EPHEMERAL STREAM-AQUIFER INTERACTION

by

Peter James Dillon , B.E.(Civil)(Hons.)

A thesis presented to the Faculty of Engineering
of the University of Adelaide for the
degree of Doctor of Philosophy

The Department of Civil Engineering
University of Adelaide

March 1984

EPHEMERAL STREAM - AQUIFER INTERACTION PROGRAM LISTINGS.

BIEMCAL

2D STREAM-AQUIFER INTERACTION

```
      C-----------------------------------------------------------------------
      C
            PROGRAM BIEMCAL(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,TAPE8,
           1TAPE9,PLOT)
      C
      C           BOUNDARY INTEGRAL EQUATION METHOD FOR SOLVING POTENTIAL FLOW
      C           PROBLEMS IN A 2-D VERTICAL PLANE.   THIS PROGRAM IS SUITABLE FOR
      C            - CONFINED STEADY STATE FLOW
      C            - FREE SURFACE TRANSIENT FLOW
      C                - WITH OR WITHOUT RECHARGE TO THE FREE SURFACE
      C                - TIME VARYING BOUNDARY CONDITIONS
      C           THIS GIVES COMPREHENSIVE TREATMENT OF STREAM-AQUIFER INTERACTION.
      C           IT ALLOWS FOR TRANSITION BETWEEN HYDRAULIC CONNECTION AND
      C           DISCONNECTION OF THE STREAM.
      C           IT PROVIDES MASS BALANCE CHECKS AT EACH TIME STEP AND ALLOWS
      C           OPTIONAL SHIFTING OF NODES ON SHIFTING BOUNDARIES.
      C
      C           CALIBRATION AND VALIDATION ROUTINES PROVIDE PERFORMANCE MEASURES
      C           WHEN USED IN CONJUNCTION WITH DATA FILES CONTAINING OBSERVED BORE
      C           WATER ELEVATIONS AND GAUGING STATION DISCHARGES.
      C           LINE PRINTER OR CALCOMP PLOTS OF GROUNDWATER ELEVATION
      C           CHANGES AND DAILY DISCHARGE LOSSES ARE OPTIONAL.
      C
      C           INPUT FILES :
      C           TAPE5 - CONTROL PARAMETERS, BOUNDARY NODE COORDINATES, AND BOUNDARY
      C                   ELEMENT TYPES.
      C           TAPE8 - BORE WATER ELEVATION DATA (CALLED AHDFILE IN THESIS)
      C                   USED WHEN ICAL > 2
      C           TAPE7 - DAILY STREAM DISCHARGE AND METEOROLOGICAL DATA (CALLED
      C                   COMBMDP IN THESIS) USED WHEN ICAL > 3
      C           OUTPUT FILES
      C           TAPE6 - LINE PRINTER OUTPUT - HEAD (PHI) AND NORMAL HYDRAULIC
      C                   GRADIENT (PHIN OR PHINL) AT EACH NODE, STREAM HEAD (H2),
      C                   INFILTRATION RATES (W) AND MASS BALANCE SUMMARY AT EACH
      C                   TIME STEP.  ALSO PRINTS AT END A SUMMARY TABLE OF KEY
      C                   PARAMETERS FOR ALL TIME STEPS.
      C           TAPE9 - FILE OF MODEL AND PROTOTYPE DISCHARGE LOSSES AND
      C                   INDEPENDENT VARIABLES FOR MULTIPLE REGRESSION ANALYSIS
      C                   LATER USING BMDP PACKAGE PROGRAM P2R. (THESIS SECTION
      C                   9.8.8) WRITTEN WHEN ICAL > 5
      C           PLOT  - PLOT FILE USED BY IN-HOUSE CALCOMP FLATBED PLOTTER TO
      C                   PLOT PROTOTYPE AND MODEL GROUNDWATER ELEVATION CHANGES
      C                   AT PIEZOMETERS AND DISCHARGE LOSSES BETWEEN GAUGING
      C                   STATIONS.
      C     LINE SUFFIX MARKERS--
      C       GMJO----0   :  TEXT FROM ORIGINAL PROGRAM OF J. LIGGETT (CORNELL UNIV.)
      C       BLANK       :  TEXT CODED BY P. DILLON (UNIV. OF ADELAIDE)
      C       CAL         :  TEXT CODED BY P. DILLON AND USED ONLY IN CALIBRATION MODE.
