

APPENDIX B

COMPUTER SUBPROGRAM PACKAGE

User information and Fortran listings of the IDBVIP/IDSFFT subprogram package are given in this appendix. This package implements the method of bivariate interpolation and smooth surface fitting for irregularly distributed data points, described in section 2 of this report. It is written in ANSI Standard Fortran (ANSI, 1966).

The package consists of a block-data subprogram and the following six subroutines; i.e., IDBVIP, IDGEOM, IDLCTN, IDPDRV, IDPTIP, and IDSFFT. Two subroutines, IDBVIP and IDSFFT, are the master subroutines of the package, and each interfaces with the user. The remaining four subroutines are common supporting subroutines called by IDBVIP and IDSFFT. The IDBVIP subroutine performs bivariate interpolation for irregularly distributed data points; it estimates the z values at the specified points in the x-y plane. The IDSFFT subroutine performs smooth surface fitting; it estimates the z values at the specified rectangular grid points in the x-y plane and generates a doubly-dimensioned array containing these estimated values.

The package includes three common blocks; i.e., IDGM, IDNN, and IDPI. Including these common areas, the package occupies approximately 3200 locations on the CDC-6600 computer.

When the user wishes to call either IDBVIP or IDSFFT subroutine repeatedly with identical data as parts of input data in two consecutive calls, he can save computation times considerably by specifying an appropriate mode of computation. (This mode is specified with the MD parameter in the call statements to be described later.)

User information on IDBVIP and that of IDSFFT will follow. This information is followed by Fortran listings of the seven subprograms --- six subroutines listed in alphabetical order, followed by the block-data subprogram.

The IDBVIP Subroutine.

This subroutine performs bivariate interpolation when the projections of the data points in the x-y plane are irregularly distributed in the plane.

This subroutine is called by the following statement:

```
CALL IDBVIP( MD, NDP, XD, YD, ZD, WK, NIP, XI, YI, ZI )
```

In this call statement, the input parameters are

MD = mode of computation (must be 1, 2, or 3),

= 1 for new XD-YD,

= 2 for old XD-YD, new XI-YI,

= 3 for old XD-YD, old XI-YI,

NDP = number of data points (must be 4 or greater),

XD = array of dimension NDP containing the x coordinates
of the data points,

YD = array of dimension NDP containing the y coordinates
of the data points,

ZD = array of dimension NDP containing the z coordinates
of the data points,

WK = array of dimension $(2 * NDP + NNP + 5) * NDP + NIP$
to be used internally as a work area,

NIP = number of points to be interpolated at (must be 1 or
greater),

XI = array of dimension NIP containing the x coordinates
of the points to be interpolated at,

YI = array of dimension NIP containing the y coordinates
of the points to be interpolated at,

where NNP is the number of additional data points used for estimating
partial derivatives at each data point. The output parameter is

ZI = array of dimension NIP, where the z coordinates
of the interpolated points will be stored.

The LUN constant in the data initialization statement is the logical
unit number of the standard output unit and is, therefore, system de-
pendent. The user must enter an appropriate number into LUN before
compiling this subroutine.

The value of NNP must be given through the IDNN common block.
NNP must be 2 or greater, but smaller than NDP. In the subprogram
package listed below, it is set to 4. The user can change it by declaring

COMMON/IDNN/NNP

in his calling program and by assigning a number of his choice to NNP
with an arithmetic assignment statement before the call to IDBVIP.

The call to this subroutine with MD = 2 must be preceded by an-
other call to this subroutine with the same NDP value and with the same
contents of the XD and YD arrays. The call with MD = 3 must be pre-
ceded by another call with the same NDP and NIP values and with the
same contents of the XD, YD, XI, and YI arrays. Between the call
with MD = 2 or 3 and its preceding call, the WK array should not be
disturbed.

Table B-1 (p. 32) shows the approximate computation times re-
quired on the CDC-6600 computer.

The IDSFFT Subroutine.

This subroutine performs smooth surface fitting when the projections of the data points in the x-y plane are irregularly distributed in the plane.

This subroutine is called by the following statement:

```
CALL IDSFFT ( MD, NDP, XD, YD, ZD, WK, NXI, NYI, XI, YI, ZI )
```

In this call statement, the input parameters are

MD = mode of computation (must be 1, 2, or 3),

= 1 for new XD-YD,

= 2 for old XD-YD, new XI-YI,

= 3 for old XD-YD, old XI-YI,

NDP = number of data points (must be 4 or greater),

XD = array of dimension NDP containing the x coordinates
of the data points,

YD = array of dimension NDP containing the y coordinates
of the data points,

ZD = array of dimension NDP containing the z coordinates
of the data points,

WK = array of dimension $(2 * NDP + NNP + 5) * NDP + NXI * NYI$
to be used internally as a work area,

NXI = number of output grid points in the x coordinate
(must be 1 or greater),

NYI = number of output grid points in the y coordinate
(must be 1 or greater),

XI = array of dimension NXI containing the x coordinates
of the output grid points,

YI = array of dimension NYI containing the y coordinates
of the output grid points,

where NNP is the number of additional data points used for estimating partial derivatives at each data point. The output parameter is

ZI = doubly-dimensioned array of dimension (NXI, NYI),
where the interpolated z values at the output grid
points will be stored.

The LUN constant in the data initialization statement is the logical unit number of the standard output unit and is, therefore, system dependent. The user must enter an appropriate number into LUN before compiling this subroutine.

The value of NNP must be given through the IDNN common block. NNP must be 2 or greater, but smaller than NDP. In the subprogram package listed below, it is set to 4. The user can change it by declaring

COMMON/IDNN/NNP

in his calling program and by assigning a number of his choice to NNP with an arithmetic assignment statement before the call to this subroutine.

The call to this subroutine with MD = 2 must be preceded by another call to this subroutine with the same NDP value and with the same contents of the XD and YD arrays. The call with MD = 3 must be preceded by another call with the same NDP, NXI, and NYI values and with the same contents of the XD, YD, XI, and YI arrays. Between the call with MD = 2 or 3 and its preceding call, the WK array should not be disturbed.

Table B-2 (p. 32) shows the approximate computation times required on the CDC-6600 computer.

Table B-1. Approximate computation times required for the IDBVIP subroutine on the CDC-6600 computer.

NDP	NIP	Time (seconds)		
		MD = 1	MD = 2	MD = 3
20	10	0.40	0.03	0.02
	100	0.50	0.12	0.06
	1000	1.4	1.0	0.35
30	10	1.3	0.04	0.03
	100	1.5	0.16	0.07
	1000	2.7	1.4	0.50
50	10	6.6	0.05	0.04
	100	6.8	0.24	0.10
	1000	8.8	2.2	0.70

Table B-2. Approximate computation times required for the IDSFFT subroutine on the CDC-6600 computer.

NDP	NXI * NYI	Time (seconds)		
		MD = 1	MD = 2	MD = 3
20	11 * 11	0.50	0.12	0.07
	33 * 33	1.1	0.70	0.40
	101 * 101	5.8	5.4	3.4
30	11 * 11	1.5	0.16	0.08
	33 * 33	2.1	0.85	0.41
	101 * 101	7.3	6.0	3.5
50	11 * 11	6.8	0.22	0.11
	33 * 33	7.8	1.2	0.50
	101 * 101	14.0	7.3	3.7

```

SUBROUTINE IDBVIP(MD,NDP,XD,YD,ZD,WK,NIP,XI,YI,ZI)           IBI 001
C THIS SUBROUTINE PERFORMS BIVARIATE INTERPOLATION WHEN THE PRO-   IBI 002
C JECTIONS OF THE DATA POINTS IN THE X-Y PLANE ARE IRREGULARLY   IBI 003
C DISTRIBUTED IN THE PLANE.                                         IBI 004
C THE INPUT PARAMETERS ARE                                         IBI 005
C   MD = MODE OF COMPUTATION (MUST BE 1, 2, OR 3),                 IBI 006
C     = 1 FOR NEW XD-YD,                                           IBI 007
C     = 2 FOR OLD XD-YD, NEW XI-YI,                                IBI 008
C     = 3 FOR OLD XD-YD, OLD XI-YI,                               IBI 009
C   NDP = NUMBER OF DATA POINTS (MUST BE 4 OR GREATER),          IBI 010
C   XD = ARRAY OF DIMENSION NDP STORING THE X COORDINATES      IBI 011
C     OF THE DATA POINTS,                                         IBI 012
C   YD = ARRAY OF DIMENSION NDP STORING THE Y COORDINATES      IBI 013
C     OF THE DATA POINTS,                                         IBI 014
C   ZD = ARRAY OF DIMENSION NDP STORING THE Z COORDINATES      IBI 015
C     OF THE DATA POINTS,                                         IBI 016
C   WK = ARRAY OF DIMENSION (2*NDP+NNP+5)*NDP+NIP               IBI 017
C     TO BE USED AS A WORK AREA,                                 IBI 018
C   NIP = NUMBER OF INTERPOLATED POINTS                         IBI 019
C     (MUST BE 1 OR GREATER),                                    IBI 020
C   XI = ARRAY OF DIMENSION NIP STORING THE X COORDINATES      IBI 021
C     OF THE INTERPOLATED POINTS,                                IBI 022
C   YI = ARRAY OF DIMENSION NIP STORING THE Y COORDINATES      IBI 023
C     OF THE INTERPOLATED POINTS,                                IBI 024
C WHERE NNP IS THE NUMBER OF ADDITIONAL DATA POINTS USED FOR    IBI 025
C ESTIMATING PARTIAL DERIVATIVES AT EACH DATA POINT. THE VALUE    IBI 026
C OF NNP MUST BE GIVEN THROUGH THE IDNN COMMON. NNP MUST BE 2      IBI 027
C OR GREATER, BUT SMALLER THAN NDP.                                IBI 028
C THE OUTPUT PARAMETER IS                                         IBI 029
C   ZI = ARRAY OF DIMENSION NIP, WHERE THE Z COORDINATES        IBI 030
C     OF THE INTERPOLATED POINTS ARE TO BE DISPLAYED.            IBI 031
C THE LUN CONSTANT IN THE DATA INITIALIZATION STATEMENT IS THE    IBI 032
C LOGICAL UNIT NUMBER OF THE STANDARD OUTPUT UNIT AND IS,        IBI 033
C THEREFORE, SYSTEM DEPENDENT.                                     IBI 034
C DECLARATION STATEMENTS                                         IBI 035
DIMENSION XD(10),YD(10),ZD(10),WK(1000),                      IBI 036
1       XI(10),YI(10),ZI(10)                                      IBI 037
COMMON/IDNN/NNP                                                 IBI 038
COMMON/IDGM/NDPC,NNPC,NT,NL                                     IBI 039
COMMON/IDPI/NCF,ICF                                           IBI 040
EQUIVALENCE (FNDPO,NDPO),(FNDPPV,NDPPV),                      IBI 041
1       (FNNPO,NNPO),(FNNPPV,NNPPV),                            IBI 042
2       (FNIP0,NIPO),(FNIPPV,NIPPV),                            IBI 043
3       (FNT,NT),(FNL,NL)                                       IBI 044
DATA LUN/6/                                                       IBI 045
C SETTING OF SOME INPUT PARAMETERS TO LOCAL VARIABLES. (ALL MD) IBI 046
10 MDO=MD                                                       IBI 047
  NDPO=NDP                                                     IBI 048
  NDPC=NDPO                                                    IBI 049
  NIPO=NIP                                                    IBI 050
  NNPO=NNP                                                    IBI 051
  NNPC=NNPO                                                   IBI 052
C ERROR CHECK. (ALL MD)                                         IBI 053
20 IF(MDO.LT.1.OR.MDO.GT.3) GO TO 90                           IBI 054
  IF(NDPO.LT.4) GO TO 90                                       IBI 055
  IF(NIPO.LT.1) GO TO 90                                       IBI 056
  IF(NNPO.LT.2.OR.NNPO.GE.NDPO) GO TO 90                      IBI 057
  IF(MDO.NE.1) GO TO 22                                         IBI 058
21 WK(1)=FNDPO                                                 IBI 059
  WK(2)=FNNPO                                                 IBI 060
  GO TO 24                                                   IBI 061
22 FNDPPV=WK(1)                                              IBI 062
  FNNPPV=WK(2)                                              IBI 063
  IF(NDPO.NE.NDPPV) GO TO 90                                  IBI 064
  IF(NNPO.NE.NNPPV) GO TO 90                                  IBI 065

