=ZSUM/ARG(ISTORE)                                00015900
      GO TO 1000                                                        00015910
100   LOOK=1+LAG                                                        00015920
      IQ=LOOK/284                                                       00015930
      IR=MOD(LOOK,284)                                                  00015940
      IF(IR.EQ.0) IR=1                                                  00015950
      IROLL=IQ*283                                                      00015960
      IF(KEY(IR).LE.IROLL) GO TO 150                                    00015970
110   IF(NORD(JREL)) 130,120,130                                        00015980
120   C=ZWORK(IR,JREL)*WT0(I)                                           00015990
      GO TO 140                                                         00016000
130   C=ZWORK(IR,JREL)*WT1(I)                                           00016010
140   ZSUM=ZSUM+C                                                       00016020
      GO TO M,(20,30,60)                                                00016030
150   KEY(IR)=IROLL+IR                                                  00016040
      G=Y                                                               00016050
      ZWORK(IR,1)=FUN1(G)                                               00016060
      NOFUN1=NOFUN1+1                                                   00016070
      IF(NREL.EQ.1) GO TO 110                                           00016080
      DO 160 J=2,NREL                                                   00016090
160     ZWORK(IR,J)=CMPLX(G**IJREL(1,J),0.0)*ZWORK(IR,1)**IJREL(2,J)    00016100
      GO TO 110                                                         00016110
1000  CONTINUE                                                          00016120
1010 CONTINUE                                                           00016130
     RETURN                                                             00016140
9999 IERR=1                                                             00016150
     RETURN                                                             00016160
     END                                                                00016170
```