      C                      THIS APPLIES TO WHOLE SUBROUTINE WHEN APPEARING IN
      C                      SUBROUTINE HEADER STATEMENT.
      C
      C     ORIGINAL PROGRAM BY J.LIGGETT,CORNELL UNIV,ITHICA,NEW YORK
      C     MODIFIED AND DEVELOPED BY P.DILLON,UNIV OF ADELAIDE,SOUTH
      C     AUSTRALIA 1982-83. SUBMITTED BY P. DILLON IN MARCH 1984 AS
      C     PART OF PHD. THESIS ENTITLED - EPHERMERAL STREAM-AQUIFER INTERACTION -
      C     TO THE FACULTY OF ENGINEERING, UNIVERSITY OF ADELAIDE.
      C
```

```
      C   GLOSSARY OF MAIN TERMS
      C     CBTA(I)   - COS OF ANGLE OF FREE SURFACE SLOPE AT NODE, I
   60 C     CORANG(I) - INTERIOR ANGLE OF BOUNDARY AT NODE, I
      C     DLTT      - SIZE OF CURRENT TIME INCREMENT
      C     H2        - STREAM HEAD AT START OF TIME STEP
      C     H2N       - STREAM HEAD AT END OF TIME STEP
      C     ICAL      - CONTROL PARAMETER (CARRIED IN CONTROL PAZAMETER, IFLGC)
   65 C                 WHICH DICTATES WHETHER CALIBRATION IS TO OCCUR AND IF SO
      C                 THE TYPE OF CALIBRATION
      C     N         - NUMBER OF NODES
      C     NTYP(I)   - NODE TYPE OF NODE, I (SEE FIGURE 4.6)
      C     PHI(I)    - HYDRAULIC HEAD AT NODE, I
   70 C     PHIN(I)   - NORMAL HYD. GRADIENT LEAVING NODE, I (CLOCKWISE)
      C     PHINL(I)  - NORMAL HYD. GRADIENT APPROACHING NODE, I (CLOCKWISE)
      C     RHS       - VECTOR OF KNOWNS (RIGHT HAND SIDE) OF SYSTEM OF LINEAR
      C                 EQNS  (ALSO USED TO STORE SOLVED UNKNOWNS (IN FSOLVE)
      C     RLN       - MATRIX L (N*N) OF EQUATION 4.56  (IN INTG)
   75 C     RN        - MATRIX R (N*N) OF EQUATION 4.56  (IN INTG) THIS IS
      C                 OVERWRITTEN BY MATRIX U (N*N) OF EQUATION 4.37 (IN ASSMBL)
      C
      C
            DIMENSION TITLE(20),XINT(44),YINT(44),PHIP(30),NPSET(20)
   80       COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,ID1,IDL,IPR,TEMPX
            COMMON /CALQ/ ISN,Q(5),TRATIO,QLOSS(65),BEDINV,ND,QISN(65),RLENG
            COMMON /HIS/ NHIS,P(15,65),TM(65),NH(15)
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                    GMJ00080
   85      $      IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,             GMJ00090
           $      NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                           GMJ00120
           $      CORANG(44),FCORN(44)                                      GMJ00130
   90       COMMON /SOL/ RN(44,44),RHS(44),IPVT(44)                         GMJ00140
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
            COMMON /SHI/ XP(10),YP(10),NESH(10),NSH
            COMMON /SAV/ PHIO(44),PHINO(44),PHINLO(44)
            COMMON /BED/ WONE,B,H2,H2N,YM(22)
   95       INTEGER ITMP(24)
            REAL TMP(24),THIME(55)
            NHIS=65
            KOR=5                                                            GMJ00170
            KOW=6                                                            GMJ00180
  100       ILP=6                                                            GMJ00190
            ICR=5                                                            GMJ00200
            IFH=NNSB=NNSBP=IUNSAT=0
            ILOOP=ITIME=0
            ICAL=NBOR=MCH=0
  105    C   THRESHOLD REL MOVEMENT OF FREE SURFACE FOR ITERATION
            THRESH=0.05
            PERM=BTEMP=BTEMPP=1.                                               CAL
         C   INITIALIZE ARRAYS
            DO 143 I=1,44
  110     143 W(I,1)=W(I,2)=0.
            DO 141 I=1,44
            PHI(I)=PHIN(I)=PHINL(I)=0.
            PHIO(I)=PHINO(I)=PHINLO(I)=0.