```

```

      IF(MD0.NE.3)          GO TO 24           IBI 066
23 FNIPPV=WK(3)          IBI 067
      IF(NIPO.NE.NIPPV)    GO TO 90           IBI 068
      GO TO 30           IBI 069
24 WK(3)=FNPO           IBI 070
C ALLOCATION OF STORAGE AREAS IN THE WK ARRAY. (ALL MD) IBI 071
30 NDNDM1=NDPO*(NDPO-1) IBI 072
      IW IPT=7           IBI 073
      IW IPL=IW IPT+NDNDM1 IBI 074
      IW IPN=IW IPL+NDNDM1 IBI 075
      IW PD =IW IPN+NDPO*NNPO IBI 075
      IW IT =IW PD +NDPO*5 IBI 077
C DIVIDES THE X-Y PLANE INTO A NUMBER OF TRIANGLES AND IBI 078
C DETERMINES NNP POINTS NEAREST EACH DATA POINT, (MD=1) IBI 079
40 IF(MD.GT.1)          GO TO 42           IBI 080
41 CALL IDGEOM(XD,YD,WK(IW IPT),WK(IW IPL),WK(IW IPN)) IBI 081
      WK(5)=FNT           IBI 082
      WK(6)=FNL           IBI 083
      GO TO 50           IBI 084
42 FNT=WK(5)           IBI 085
      FNL=WK(6)           IBI 086
C ESTIMATES PARTIAL DERIVATIVES AT ALL DATA POINTS. (ALL MD) IBI 087
50 CALL IDPDRV(XD,YD,ZD,WK(IW IPN),WK(IW PD)) IBI 088
C LOCATES ALL INTERPOLATED POINTS. (MD=1,2) IBI 089
60 IF(MD0.EQ.3)          GO TO 70           IBI 090
      JW IT=IW IT-1        IBI 091
      DO 61   IIP=1,NIPO   IBI 092
      JW IT=JW IT+1        IBI 093
      CALL IDLCTN(XD,YD,WK(IW IPT),WK(IW IPL), IBI 094
      1                  XI(IIP),YI(IIP),WK(JW IT)) IBI 095
61 CONTINUE             IBI 096
C INTERPOLATION OF THE ZI VALUES. (ALL MD) IBI 097
70 NCF=0                IBI 098
      ICF=0               IBI 099
      JW IT=IW IT-1        IBI 100
      DO 71   IIP=1,NIPO   IBI 101
      JW IT=JW IT+1        IBI 102
      CALL IDPTIP(XD,YD,ZD,WK(IW IPT),WK(IW IPL),WK(IW PD), IBI 103
      1                  WK(JW IT),XI(IIP),YI(IIP),ZI(IIP)) IBI 104
71 CONTINUE             IBI 105
C NORMAL EXIT           IBI 106
80 RETURN               IBI 107
C ERROR EXIT            IBI 108
90 WRITE (LUN,2090) MD0,NDPO,NIPO,NNPO IBI 109
      RETURN              IBI 110
C FORMAT STATEMENT FOR ERROR MESSAGE IBI 111
2090 FORMAT(1X/41H *** IMPROPER INPUT PARAMETER VALUE(S).// IBI 112
      1 7H MD =,I4,10X,5HNDP =,I6,10X,5HNIP =,I6, IBI 113
      2 10X,5HNPN =,I6/ IBI 114
      3 35H ERROR DETECTED IN ROUTINE IDBVIP/) IBI 115
      END                 IBI 116

```

```

      SUBROUTINE IDGEOM(XD,YD,IPT,IPL,IPN)           IGM 001
C THIS SUBROUTINE DIVIDES THE X-Y PLANE INTO A NUMBER OF IGM 002
C TRIANGULAR AREAS ACCORDING TO GIVEN DATA POINTS IN THE PLANE. IGM 003
C DETERMINES LINE SEGMENTS THAT FORM THE BORDER OF DATA AREA. IGM 004
C DETERMINES THE TRIANGLE NUMBERS CORRESPONDING TO THE BORDER IGM 005
C LINE SEGMENTS, AND SELECTS SEVERAL DATA POINTS THAT ARE IGM 006
C NEAREST TO EACH OF THE DATA POINTS. IGM 007
C AT COMPLETION, POINT NUMBERS OF THE VERTEXES OF EACH TRIANGLE IGM 008
C ARE LISTED COUNTER-CLOCKWISE. POINT NUMBERS OF THE END POINTS IGM 009
C OF EACH BORDER LINE SEGMENT ARE LISTED COUNTER-CLOCKWISE, IGM 010
C LISTING ORDER OF THE LINE SEGMENTS BEING COUNTER-CLOCKWISE. IGM 011

```

```

C THE INPUT PARAMETERS ARE IGM 012
C XD,YD = ARRAYS STORING THE X AND Y COORDINATES, RESP., IGM 013
C OF DATA POINTS. IGM 014
C THE OUTPUT PARAMETERS ARE IGM 015
C IPT = ARRAY OF DIMENSION 3*NT, WHERE THE POINT NUMBERS IGM 016
C OF THE VERTEXES OF THE (IT)TH TRIANGLE ARE TO BE IGM 017
C DISPLAYED AS THE (3*IT-2)ND, (3*IT-1)ST, AND IGM 018
C (3*IT)TH ELEMENTS, IT=1,2,...,NT, IGM 019
C IPL = ARRAY OF DIMENSION 3*NL, WHERE THE POINT NUMBERS IGM 020
C OF THE END POINTS OF THE (IL)TH BORDER LINE IGM 021
C SEGMENT AND ITS RESPECTIVE TRIANGLE NUMBER ARE IGM 022
C TO BE DISPLAYED AS THE (3*IL-2)ND, (3*IL-1)ST, IGM 023
C AND (3*IL)TH ELEMENTS, IL=1,2,..., NL, IGM 024
C IPN = ARRAY OF DIMENSION NDP*NNP, WHERE THE POINT IGM 025
C NUMBERS OF NNP DATA POINTS NEAREST TO EACH OF IGM 026
C THE DATA POINTS ARE TO BE DISPLAYED, IGM 027
C WHERE NDP IS THE TOTAL NUMBER OF DATA POINTS, NNP IS THE IGM 028
C NUMBER OF DATA POINTS NEAREST TO EACH DATA POINT, NL IS IGM 029
C THE NUMBER OF BORDER LINE SEGMENTS, AND NT IS THE NUMBER IGM 030
C OF TRIANGLES. NDP AND NNP ARE GIVEN TO THIS SUBROUTINE IGM 031
C THROUGH THE IDGM COMMON. NL AND NT ARE CALCULATED BY THIS IGM 032
C SUBROUTINE AND ARE LEFT IN THE IDGM COMMON AT COMPLETION. IGM 033
C DECLARATION STATEMENTS IGM 034
      DIMENSION XD(10),YD(10),IPT(100),IPL(100),IPN(50) IGM 035
      COMMON/IDGM/NDP,NNP,NT,NL IGM 036
      EQUIVALENCE (DSQ1,IDSQ1),(DSQ2,IDSQ2),(DSQM,IDSQM) IGM 037
C PRELIMINARY PROCESSING IGM 038
10  NDPO=NDP IGM 039
    NDPM1=NDP-1 IGM 040
    NNPO=NNP IGM 041
    NNPM1=NNP-1 IGM 042
C DETERMINES THE NEAREST NNP POINTS. IGM 043
20  DO 29  IP1=1,NDPO IGM 044
    X1=XD(IP1) IGM 045
    Y1=YD(IP1) IGM 046
    J1MX=IP1*NNP IGM 047
    J1MN=J1MX-NNPM1 IGM 048
    DO 28  J1=J1MN,J1MX IGM 049
        J2MX=J1-1 IGM 050
        IDMN=0 IGM 051
        DO 27  IP2=1,NDPO IGM 052
            IF(IP2.EQ.IP1)          GO TO 27 IGM 053
            IF(J1.GT.J1MN)          GO TO 22 IGM 054
21      DSQ1=(XD(IP2)-X1)**2+(YD(IP2)-Y1)**2 IGM 055
            IPT(IP2)=IDSQ1 IGM 056
            GO TO 23 IGM 057
22      IDSQ1=IPT(IP2) IGM 058
23      IF(IDMN.EQ.0)          GO TO 24 IGM 059
            IF(DSQ1.GE.DSQMN)      GO TO 27 IGM 060
24      IF(J1MN.GT.J2MX)      GO TO 26 IGM 061
            DO 25  J2=J1MN,J2MX IGM 062
                IF(IP2.EQ.IPN(J2))  GO TO 27 IGM 063
25      CONTINUE IGM 064
26      DSQMN=DSQ1 IGM 065
            IDMN=IP2 IGM 066
27      CONTINUE IGM 067
            IPN(J1)=IDMN IGM 068
28      CONTINUE IGM 069
29  CONTINUE IGM 070
C LISTS ALL THE POSSIBLE LINE SEGMENTS IN THE IPL ARRAY, IGM 071
C CALCULATES THE SQUARES OF THE LINE SEGMENT LENGTHS, AND STORE IGM 072
C THEM IN THE IPT ARRAY. IGM 073
30  IL=0 IGM 074
    DO 32  IP1=1,NDPM1 IGM 075
        X1=XD(IP1) IGM 076

```

```

Y1=YD(IP1)                                IGM 077
IP1P1=IP1+1                               IGM 078
DO 31 IP2=IP1P1,NDPO                      IGM 079
    IL=IL+1                                 IGM 080
    ILT2=IL+IL                               IGM 081
    IPL(ILT2-1)=IP1                         IGM 082
    IPL(ILT2) =IP2                         IGM 083
    DSQ1=(XD(IP2)-X1)**2+(YD(IP2)-Y1)**2   IGM 084
    IPT(IL)=IDSQ1                           IGM 085
31  CONTINUE                                IGM 086
32 CONTINUE                                IGM 087
    NLO=IL                                 IGM 088
C SORTS THE IPL AND IPT ARRAYS IN ASCENDING ORDER OF THE LINE   IGM 089
C SEGMENT LENGTH (DISTANCE).                                     IGM 090
35 NLM1=NLO-1                               IGM 091
    DO 37 IL1=1,NLM1                         IGM 092
        IDSQ1=IPT(IL1)                        IGM 093
        ILM=IL1                               IGM 094
        DSQM=DSQ1                            IGM 095
        IL2MN=IL1+1                          IGM 096
        DO 36 IL2=IL2MN,NLO                  IGM 097
            IDSQ2=IPT(IL2)                    IGM 098
            IF(DSQ2.GE.DSQM)      Go To 36   IGM 099
            ILM=IL2                            IGM 100
            DSQM=DSQ2                           IGM 101
36  CONTINUE                                IGM 102
        IPT(ILM)=IDSQ1                        IGM 103
        IPT(IL1)=DSQM                         IGM 104
        IL1T2=IL1+IL1                         IGM 105
        ILMT2=ILM+ILM                         IGM 106
        ITS=IPL(IL1T2-1)                      IGM 107
        IPL(IL1T2-1)=IPL(ILMT2-1)           IGM 108
        IPL(ILMT2-1)=ITS                     IGM 109
        ITS=IPL(IL1T2)                       IGM 110
        IPL(IL1T2)=IPL(ILMT2)              IGM 111
        IPL(ILMT2)=ITS                     IGM 112
37  CONTINUE                                IGM 113
C ELIMINATES LINE SEGMENTS THAT CROSS OR LIE OVER SHORTER ONE.   IGM 114
40 IL0=1                                   IGM 115
    DO 46 IL1=2,NLO                         IGM 116
        IL1T2=IL1+IL1                         IGM 117
        IP1=IPL(IL1T2-1)                      IGM 118
        IP2=IPL(IL1T2)                        IGM 119
        X1=XD(IP1)                           IGM 120
        X2=XD(IP2)                           IGM 121
        Y1=YD(IP1)                           IGM 122
        Y2=YD(IP2)                           IGM 123
        DX21=X2-X1                           IGM 124
        DY21=Y2-Y1                           IGM 125
        DO 45 IL2=1,IL0                         IGM 126
            IL2T2=IL2+IL2                      IGM 127
            IP3=IPL(IL2T2-1)                   IGM 128
            IP4=IPL(IL2T2)                     IGM 129
            X3=XD(IP3)                         IGM 130
            X4=XD(IP4)                         IGM 131
            Y3=YD(IP3)                         IGM 132
            Y4=YD(IP4)                         IGM 133
            DX43=X4-X3                         IGM 134
            DX42=X4-X2                         IGM 135
            DX41=X4-X1                         IGM 136
            DX32=X3-X2                         IGM 137
            DX31=X3-X1                         IGM 138
            DY43=Y4-Y3                         IGM 139
            DY42=Y4-Y2                         IGM 140
            DY41=Y4-Y1                         IGM 141

```

```

    DY32=Y3-Y2          IGM 142
    DY31=Y3-Y1          IGM 143
    IF(IP3.NE.IP1) GO TO 41   IGM 144
    IF(DY41*DX21-DX41*DY21.NE.0.0) GO TO 45   IGM 145
    41 IF(DX41*DX21+DY41*DY21)      45,45,46   IGM 146
    IF(IP4.NE.IP1) GO TO 42   IGM 147
    IF(DY31*DX21-DX31*DY21.NE.0.0) GO TO 45   IGM 148
    42 IF(DX31*DX21+DY31*DY21)      45,45,46   IGM 149
    IF(IP3.NE.IP2) GO TO 43   IGM 150
    IF(DY42*DX21-DX42*DY21.NE.0.0) GO TO 45   IGM 151
    43 IF(DX42*DX21+DY42*DY21)      46,45,45   IGM 152
    IF(IP4.NE.IP2) GO TO 44   IGM 153
    IF(DY32*DX21-DX32*DY21.NE.0.0) GO TO 45   IGM 154
    44 IF(DX32*DX21+DY32*DY21)      46,45,45   IGM 155
    1 IF((DY31*DX21-DX31*DY21)*(DY41*DX21-DX41*DY21).GE.0.0)   IGM 156
    1 GO TO 45           IGM 157
    1 IF((DY31*DX43-DX31*DY43)*(DY32*DX43-DX32*DY43).LT.0.0)   IGM 158
    1 GO TO 46           IGM 159
    45 CONTINUE          IGM 160
    IL0=IL0+1            IGM 161
    IL0T2=IL0+IL0        IGM 162
    IPL(IL0T2-1)=IP1     IGM 163
    IPL(IL0T2) =IP2     IGM 164
    46 CONTINUE          IGM 165
    NLO=IL0              IGM 166
C RE-SORTS THE IPL ARRAY IN ASCENDING ORDER OF ITS ELEMENTS.
    50 NLT2=NLO+NLO        IGM 167
    NLM1T2=NLT2-2        IGM 168
    DO 54 IL1T2=2,NLM1T2,2   IGM 169
    ILMT2=IL1T2            IGM 170
    IPM1=IPL(ILMT2-1)      IGM 171
    IPM2=IPL(ILMT2)        IGM 172
    IL2T2M=IL1T2+2        IGM 173
    DO 53 IL2T2=IL2T2M,NLT2,2   IGM 174
    IP21=IPL(IL2T2-1)      IGM 175
    IP22=IPL(IL2T2)        IGM 176
    51 IF(IPM1-IP21)      53,51,52   IGM 177
    IF(IPM2-IP22)      53,53,52   IGM 178
    52 ILMT2=IL2T2          IGM 179
    IPM1=IP21             IGM 180
    IPM2=IP22             IGM 181
    53 CONTINUE          IGM 182
    IPL(ILMT2-1)=IPL(IL1T2-1)   IGM 183
    IPL(ILMT2) =IPL(IL1T2)     IGM 184
    IPL(IL1T2-1)=IPM1       IGM 185
    IPL(IL1T2) =IPM2       IGM 186
    54 CONTINUE          IGM 187
C DETERMINES TRIANGLES.
    60 IT=0              IGM 188
    NLM1=NLO-1            IGM 189
    NLM2=NLO-2            IGM 190
    DO 67 IL1=1,NLM2        IGM 191
    IL1T2=IL1+IL1          IGM 192
    IP1=IPL(IL1T2-1)      IGM 193
    IP2=IPL(IL1T2)        IGM 194
    IL1P1=IL1+1            IGM 195
    DO 66 IL2=IL1P1,NLM1   IGM 196
    IL2T2=IL2+IL2          IGM 197
    IF(IPL(IL2T2-1).NE.IP1) GO TO 67   IGM 198
    IP3=IPL(IL2T2)        IGM 199
    IL2P1=IL2+1            IGM 200
    DO 62 IL3=IL2P1,NLO   IGM 201
    IL3T2=IL3+IL3          IGM 202
    IF(IPL(IL3T2-1)-IP2)  62,61,66   IGM 203
    61 IF(IPL(IL3T2) -IP3) 62,63,66   IGM 204
                                IGM 205
                                IGM 206