          141 CONTINUE
```

```
      C
      C   READ AND ECHO PRINT INPUT DATA                                    GMJ00210
          WRITE (ILP,6010)
 6010 FORMAT (1H1,*BIEM, LINEAR ELEMENTS, TIME VARYING HEAD/RECHARGE *,
     12X,*( GMJS8 VN 23/05/83 )*)
          READ (ICR,5010) TITLE                                            GMJ00230
 5010 FORMAT (20A4)                                                        GMJ00240
          WRITE (ILP,6020) TITLE                                           GMJ00250
 6020 FORMAT (1X,*TITLE...   *,4--*,20A4,*--*)                             GMJ00260
      C    IFLGA=1 ..PRINT REFORMATTED INPUT
      C        >=2 ..CHECK MASS BALANCE AT EACH TIME STEP

      C        =N ..( 2 < N < 10 ) PRINT PHI HISTORY FOR N SELECTED NODES
      C    IFLGB=0 ..DO NOT PRINT MATRICES, CONSTANT SPECIFIED HEAD BDYS
      C          =1 ..PRINT MATRICES AT EACH TIME STEP
      C          =N ..RESET HEAD AT N NODES AT EACH TIME STEP
      C    IFLGC=0 ..NO FREE SURFACE OR INITIAL PHIN ON F.S. IS KNOWN
      C          =1 ..SETS THETA2=1 FOR FREE SURFACE POSITION CALCULATION
      C              ON FIRST TIME STEP (USE IF INITIAL PHIN ON F.S.UNKNOWN)
      C          =2 ..CALIBRATE ON MATCH POINTS FOR FREE SURFACE ONLY          CAL
      C          =3 ..    ..    USING FULL GROUNDWATER RECORDS               CAL
      C          =4 ..    ..    USING GW AND STREAMFLOW RECORDS              CAL
      C          =5 ..VALIDATE USING GW AND STREAMFLOW RECORDS               CAL
      C          =6 ..AS ABOVE WITH FLOW RESIDUAL REGRESSION FILE (TAPE9) CREATED CAL
      C          =7 ..AS ABOVE WITH TEMPERATURE DEPENDENT STREAMBED IMPEDANCE  CAL
      C    IFLGD=1 ..READ IN CONSTANT RECHARGE RATES
      C          =N ..READ IN CHANGING RECHARGE RATES FOR 1ST N NODES
      C              ON FREE SURFACE AT EACH TIME STEP
      C          =-N .FIRST N NODES ON FREE SURFACE MAY BECOME STREAMBED
      C              NODES BY HYDRAULIC CONNECTION WITH LOSING STREAM
          READ (ICR,*) IFLGA,IFLGB,IFLGC,IFLGD
          WRITE (ILP,6110)                                                 GMJ00680
 6110 FORMAT (/1X,*FLAG A, FLAG B, FLAG C, FLAG D*)
          WRITE (ILP,*) IFLGA,IFLGB,IFLGC,IFLGD
          IF(IFLGC.LE.1) GOTO 600                                          CAL
          ICAL=IFLGC                                                       CAL
          WRITE(ILP,1205)                                                  CAL
 1205 FORMAT(*   <<<<<< CALIBRATION RUN >>>>>> *)                          CAL
      C    NBOR = NO OF BORES                                              CAL
      C    ID1  = FIRST DAY OF RECORD                                      CAL
      C    IDL  = LAST  DAY OF RECORD                                      CAL
      C    IPR  = 0/1 DETAILED PRINT REQUIRED : NO/YES                     CAL
      C         = 2   LINE PRINTER PLOT OF OBSERVED AND MODEL G.W. ELEVATION CAL
      C                              CHANGES AND STREAM DISCHARGE LOSSES    CAL
      C         = 3   25CM CALCOMP PLOT OF ..    ..    ..    ..    ..    .. CAL
          READ(ICR,*) NBOR,ID1,IDL,IPR                                     CAL
          WRITE(ILP,1210) NBOR,ID1,IDL,IPR                                 CAL
          WRITE(ILP,1215)                                                  CAL
 1210 FORMAT(* NO OF BORES =*,I5,*  FIRST AND LAST DAY NOS. =*,            CAL
     1216,*  PRINT DETAILS (0/1/2/3) =*,I5)                                CAL
 1215 FORMAT(/,* BORE NO   DISTANCE   INITIAL EL   NO. MATCH PTS   *,      CAL