```

```

62    CONTINUE                                IGM 207
      GO TO 66                                IGM 208
63    IPT1=IP1                                IGM 209
      IPT2=IP2                                IGM 210
      IPT3=IP3                                IGM 211
      IF((YD(IPT3)-YD(IPT1))*(XD(IPT2)-XD(IPT1))-  

1       (XD(IPT3)-XD(IPT1))*(YD(IPT2)-YD(IPT1)).GE.0.0)   IGM 212
      GO TO 64                                IGM 213
2
      ITS=IPT2                                IGM 214
      IPT2=IPT3                                IGM 215
      IPT3=ITS                                 IGM 216
      IGM 217
64    X1=XD(IPT1)                            IGM 218
      X2=XD(IPT2)                            IGM 219
      X3=XD(IPT3)                            IGM 220
      Y1=YD(IPT1)                            IGM 221
      Y2=YD(IPT2)                            IGM 222
      Y3=YD(IPT3)                            IGM 223
      DX32=X3-X2                            IGM 224
      DX21=X2-X1                            IGM 225
      DX13=X1-X3                            IGM 226
      DY32=Y3-Y2                            IGM 227
      DY21=Y2-Y1                            IGM 228
      DY13=Y1-Y3                            IGM 229
      DO 65 IPO=1,NDPO                      IGM 230
      IF(IPO.EQ.IPT1.OR.IPO.EQ.IPT2.OR.IPO.EQ.IPT3)  IGM 231
      GO TO 65                                IGM 232
1
      X0=XD(IPO)                            IGM 233
      Y0=YD(IPO)                            IGM 234
      IF((Y0-Y1)*DX21-(X0-X1)*DY21.LT.0.0)  GO TO 65  IGM 235
      IF((Y0-Y2)*DX32-(X0-X2)*DY32.LT.0.0)  GO TO 65  IGM 236
      IF((Y0-Y3)*DX13-(X0-X3)*DY13.GE.0.0)  GO TO 66  IGM 237
65    CONTINUE                                IGM 238
      IT=IT+1                                IGM 239
      ITT3=IT*3                                IGM 240
      IPT(ITT3-2)=IPT1                        IGM 241
      IPT(ITT3-1)=IPT2                        IGM 242
      IPT(ITT3) =IPT3                        IGM 243
66    CONTINUE                                IGM 244
67    CONTINUE                                IGM 245
      NTO=IT                                IGM 246
      NT=NTO                                 IGM 247
C SELECTS AND SORTS LINE SEGMENTS THAT FORM THE BORDER.
70    IL0=0                                IGM 248
      DO 75 IL1=1,NL0                      IGM 249
      IL1T2=IL1+IL1                        IGM 250
      IP1=IPL(IL1T2-1)                      IGM 251
      IP2=IPL(IL1T2)                        IGM 252
      X1=XD(IP1)                            IGM 253
      Y1=YD(IP1)                            IGM 254
      X2=XD(IP2)                            IGM 255
      Y2=YD(IP2)                            IGM 256
      DX21=X2-X1                            IGM 257
      DY21=Y2-Y1                            IGM 258
      DO 71 IPO=1,NDPO                      IGM 259
      IF(IPO.EQ.IP1.OR.IPO.EQ.IP2)  GO TO 71  IGM 260
      S=(YD(IPO)-Y1)*DX21-(XD(IPO)-X1)*DY21  IGM 261
      IF(S.NE.0.0)  GO TO 72                IGM 262
71    CONTINUE                                IGM 263
72    IPOMN=IPn+1                          IGM 264
      DO 73 IPO=IPoMN,NDPO                  IGM 265
      IF(IPO.EQ.IP1.OR.IPO.EQ.IP2)  GO TO 73  IGM 266
      IF(((YD(IPO)-Y1)*DX21-(XD(IPO)-X1)*DY21)*S.LT.0.0)  IGM 267
      GO TO 75                                IGM 268
1
73    CONTINUE                                IGM 269
      IGM 270

```

```

ILO=ILO+1                                IGM 271
ILOT2=IL0+ILO                            IGM 272
IF(S.LT.0.0) GO TO 74                    IGM 273
IPL(ILOT2-1)=IP1                         IGM 274
IPL(ILOT2) =IP2                          IGM 275
GO TO 75                                  IGM 276
74   IPL(ILOT2-1)=IP2                   IGM 277
     IPL(ILOT2) =IP1                     IGM 278
75 CONTINUE                               IGM 279
NLO=ILO                                 IGM 280
NLMI=NLO-1                             IGM 281
DO 79  IL1=2,NLMI                      IGM 282
    IL1T2=IL1+IL1
    IP2=IPL(IL1T2-2)
    IF(IPL(IL1T2-1).EQ.IP2)      GO TO 79
    IL1P1=IL1+1
    DO 77  IL2=IL1P1,NLO
        IL2T2=IL2+IL2
        IF(IPL(IL2T2-1).EQ.IP2)      GO TO 78
77   CONTINUE                               IGM 284
78   IP1=IPL(IL1T2-1)                   IGM 285
     IP2=IPL(IL1T2)
     IPL(IL1T2-1)=IPL(IL2T2-1)
     IPL(IL1T2) =IPL(IL2T2)
     IPL(IL2T2-1)=IP1
     IPL(IL2T2) =IP2
79 CONTINUE                               IGM 286
NL=NLO                                 IGM 287
C FINDS OUT TRIANGLES CORRESPONDING TO THE BORDER LINE
C SEGMENTS.
80  NLP1=NLo+1                           IGM 288
    DO 83  ILR=1,NLO
        IL=NLP1-ILR
        ILT2=IL+IL
        ILT3=ILT2+IL
        IPL1=IPL(ILT2-1)
        IPL2=IPL(ILT2)
        DO 81  IT=1,NT0
            ITT3=IT*3
            IPT1=IPT(ITT3-2)
            IPT2=IPT(ITT3-1)
            IPT3=IPT(ITT3)
            IF(IPL1.NE.IPT1.AND.IPL1.NE.IPT2.AND.IPL1.NE.IPT3)
1             GO TO 81
            IF(IPL2.EQ.IPT1.OR.IPL2.EQ.IPT2.OR.IPL2.EQ.IPT3)
1             GO TO 82
81   CONTINU
82   IPL(ILT3-2)=IPL1                   IGM 300
     IPL(ILT3-1)=IPL2                   IGM 301
     IPL(ILT3) =IT                      IGM 302
83 CONTINUE                               IGM 303
RETURN
END

```

```

SUBROUTINE IDLCTN(XD,YD,IPT,IPL,XII,YII,ITI)          ILC 001
C THIS SUBROUTINE LOCATES A POINT, I.E., DETERMINES WHAT    ILC 002
C TRIANGLE A GIVEN POINT (XII,YII) BELONGS TO. WHEN THE GIVEN    ILC 003
C POINT DOES NOT LIE INSIDE THE DATA AREA, THIS SUBROUTINE    ILC 004
C DETERMINES THE BORDER LINE SEGMENT IN THE AREA ABOVE WHICH THE    ILC 005
C POINT LIES, OR TWO BORDER LINE SEGMENTS BETWEEN TWO AREAS    ILC 006
C ABOVE WHICH THE POINT LIES.                                     ILC 007
C THE INPUT PARAMETERS ARE                                     ILC 008

```

```

C XD,YD = ARRAYS STORING THE X AND Y COORDINATES, RESP.,      ILC 009
C OF DATA POINTS,                                              ILC 010
C IPT = ARRAY STORING THE POINT NUMBERS OF THE VERTICES        ILC 011
C OF THE TRIANGLES,                                             ILC 012
C IPL = ARRAY STORING THE POINT NUMBERS OF THE END             ILC 013
C POINTS OF THE BORDER LINE SEGMENTS AND THEIR                 ILC 014
C RESPECTIVE TRIANGLE NUMBERS,                                 ILC 015
C XII,YII = X AND Y COORDINATES, RESP., OF                   ILC 016
C INTERPOLATED POINT.                                         ILC 017
C THE OUTPUT PARAMETER IS                                     ILC 018
C ITI = TRIANGLE NUMBER, WHEN THE POINT IS INSIDE THE       ILC 019
C DATA AREA, OR                                               ILC 020
C TWO BORDER LINE SEGMENT NUMBERS, IL1 AND IL2,              ILC 021
C CODED TO IL1*(NT+NL)+IL2, WHEN THE POINT IS                ILC 022
C OUTSIDE THE DATA AREA, WHERE NT IS THE NUMBER OF          ILC 023
C TRIANGLES AND NL, THAT OF BORDER LINE SEGMENTS.           ILC 024
C DECLARATION STATEMENTS                                     ILC 025
DIMENSION XD(10),YD(10),IPT(100),IPL(100)                  ILC 026
COMMON/IDGM/NDP,NNP,NT,NL                                     ILC 027
DATA NTPV/0/,NLPV/0/                                         ILC 028
C PRELIMINARY PROCESSING                                     ILC 029
10 NTO=NT                                                 ILC 030
NLO=NL                                                 ILC 031
NTL=NTO+NLO                                              ILC 032
XO=XII                                                 ILC 033
YO=YII                                                 ILC 034
C CHECK IF IN THE SAME TRIANGLE AS PREVIOUS               ILC 035
20 IF(NTO.NE.NTPV)      GO TO 35                           ILC 036
IF(NLO.NE.NLPV)      GO TO 35                           ILC 037
ITO=ITIPV                                              ILC 038
IF(IT0.GT.NTO)      GO TO 25                           ILC 039
ITOT3=IT0*3                                              ILC 040
IP1=IPT(ITOT3-2)                                         ILC 041
IP2=IPT(ITOT3-1)                                         ILC 042
IP3=IPT(ITOT3)                                           ILC 043
X1=XD(IP1)                                              ILC 044
X2=XD(IP2)                                              ILC 045
X3=XD(IP3)                                              ILC 046
Y1=YD(IP1)                                              ILC 047
Y2=YD(IP2)                                              ILC 048
Y3=YD(IP3)                                              ILC 049
IF((YO-Y1)*(X2-X1)-(X0-X1)*(Y2-Y1))      50,21,21   ILC 050
21 IF((YO-Y2)*(X3-X2)-(X0-X2)*(Y3-Y2))      50,22,22   ILC 051
22 IF((YO-Y3)*(X1-X3)-(X0-X3)*(Y1-Y3))      50,80,80   ILC 052
C CHECK IF ON THE SAME BORDER LINE SEGMENT                ILC 053
25 IL1=IT0/NTL                                            ILC 054
IL2=IT0-IL1*NTL                                         ILC 055
IL1T3=IL1*3                                              ILC 056
IP1=IPL(IL1T3-2)                                         ILC 057
IP2=IPL(IL1T3-1)                                         ILC 058
X1=XD(IP1)                                              ILC 059
X2=XD(IP2)                                              ILC 060
Y1=YD(IP1)                                              ILC 061
Y2=YD(IP2)                                              ILC 062
DX02=X0-X2                                              ILC 063
DY02=Y0-Y2                                              ILC 064
DX21=X2-X1                                              ILC 065
DY21=Y2-Y1                                              ILC 066
CS0221=DX02*DX21+DY02*DY21                            ILC 067
IF(IL2.NE.IL1)      GO TO 30                           ILC 068
IF(CS0221)      26,26,50                                ILC 069
26 DX01=X0-X1                                           ILC 070
DY01=Y0-Y1                                           ILC 071
IF(DY01*DX21-DX01*DY21)  27,27,50                    ILC 072
27 IF(DX01*DX21+DY01*DY21)  50,80,80                  ILC 073

```

```

C CHECK IF BETWEEN THE SAME TWO BORDER LINE SEGMENTS           ILC 074
30 IF(CS0221)      50,31,31                                ILC 075
31 IL2T3=IL2*3                                         ILC 076
   IP3=IPL(IL2T3-1)                                     ILC 077
   X3=XD(IP3)                                         ILC 078
   Y3=YD(IP3)                                         ILC 079
   DX32=X3-X2                                         ILC 080
   DY32=Y3-Y2                                         ILC 081
   IF(DXn2*DX32+DY02*DY32)  80,80,50                   ILC 082
C WHEN CALLED WITH A NEW SET OF NT AND NL                   ILC 083
35 NTPV=NTO                                         ILC 084
   NLPV=NLA                                         ILC 085
   ITIPV=0                                         ILC 086
C LOCATION INSIDE THE DATA AREA                         ILC 087
50 ITOT3=0
   DO 69  ITO=1,NTO
   ITOT3=ITOT3+3
   IF(IT0.EQ.ITIPV)  GO TO 69
   IP1=IPT(ITOT3-2)
   IP2=IPT(ITOT3-1)
   IP3=IPT(ITOT3)
   X1=XD(IP1)
   X2=XD(IP2)
   X3=XD(IP3)
   IF(X0-X1)      53,55,51
51  IF(X0-X2)      55,55,52
52  IF(X0-X3)      55,55,69
53  IF(X0-X2)      54,55,55
54  IF(X0-X3)      69,55,55
55  Y1=YD(IP1)
   Y2=YD(IP2)
   Y3=YD(IP3)
   IF(Y0-Y1)      58,60,56
56  IF(Y0-Y2)      60,60,57
57  IF(Y0-Y3)      60,60,69
58  IF(Y0-Y2)      59,60,60
59  IF(Y0-Y3)      69,60,60
60  IF((Y0-Y1)*(X2-X1)-(X0-X1)*(Y2-Y1))  69,61,61
61  IF((Y0-Y2)*(X3-X2)-(X0-X2)*(Y3-Y2))  69,62,62
62  IF((Y0-Y3)*(X1-X3)-(X0-X3)*(Y1-Y3))  69,80,80
69 CONTINUE
C LOCATION OUTSIDE THE DATA AREA                      ILC 114
70 NLOT3=NLO*3
   IP1=IPL(NLOT3-2)
   IP2=IPL(NLOT3-1)
   X1=XD(IP1)
   Y1=YD(IP1)
   X2=XD(IP2)
   Y2=YD(IP2)
   DX02=X0-X2
   DY02=Y0-Y2
   DX21=X2-X1
   DY21=Y2-Y1
   CS0221=DXn2*DX21+DY02*DY21
   DO 74  IL0=1,NLO
   X1=X2
   Y1=Y2
   DX01=DX02
   DY01=DY02
   IP2=IPL(3*IL0-1)
   X2=XD(IP2)
   Y2=YD(IP2)
   DX02=X0-X2
   DY02=Y0-Y2
   ILC 115
   ILC 116
   ILC 117
   ILC 118
   ILC 119
   ILC 120
   ILC 121
   ILC 122
   ILC 123
   ILC 124
   ILC 125
   ILC 126
   ILC 127
   ILC 128
   ILC 129
   ILC 130
   ILC 131
   ILC 132
   ILC 133
   ILC 134
   ILC 135
   ILC 136
   ILC 137

```

DX21=X2-X1	ILC 138
DY21=Y2-Y1	ILC 139
CSPV=CS0221	ILC 140
CS0221=DX02*DX21+DY02*DY21	ILC 141
IF(CS0221) 71,71,74	ILC 142
71 IF(DX01*DX21+DY01*DY21) 73,72,72	ILC 143
72 IF(DY01*DX21-DX01*DY21) 76,76,74	ILC 144
73 IF(CSPV) 74,74,75	ILC 145
74 CONTINUE	ILC 146
ILO=1	ILC 147
75 ITO=ILO-1	ILC 148
IF(ITO.EQ.0) ITO=NLO	ILC 149
GO TO 77	ILC 150
76 ITO=ILO	ILC 151
77 ITO=ITO*NTL+ILO	ILC 152
C NORMAL EXIT	ILC 153
80 ITI=ITO	ILC 154
ITIPV=ITO	ILC 155
RETURN	ILC 156
END	ILC 157

SUBROUTINE IDPDRV(XD,YD,ZD,IPN,PD)	IPD 001
C THIS SUBROUTINE ESTIMATES PARTIAL DERIVATIVES OF THE FIRST AND	IPD 002
C SECOND ORDER AT THE DATA POINTS.	IPD 003
C THE INPUT PARAMETERS ARE	IPD 004
C XD,YD,ZD = ARRAYS STORING THE X, Y, AND Z COORDINATES,	IPD 005
C RESP., OF DATA POINTS,	IPD 006
C IPN = ARRAY STORING THE POINT NUMBERS OF NNP DATA	IPD 007
C POINTS NEAREST TO EACH OF THE DATA POINTS,	IPD 008
C WHERE NNP IS THE NUMBER OF DATA POINTS USED FOR ESTIMATION	IPD 009
C OF PARTIAL DERIVATIVES AT EACH DATA POINT. NNP IS GIVEN	IPD 010
C THROUGH THE IDGM COMMON.	IPD 011
C THE OUTPUT PARAMETER IS	IPD 012
C PD = ARRAY OF DIMENSION 5*NDP, WHERE THE ESTIMATED	IPD 013
C ZX, ZY, ZXX, ZXY, AND ZYY VALUES AT THE DATA	IPD 014
C POINTS ARE TO BE DISPLAYED,	IPD 015
C WHERE NDP IS THE TOTAL NUMBER OF DATA POINTS. NDP IS GIVEN	IPD 016
C THROUGH THE IDGM COMMON.	IPD 017
C DFCLARATION STATEMENTS	IPD 018
DIMENSION XD(10),YD(10),ZD(10),IPN(100),PD(50)	IPD 019
COMMON/IDGM/NDP,NNP,NT,NL	IPD 020
REAL NMX,NMY,NMZ,NMXX,NMXY,NMYX,NMYY	IPD 021
C PRELIMINARY PROCESSING	IPD 022
10 NDPO=NDP	IPD 023
NNPO=NNP	IPD 024
NNPM1=NNPO-1	IPD 025
C ESTIMATION OF ZX AND ZY	IPD 026
20 JPDO=-5	IPD 027
JIPNO=-NNPO	IPD 028
DO 24 IPO=1,NDPO	IPD 029
JPDO=JPDO+5	IPD 030
X0=XD(IPO)	IPD 031
Y0=YD(IPO)	IPD 032
Z0=ZD(IPO)	IPD 033
NMX=0.0	IPD 034
NMY=0.0	IPD 035
NMZ=0.0	IPD 036
JIPNO=JIPNO+NNPO	IPD 037
DO 23 IN1=1,NNPM1	IPD 038
JIPN=JIPNO+IN1	IPD 039
IPI=IPN(JIPN)	IPD 040
DX1=XD(IPI)-X0	IPD 041

```

      DY1=YD(IPI)-Y0          IPD 042
      DZ1=ZD(IPI)-Z0          IPD 043
      IN2MN=IN1+1              IPD 044
      DO 22 IN2=IN2MN+NNPO    IPD 045
         JIPN=JIPNO+IN2        IPD 046
         IPI=IPN(JIPN)          IPD 047
         DX2=XD(IPI)-X0        IPD 048
         DY2=YD(IPI)-Y0        IPD 049
         DZ2=ZD(IPI)-Z0        IPD 050
         DNMX=DY1*DZ2-DZ1*DY2  IPD 051
         DNMY=DZ1*DX2-DX1*DZ2  IPD 052
         DNMZ=DX1*DY2-DY1*DX2  IPD 053
         IF(DNMZ.GE.0.0)       GO TO 21  IPD 054
         DNMX=-DNMX             IPD 055
         DNMY=-DNMY             IPD 056
         DNMZ=-DNMZ             IPD 057
21      NMX=NMX+DNMX           IPD 058
         NMY=NMY+DNMY           IPD 059
         NMZ=NMZ+DNMZ           IPD 060
22      CONTINUE               IPD 061
23      CONTINUE               IPD 062
         PD(JPD0+1)=-NMX/NMZ   IPD 063
         PD(JPD0+2)=-NMY/NMZ   IPD 064
24      CONTINUE               IPD 065
C ESTIMATION OF ZXX, ZXY, AND ZYY
20      JPD0=-5                IPD 066
         JIPNO=-NNPO            IPD 067
         DO 34 TPO=1,NDPO        IPD 068
         JPD0=JPD0+5             IPD 069
         X0=XD(IPO)              IPD 070
         Y0=YD(IPO)              IPD 071
         ZX0=PD(JPD0+1)          IPD 072
         ZY0=PD(JPD0+2)          IPD 073
         NMXX=0.0                 IPD 074
         NMXY=0.0                 IPD 075
         NMYX=0.0                 IPD 076
         NMYY=0.0                 IPD 077
         NMZ =0.0                  IPD 078
         JIPNO=JIPNO+NNPO        IPD 079
         DO 33 IN1=1,NNPM1        IPD 080
         JIPN=JIPNO+IN1           IPD 081
         IPI=IPN(JIPN)            IPD 082
         DX1=XD(IPI)-X0           IPD 083
         DY1=YD(IPI)-Y0           IPD 084
         JPD=5*(IPI-1)            IPD 085
         DZX1=PD(JPD+1)-ZX0        IPD 086
         DZY1=PD(JPD+2)-ZY0        IPD 087
         IN2MN=IN1+1              IPD 088
         DO 32 IN2=IN2MN+NNPO    IPD 089
         JIPN=JIPNO+IN2           IPD 090
         IPI=IPN(JIPN)            IPD 091
         DX2=XD(IPI)-X0           IPD 092
         DY2=YD(IPI)-Y0           IPD 093
         JPD=5*(IPI-1)            IPD 094
         DZX2=PD(JPD+1)-ZX0        IPD 095
         DZY2=PD(JPD+2)-ZY0        IPD 096
         DNMXX=DY1*DZ2-DZ1*DY2  IPD 097
         DNMYX=DZX1*DX2-DX1*DZ2  IPD 098
         DNMYX=DY1*DZY2-DZY1*DY2  IPD 099
         DNMYY=DZY1*DX2-DX1*DZY2  IPD 100
         DNMZ =DX1*DY2 -DY1*DX2  IPD 101
         IF(DNMZ.GE.0.0)       GO TO 31  IPD 102
         DNMXX=-DNMXX             IPD 103
         DNMYX=-DNMXY             IPD 104
                                         IPD 105