     1* MATCH DAY   MATCH EL *)                                           CAL
          MCH=0                                                            CAL
          DO 1220 I=1,NBOR                                                 CAL
          READ(ICR,*) NB(I),SL(I),ZPO(I),NMB,((TP(MCH+J),ZP(MCH+J)),       CAL
     1J=1,NMB)                                                             CAL
          WRITE(ILP,1230) NB(I),SL(I),ZPO(I),NMB,((TP(MCH+J),ZP(MCH+J)),   CAL
```

```
     1J=1,NMB)                                                                  CAL
      MB(I)=NMB                                                                 CAL
 1230 FORMAT(I10,F8.1,2X,F10.3,3X,I8,8X,2F10.3,/49X,2F10.3)                     CAL
      MCH=MCH+MB(I)                                                             CAL
 1220 CONTINUE                                                                  CAL
C                                                                               CAL
      IF(ICAL.LT.3) GOTO 1200                                                   CAL
C   READ GROUNDWATER RECORDS (TAPE8) AFTER MODEL RUN                            CAL
      READ(ICR,*) TRATIO                                                        CAL
      WRITE(ILP,1300) TRATIO                                                    CAL
 1300 FORMAT(* ADOPTED TIME RATIO = TMOD/TPROTO =K/NE =*,F10.5)                 CAL
C                                                                               CAL
      IF(ICAL.LT.4) GOTO 1200                                                   CAL
C   READ STREAMFLOW PARAMETERS AND RECORD (TAPE7)                               CAL
      READ(ICR,*) ISN,RLENG                                                     CAL
      WRITE(ILP,780) ISN,RLENG                                                  CAL
  780 FORMAT(* U/S STATION NO =*,I5,*   REACH LENGTH (M) =*,F10.0)              CAL
C                                                                               CAL
      IF(ICAL.LT.5) GOTO 1200                                                   CAL
C   READ HYDRAULIC CONDUCTIVITY FOR VALIDATION RUN                              CAL
      READ(ICR,*) PERM                                                          CAL
      WRITE(ILP,790) PERM                                                       CAL
  790 FORMAT(* HYDRAULIC CONDUCTIVITY =*,F10.5)                                 CAL
C                                                                               CAL
 1200 CONTINUE                                                                  CAL
      IFLGC=1                                                                   CAL
  600 CONTINUE                                                                  CAL
C   READ TIME PARAMETERS
C   THETA2 = TIME WEIGHTING FACTOR. FOR ALL THESIS RUNS THETA2=0.53 (SEE
C            SECTION 5.1.1)
C   DLTT   = SIZE OF FIRST TIME INCREMENT
C   TFACT  = TIME STEPPING CONTROL PARAMETER.
C            FOR TFACT < 10 : DLTT(K+1) = DLTT(K)*TFACT
C   IE. CONSTANT TIME STEPS IF TFACT = 1.  EXPONENTIALLY
C   INCREASING TIME STEPS  1 < TFACT < 1.5 (USUAL MAXIMUM).
C   FOR BASE 10 LOGARITHMIC TIME STEPS SET TFACT=10. (3 STEPS/LOG 10 CYCLE)
C   FOR SPECIFIED SET OF (NTHIME>10) TIME STEPS SET TFACT=REAL(NTHIME)
C   FOR ICAL > 2  TFACT=10). MAY BE USEFUL (SETS DLTT=TRATIO AFTER          CAL
C   SMALL INITIAL TIME STEPS)                                               CAL
C   TIME   = MODEL TIME AT START OF RUN.  TIME = 0 FOR ALL THESIS RUNS.