```

	DNMYX=-DNMYX	IPD 106
	DNMYY=-DNMYY	IPD 107
	DNMZ =-DNMZ	IPD 108
31	NMXX=NMXX+DNMXX	IPD 109
	NMXY=NMXY+DNMXY	IPD 110
	NMYX=NMYX+DNMYX	IPD 111
	NMYY=NMYY+DNMYY	IPD 112
	NMZ =NMZ +DNMZ	IPD 113
32	CONTINUE	IPD 114
33	CONTINUE	IPD 115
	PD(JPDn+3)=-NMXX/NMZ	IPD 116
	PD(JPDn+4)=-(NMXY+NMYX)/(2.0*NMZ)	IPD 117
	PD(JPDn+5)=-NMYY/NMZ	IPD 118
34	CONTINUF	IPD 119
	RETURN	IPD 120
	END	IPD 121

	SUBROUTINE IDPTIP(XD,YD,ZD,IPT,IPL,PDD,ITI,XII,YII,ZII)	IPI 001
C	THIS SUBROUTINE PERFORMS PUNCTUAL INTERPOLATION OR EXTRAPOLATION, I.E., DETERMINES THE Z VALUE AT A POINT.	IPI 002
C	THE INPUT PARAMETERS ARE	IPI 003
C	XD,YD,ZD = ARRAYS STORING THE X, Y, AND Z COORDINATES, RESP., OF DATA POINTS,	IPI 004
C	IPT = ARRAY STORING THE POINT NUMBERS OF THE VERTEXES OF THE TRIANGLES,	IPI 005
C	IPL = ARRAY STORING THE POINT NUMBERS OF THE END POINTS OF THE BORDER LINE SEGMENTS AND THEIR RESPECTIVE TRIANGLE NUMBERS,	IPI 006
C	PDD = ARRAY STORING THE PARTIAL DERIVATIVES AT THE DATA POINTS,	IPI 007
C	ITI = TRIANGLE NUMBER OF THE TRIANGLE IN WHICH THE INTERPOLATED POINT LIES,	IPI 008
C	XII,YII = X AND Y COORDINATES, RESP., OF THE INTERPOLATED POINT.	IPI 009
C	THE OUTPUT PARAMETER IS	IPI 010
C	ZII = INTERPOLATED Z VALUE.	IPI 011
C	DECLARATION STATEMENTS	IPI 012
	DIMENSION XD(10),YD(10),ZD(10),IPT(100),IPL(100),PDD(50)	IPI 013
	COMMON/IDGM/NDP,NNP,NT,NL	IPI 014
	COMMON/IDPI/NCF,ICF	IPI 015
	DIMENSION CFO(27)	IPI 016
	EQUIVALENCE (X0,CFO(1)), (Y0,CFO(2)), (AP,CFO(3)),	IPI 017
1	(BP,CFO(4)), (CP,CFO(5)), (DP,CFO(6)),	IPI 018
2	(P00,CFO(7)), (P10,CFO(8)), (P20,CFO(9)),	IPI 019
3	(P30,CFO(10)), (P40,CFO(11)), (P50,CFO(12)),	IPI 020
4	(P01,CFO(13)), (P11,CFO(14)), (P21,CFO(15)),	IPI 021
5	(P31,CFO(16)), (P41,CFO(17)), (P02,CFO(18)),	IPI 022
6	(P12,CFO(19)), (P22,CFO(20)), (P32,CFO(21)),	IPI 023
7	(P03,CFO(22)), (P13,CFO(23)), (P23,CFO(24)),	IPI 024
8	(P04,CFO(25)), (P14,CFO(26)), (P05,CFO(27))	IPI 025
	DIMENSION CF(980)	IPI 026
	DIMENSION X(3),Y(3),Z(3),PD(15),	IPI 027
1	ZU(3),ZV(3),ZUU(3),ZUV(3),ZVV(3)	IPI 028
	EQUIVALENCE (ITO,FLITO),(ITJ,FLITJ)	IPI 029
	REAL LU,LV,LUSNUV,LVSNUV	IPI 030
	EQUIVALENCE (P5,P05)	IPI 031
	DATA NCFMX/35/	IPI 032
C	SETTING OF SOME LOCAL VARIABLES.	IPI 033
10	ITO=ITI	IPI 034
	XIO=XII	IPI 035
	YIO=YII	IPI 036
	NTL=NT+NL	IPI 037
		IPI 038
		IPI 039
		IPI 040
		IPI 041
		IPI 042
		IPI 043
		IPI 044
		IPI 045

```

C DETERMINES IF SIMPLE INTERPOLATION IS APPLICABLE.          IPI 046
20 IF(ITn.EQ.NTL)      GO TO 30                           IPI 047
    IL1=IT0/NTL                                         IPI 048
    IL2=IT0-IL1*NTL                                     IPI 049
    IL1T3=IL1*3                                         IPI 050
    IL2T3=IL2*3                                         IPI 051
    IT0=IPL(IL1T3)                                      IPI 052
    IF(IL1.EQ.IL2)      GO TO 40                           IPI 053
C CALCULATION OF ZII BY SIMPLE INTERPOLATION OR EXTRAPOLATION. IPI 054
30 ASSIGN 31 TO LBL                                     IPI 055
    GO TO 50                                         IPI 056
31 ZII=ZIO                                         IPI 057
    RETURN                                         IPI 058
C CALCULATION OF ZII AS A WEIGHTED MEAN OF TWO EXTRAPOLATED IPI 059
C VALUES.                                              IPI 060
40 ASSIGN 41 TO LBL                                     IPI 061
    GO TO 50                                         IPI 062
41 ZI1=ZIO                                         IPI 063
    IT0=IPL(IL2T3)                                      IPI 064
    ASSIGN 42 TO LBL                                     IPI 065
    GO TO 50                                         IPI 066
42 ZI2=ZIO                                         IPI 067
C CALCULATES THE WEIGHTING COEFFICIENTS FOR EXTRAPOLATED VALUES. IPI 068
45 IP1=IPL(IL1T3-2)                                     IPI 069
    IP2=IPL(IL1T3-1)                                     IPI 070
    IP3=IPL(IL2T3-1)                                     IPI 071
    X1=XD(IP1)                                         IPI 072
    Y1=YD(IP1)                                         IPI 073
    X2=XD(IP2)                                         IPI 074
    Y2=YD(IP2)                                         IPI 075
    X3=XD(IP3)                                         IPI 076
    Y3=YD(IP3)                                         IPI 077
    DX02=X10-X2                                         IPI 078
    DY02=Y10-Y2                                         IPI 079
    DX32=X3-X2                                         IPI 080
    DY32=Y3-Y2                                         IPI 081
    DX21=X2-X1                                         IPI 082
    DY21=Y2-Y1                                         IPI 083
    W1=(DX02*DX32+DY02*DY32)**2/(DX32*DX32+DY32*DY32) IPI 084
    W2=(DX02*DX21+DY02*DY21)**2/(DX21*DX21+DY21*DY21) IPI 085
C CALCULATES ZII AS A WEIGHTED MEAN.                      IPI 086
46 ZII=(W1*ZI1+W2*ZI2)/(W1+W2)                         IPI 087
    RETURN                                         IPI 088
C INTERNAL ROUTINE FOR PUNCTUAL INTERPOLATION.           IPI 089
C CHECKS IF THE NECESSARY CFO VALUES ARE SAVED.          IPI 090
50 IF(NCF.EQ.0)      GO TO 60                           IPI 091
    JCF=-27                                         IPI 092
    DO 51   LCF=1,NCF                                     IPI 093
        JCF=JCF+28                                       IPI 094
        FLITJ=CF(JCF)                                     IPI 095
        IF(ITn.EQ.ITJ)      GO TO 70                     IPI 096
51 CONTINUE                                         IPI 097
C CALCULATION OF NEW CFO VALUES.                        IPI 098
C DETERMINES THE COEFFICIENTS FOR THE COORDINATE SYSTEM TRANS- IPI 099
C FORMATION FROM THE X-Y SYSTEM TO THE U-V SYSTEM, AND CALCU- IPI 100
C LATES THE COEFFICIENTS OF THE POLYNOMIAL FOR INTERPOLATION. IPI 101
C LOADS COORDINATE AND PARTIAL DERIVATIVE VALUES AT THE IPI 102
C VERTEXES.                                              IPI 103
60 JIPT=3*(IT0-1)                                     IPI 104
    JPD=0                                         IPI 105
    DO 62   I=1,3                                     IPI 106
        JIPT=JIPT+1                                    IPI 107
        IDP=IP1(JIPT)                                 IPI 108
        X(I)=XD(IDP)                                 IPI 109
        Y(I)=YD(IDP)                                 IPI 110

```

```

Z(I)=ZD(IDP) IPI 111
JPDD=5*(IDP-1) IPI 112
DO 61 KPD=1,5 IPI 113
    JPD=JPD+1 IPI 114
    JPDD=JPDD+1 IPI 115
    PD(JPD)=PDD(JPDD) IPI 116
61   CONTINUE IPI 117
62   CONTINUE IPI 118
C DETERMINING THE COEFFICIENTS FOR THE COORDINATE SYSTEM IPI 119
C TRANSFORMATION FROM THE X-Y SYSTEM TO THE U-V SYSTEM IPI 120
C AND VICE VERSA IPI 121
63 X0=X(1) IPI 122
Y0=Y(1) IPI 123
A=X(2)-X0 IPI 124
B=X(3)-X0 IPI 125
C=Y(2)-Y0 IPI 126
D=Y(3)-Y0 IPI 127
AD=A*D IPI 128
BC=B*C IPI 129
DLT=AD-BC IPI 130
AP= D/DLT IPI 131
BP=-B/DLT IPI 132
CP=-C/DLT IPI 133
DP= A/DLT IPI 134
C CONVERSION OF THE PARTIAL DERIVATIVES AT THE VERTEXES OF THE IPI 135
C TRIANGLE FOR THE U-V COORDINATE SYSTEM IPI 136
64 AA=A*A IPI 137
ACT2=2.*0*A*C IPI 138
CC=C*C IPI 139
AB=A*B IPI 140
ADBC=AD+BC IPI 141
CD=C*D IPI 142
BB=B*B IPI 143
BDT2=2.*0*B*D IPI 144
DD=D*D IPI 145
DO 65 I=1,3 IPI 146
    JPD=5*I IPI 147
    ZU(I)=A*PD(JPD-4)+C*PD(JPD-3) IPI 148
    ZV(I)=B*PD(JPD-4)+D*PD(JPD-3) IPI 149
    ZUU(I)=AA*PD(JPD-2)+ACT2*PD(JPD-1)+CC*PD(JPD) IPI 150
    ZUV(I)=AB*PD(JPD-2)+ADBC*PD(JPD-1)+CD*PD(JPD) IPI 151
    ZVV(I)=BB*PD(JPD-2)+BDT2*PD(JPD-1)+DD*PD(JPD) IPI 152
65   CONTINUE IPI 153
C CALCULATION OF THE COEFFICIENTS OF THE POLYNOMIAL IPI 154
66 P00=Z(1) IPI 155
P10=ZU(1) IPI 156
P01=ZV(1) IPI 157
P20=0.5*ZUU(1) IPI 158
P11=ZUV(1) IPI 159
P02=0.5*ZVV(1) IPI 160
H1=Z(2)-P00-P10-P20 IPI 161
H2=ZU(2)-P10-ZUU(1) IPI 162
H3=ZUU(2)-ZUU(1) IPI 163
P30= 10.0*H1-4.0*H2+0.5*H3 IPI 164
P40=-15.0*H1+7.0*H2 -H3 IPI 165
P50= 6.0*H1-3.0*H2+0.5*H3 IPI 166
H1=Z(3)-P00-P01-P02 IPI 167
H2=ZV(3)-P01-ZVV(1) IPI 168
H3=ZVV(3)-ZVV(1) IPI 169
P03= 10.0*H1-4.0*H2+0.5*H3 IPI 170
P04=-15.0*H1+7.0*H2 -H3 IPI 171
P05= 6.0*H1-3.0*H2+0.5*H3 IPI 172
LU=SQRT(AA+CC) IPI 173
LV=SQRT(BB+DD) IPI 174
THXU=ATAN2(C,A) IPI 175

```

```

THUV=ATAN2(D,B1)-THXU
CSUV=COS(THUV)                                IPI 176
P41=5.0*LV*CSUV/LU*P50                         IPI 177
P14=5.0*LU*CSUV/LV*P05                         IPI 178
H1=ZV(2)-P01-P11-P41                          IPI 179
H2=ZUV(2)-P11-4.0*P41                         IPI 180
P21= 3.0*H1-H2                                 IPI 181
P31=-2.0*H1+H2                                IPI 182
H1=ZU(3)-P10-P11-P14                          IPI 183
H2=ZUV(3)-P11-4.0*P14                         IPI 184
P12= 3.0*H1-H2                                 IPI 185
P13=-2.0*H1+H2                                IPI 186
THUS=ATAN2(D+C,B-A)-THXU                      IPI 187
THSV=THUV-THUS                                  IPI 188
SNUV=SIN(THUV)                                IPI 189
LUSNUV=LU*SNUV                                 IPI 190
LVSNUV=LV*SNUV                                 IPI 191
AA= SIN(THSV)/LUSNUV                           IPI 192
BB=-COS(THSV)/LUSNUV                           IPI 193
CC= SIN(THUS)/LVSNUV                           IPI 194
DD= COS(THUS)/LVSNUV                           IPI 195
AC=AA*CC                                       IPI 196
AD=AA*DD                                       IPI 197
BC=BB*CC                                       IPI 198
G1=AA*AC*(3.0*BC+2.0*AD)                      IPI 199
G2=CC*AC*(3.0*AD+2.0*BC)                      IPI 200
H1=-AA*AA*AA*(5.0*AA*BB*P50+(4.0*RC+AD)*P41)
1 -CC*CC*CC*(5.0*CC*DD*P05+(4.0*AD+BC)*P14) IPI 201
H2=0.5*ZVV(2)-P02-P12                         IPI 202
H3=0.5*ZUU(3)-P20-P21                         IPI 203
P22=(G1*H2+G2*H3-H1)/(G1+G2)                  IPI 204
P32=H2-P22                                     IPI 205
P23=H3-P22                                     IPI 206
P23=H3-P22                                     IPI 207
C SAVES THE CFO VALUES IN THE CF ARRAY.
67 IF(NCF.LT.NCFMX)      NCF=NCF+1           IPI 208
    ICF=ICF+1                                     IPI 209
    IF(ICF.GT.NCFMX)      ICF=1               IPI 210
    JCF=28*ICF-27                                IPI 211
    CF(JCF)=FLITO                                IPI 212
    DO 68   KCF=1,27                            IPI 213
        JCF=JCF+1                               IPI 214
        CF(JCF)=CF0(KCF)                         IPI 215
68 CONTINUE                                     IPI 216
    GO TO 80                                     IPI 217
C LOADS THE CFO VALUES FROM THE CF ARRAY.
70 DO 71   KCF=1,27                            IPI 218
    JCF=JCF+1                               IPI 219
    CFO(KCF)=CF(JCF)                         IPI 220
71 CONTINUE                                     IPI 221
C TRANSFORMATION OF THE COORDINATE SYSTEM FROM X-Y TO U-V
80 DX=XII-X0                                 IPI 222
    DY=YII-Y0                                 IPI 223
    U=AP*DX+BP*DY                            IPI 224
    V=CP*DX+DP*DY                            IPI 225
C EVALUATION OF THE POLYNOMIAL
85 P0=P00+U*(P10+U*(P20+U*(P30+U*(P40+U*P50)))) IPI 226
    P1=P01+U*(P11+U*(P21+U*(P31+U*P41)))       IPI 227
    P2=P02+U*(P12+U*(P22+U*P32))                 IPI 228
    P3=P03+U*(P13+U*P23)                         IPI 229
    P4=P04+U*P14                                IPI 230
    Z10=P0+V*(P1+V*(P2+V*(P3+V*(P4+V*P5)))) IPI 231
    GO TO LBL, (31,41,42)                         IPI 232
    END                                         IPI 233
                                                IPI 234
                                                IPI 235
                                                IPI 236
                                                IPI 237
                                                IPI 238

```

```

SUBROUTINE IDSFFT(MD,NDP,XD,YD,ZD,WK,NXI,NYI,XI,YI,ZI) ISF 001
C THIS SUBROUTINE PERFORMS SMOOTH SURFACE FITTING WHEN THE PRO- ISF 002
C JECTIONS OF THE DATA POINTS IN THE X-Y PLANE ARE IRREGULARLY ISF 003
C DISTRIBUTED IN THE PLANE. ISF 004
C THE INPUT PARAMETERS ARE ISF 005
C   MD = MODE OF COMPUTATION (MUST BE 1, 2, OR 3), ISF 006
C       = 1 FOR NEW XD-YD, ISF 007
C       = 2 FOR OLD XD-YD, NEW XI-YI, ISF 008
C       = 3 FOR OLD XD-YD, OLD XI-YI, ISF 009
C   NDP = NUMBER OF DATA POINTS (MUST BE 4 OR GREATER), ISF 010
C   XD = ARRAY OF DIMENSION NDP STORING THE X COORDINATES ISF 011
C       OF THE DATA POINTS, ISF 012
C   YD = ARRAY OF DIMENSION NDP STORING THE Y COORDINATES ISF 013
C       OF THE DATA POINTS, ISF 014
C   ZD = ARRAY OF DIMENSION NDP STORING THE Z COORDINATES ISF 015
C       OF THE DATA POINTS, ISF 016
C   WK = ARRAY OF DIMENSION (2*NDP+NNP+5)*NDP+NXI*NYI ISF 017
C       TO BE USED AS A WORK AREA, ISF 018
C   NXI = NUMBER OF OUTPUT GRID POINTS IN THE X COORDINATE ISF 019
C       (MUST BE 1 OR GREATER), ISF 020
C   NYI = NUMBER OF OUTPUT GRID POINTS IN THE Y COORDINATE ISF 021
C       (MUST BE 1 OR GREATER), ISF 022
C   XI = ARRAY OF DIMENSION NXI STORING THE X COORDINATES ISF 023
C       OF THE OUTPUT GRID POINTS, ISF 024
C   YI = ARRAY OF DIMENSION NYI STORING THE Y COORDINATES ISF 025
C       OF THE OUTPUT GRID POINTS, ISF 026
C WHERE NNP IS THE NUMBER OF ADDITIONAL DATA POINTS USED FOR ISF 027
C ESTIMATING PARTIAL DERIVATIVES AT EACH DATA POINT. THE VALUE ISF 028
C OF NNP MUST BE GIVEN THROUGH THE IDNN COMMON. NNP MUST BE 2 ISF 029
C OR GREATER, BUT SMALLER THAN NDP. ISF 030
C THE OUTPUT PARAMETER IS ISF 031
C   ZI = DOUBLY-DIMENSIONED ARRAY OF DIMENSION (NXI,NYI), ISF 032
C       WHERE THE INTERPOLATED Z VALUES AT THE OUTPUT ISF 033
C       GRID POINTS ARE TO BE DISPLAYED. ISF 034
C THE LUN CONSTANT IN THE DATA INITIALIZATION STATEMENT IS THE ISF 035
C LOGICAL UNIT NUMBER OF THE STANDARD OUTPUT UNIT AND IS, ISF 036
C THEREFORE, SYSTEM DEPENDENT. ISF 037
C DECLARATION STATEMENTS ISF 038
      DIMENSION XD(10),YD(10),ZD(10),WK(1000), ISF 039
      1           XI(10),YI(10),ZI(100) ISF 040
      COMMON/IDNN/NNP ISF 041
      COMMON/IDGM/NDPC,NNPC,NT,NL ISF 042
      COMMON/IDPI/NCF,ICF ISF 043
      EQUIVALENCE (FNDPO,NDPO),(FNDPPV,NDPPV), ISF 044
      1           (FNNPO,NNPO),(FNNPPV,NNPPV), ISF 045
      2           (FNXIO,NXIO),(FNXPV,NXIPV), ISF 046
      3           (FNYIO,NYIO),(FNYIPV,NYIPV), ISF 047
      4           (FNT,NT),(FNL,NL) ISF 048
      DATA LUN/6/ ISF 049
C SETTING OF SOME INPUT PARAMETERS TO LOCAL VARIABLES. (ALL MD) ISF 050
      10 MD0=MD ISF 051
      NDPO=NDP ISF 052
      NDPC=NDPO ISF 053
      NXIO=NXI ISF 054
      NYIO=NYI ISF 055
      NNPO=NNP ISF 056
      NNPC=NNPO ISF 057
C ERROR CHECK. (ALL MD) ISF 058
      20 IF(MD0.LT.1.OR.MD0.GT.3) GO TO 90 ISF 059
      IF(NDPO.LT.4) GO TO 90 ISF 060
      IF(NXIO.LT.1.OR.NYIO.LT.1) GO TO 90 ISF 061
      IF(NNPO.LT.2.OR.NNPO.GE.NDPO) GO TO 90 ISF 062
      IF(MDO.NE.1) GO TO 22 ISF 063
      21 WK(1)=FNDPO ISF 064
      WK(2)=FNNPO ISF 065

```

```

      GO TO 24
22 FNDPPV=WK(1) ISF 066
      FNNPPV=WK(2) ISF 067
      IF(NDPO*NE*NDPPV) GO TO 90 ISF 068
      IF(NNPO*NE*NNPPV) GO TO 90 ISF 069
      IF(MDO*NE*3) GO TO 24 ISF 070
23 FNXIPV=WK(3) ISF 071
      FNYIPV=WK(4) ISF 072
      IF(NXIO*NE*NXIPV) GO TO 90 ISF 073
      IF(NYIO*NE*NYIPV) GO TO 90 ISF 074
      GO TO 30 ISF 075
24 WK(3)=FNXIO ISF 076
      WK(4)=FNYIO ISF 077
C ALLOCATION OF STORAGE AREAS IN THE WK ARRAY. (ALL MD) ISF 078
30 NDNDM1=NDPO*(NDPO-1) ISF 079
      IW IPT=7 ISF 080
      IW IPL=IW IPT+NDNDM1 ISF 081
      IW IPN=IW IPL+NDNDM1 ISF 082
      IW PD=IW IPN+NDPO*NNPO ISF 083
      IW IT =IW PD +NDPO*5 ISF 084
      ISF 085
C DIVIDES THE X-Y PLANE INTO A NUMBER OF TRIANGLES AND ISF 086
C DETERMINES NNP POINTS NEAREST EACH DATA POINT. (MD=1) ISF 087
40 IF(MD.GT.1) GO TO 42 ISF 088
41 CALL IDGEOM(XD,YD,ZD,WK(IW IPT),WK(IW IPL),WK(IW IPN)) ISF 089
      WK(5)=FNT ISF 090
      WK(6)=FNL ISF 091
      GO TO 50 ISF 092
42 FNT=WK(5) ISF 093
      FNL=WK(6) ISF 094
C ESTIMATES PARTIAL DERIVATIVES AT ALL DATA POINTS. (ALL MD) ISF 095
50 CALL IDPDRV(XD,YD,ZD,WK(IW IPN),WK(IW PD)) ISF 096
C LOCATES ALL INTERPOLATED POINTS. (MD=1,2) ISF 097
60 IF(MDO*EQ.3) GO TO 70 ISF 098
      IXI=0 ISF 099
      JWIT=JWIT-1 ISF 100
      INC=-1 ISF 101
      DO 62 IYI=1,NYIO ISF 102
          INC=-INC ISF 103
          YII=YI(IYI) ISF 104
          DO 61 IXIO=1,NXIO ISF 105
              IXI=IXI+INC ISF 106
              JWIT=JWIT+INC ISF 107
              CALL IDLCTN(XD,YD,WK(IW IPT),WK(IW IPL),
                XI(IXI),YII,WK(JWIT)) ISF 108
1           ISF 109
61   CONTINUE ISF 110
      IXI=IXI+INC ISF 111
      JWIT=JWIT+INC+NXIO ISF 112
62   CONTINUE ISF 113
C INTERPOLATION OF THE ZI VALUES. (ALL MD) ISF 114
70 NCF=0 ISF 115
      ICF=0 ISF 116
      JWIT=JWIT-1 ISF 117
      IXI=0 ISF 118
      IZI=0 ISF 119
      INC=-1 ISF 120
      DO 72 IYI=1,NYIO ISF 121
          INC=-INC ISF 122
          YII=YI(IYI) ISF 123
          DO 71 IXIO=1,NXIO ISF 124
              JWIT=JWIT+INC ISF 125
              IXI=IXI+INC ISF 126
              IZI=IZI+INC ISF 127
              CALL IDPTIP(XD,YD,ZD,WK(IW IPT),WK(IW IPL),WK(IW PD),
                WK(JWIT),XI(IXI),YII,ZI(IZI)) ISF 128
1           ISF 129
71   CONTINUE ISF 130

```

```

JWIT=JWIT+INC+NXIO      ISF 131
IXI=IXI+INC      ISF 132
IZI=IZI+INC+NXIO      ISF 133
72 CONTINUE      ISF 134
C NORMAL EXIT      ISF 135
 80 RETURN      ISF 136
C ERROR EXIT      ISF 137
 90 WRITE (LUN,2090) MD0,NDP0,NXIO,NYI0,NNP0      ISF 138
    RETURN      ISF 139
C FORMAT STATEMENT FOR ERROR MESSAGE      ISF 140
2090 FORMAT(1X/41H *** IMPROPER INPUT PARAMETER VALUE(S)./
 1 7H MD =,I4,10X,5HNPD =,I6,10X,5HNXI =,I6,      ISF 142
 2 10X,5HNYI =,I6,10X,5HNNP =,I6/      ISF 143
 3 35H ERROR DETECTED IN ROUTINE     IDSFFT/)      ISF 144
END      ISF 145

```

```

BLOCK DATA      IBD 001
C THIS SUBPROGRAM ENTERS A NUMBER INTO THE NNP CONSTANT IN THE      IBD 002
C IDNN COMMON, WHERE NNP IS THE NUMBER OF ADDITIONAL DATA POINTS      IBD 003
C USED FOR ESTIMATING PARTIAL DERIVATIVES AT EACH DATA POINT IN      IBD 004
C THE IDBVIP/IDSFFT SUBPROGRAM PACKAGE. NNP IS SET TO 4      IBD 005
C INITIALLY BY THIS SUBPROGRAM.      IBD 006
COMMON/IDNN/NNP      IBD 007
DATA NNP/4/      IBD 008
END.      IBD 009

```