C   TMLMT  = MODEL TIME LIMIT. (NOT CP TIME LIMIT)
      READ(ICR,*) THETA2,DLTT,TFACT,TIME,TMLMT
      WRITE(ILP,7686)
 7686 FORMAT(* THETA2,TIME INCREMENT,TFACT,INITIAL TIME,TIME LIMIT*)
      WRITE(ILP,7687) THETA2,DLTT,TFACT,TIME,TMLMT
 7687 FORMAT(10(1PE13.6))
      TIME=TIME+DLTT
      READ (ICR,*) N                                                    GMJ00280
      WRITE (ILP,6160)                                                  GMJ00710
 6160 FORMAT (1X,*NUMBER OF NODE POINTS*)                               GMJ00720
      WRITE (ILP,*) N                                                   GMJ00730
      NP=N+1                                                            GMJ00290
      NM=N-1                                                            GMJ00300
      WRITE(ILP,125)
  125 FORMAT(*    I      X(I)          Y(I)*)
      DO 6000 I=1,N                                                     GMJ00310
      READ (ICR,*) X(I),Y(I)                                            GMJ00320
```

```
      WRITE(ILP,126) I,X(I),Y(I)
  126 FORMAT(1X,I5,2F10.5)
 6000 CONTINUE                                                    GMJ00330
      READ (ICR,*) NSEG                                           GMJ00340
      WRITE(ILP,127) NSEG
  127 FORMAT(1X,*NO OF SEGMENTS*,/,I5)
      NFS1=0
      NFSL=0
      WRITE(ILP,128)
  128 FORMAT(*  ITA  NTA  IST  IFSH  INB  (TMP(IJ),IJ=1,INB)*)
      NTNF=0                                                      GMJ00350
      NTNFP=1
      NSH=0
      DO 1040 II=1,NSEG                                           GMJ00360
C   ITA  =  SEGMENT NO.
C   NTA  =  NO. OF NODES IN SEGMENT (INCLUDING END NODES)
C   IST  =  1  ..PHIN SPECIFIED
C        =  2  ..PHI SPECIFIED
C        =  3  ..FREE SURFACE
C   IFSH =  0  ..THERE ARE NO SHIFTING NODES
C        =  1  ..THESE NODES MAY SHIFT
C   INB  =  NO. OF SPECIFIED NODAL VALUES (PHI OR PHIN) TO READ IN
C           IF(INB.LT.NTA) LAST TEMP ASSIGNED TO REMAINING NODES
C   TMP(IJ)=SPECIFIED NODAL VALUES
      READ (ICR,*) ITA,NTA,IST,IFSH,INB,(TMP(IJ),IJ=1,INB)
      WRITE (ILP,*) ITA,NTA,IST,IFSH,INB,(TMP(IJ),IJ=1,INB)
      NPSG(ITA)=NTA                                               GMJ00390
      IF (INB.EQ.NTA) GO TO 2058                                  GMJ00380
      MJLL=INB+1                                                  GMJ00400
      DO 1010 MJ=MJLL,NTA                                         GMJ00410
      TMP(MJ)=TMP(INB)                                            GMJ00420
 1010 CONTINUE                                                    GMJ00430
 2058 IF(IST.NE.3) GOTO 2060
      NFS1=NTNF+1
      NFSL=NFS1+NTA-1
 2060 DO 1110 MM=1,NTA                                            GMJ00440
      MMT=MM+NTNF                                                 GMJ00450
      IF (MMT.EQ.NP) MMT=1                                        GMJ00460
      NTYP(MMT)=IST                                               GMJ00470
      IF(IST.EQ.3) NTYP(MMT)=6
      IF (IST.EQ.2) PHI(MMT)=TMP(MM)                              GMJ00480
      IF ((IST.NE.2).AND.(MM.NE.1)) PHIN(MMT)=TMP(MM)             GMJ00490
      IF ((IST.NE.2).AND.(MM.EQ.1)) PHINL(MMT)=TMP(MM)            GMJ00500
 1110 CONTINUE                                                    GMJ00510
      IF (II.EQ.1) GO TO 2040                                     GMJ00520
      NTNFP=NTNF+1                                                GMJ00530
      IF ((ISTF.EQ.1).AND.(IST.EQ.1)) NTYP(NTNFP)=5               GMJ00540
      IF ((ISTF.EQ.1).AND.(IST.EQ.2)) NTYP(NTNFP)=3               GMJ00550
      IF ((ISTF.EQ.2).AND.(IST.EQ.1)) NTYP(NTNFP)=4               GMJ00560
      IF ((ISTF.EQ.2).AND.(IST.EQ.2)) NTYP(NTNFP)=?               GMJ00570
      IF(ISTF.EQ.1 .AND. IST.EQ.3) NTYP(NTNFP)=7
      IF(ISTF.EQ.2 .AND. IST.EQ.3) NTYP(NTNFP)=8
      IF(ISTF.EQ.3 .AND. IST.EQ.1) NTYP(NTNFP)=9
      IF(ISTF.EQ.3 .AND. IST.EQ.2) NTYP(NTNFP)=10
      GO TO 2050                                                  GMJ00580
 2040 IST1=IST                                                    GMJ00590
 2050 ISTF=IST                                                    GMJ00600
```

```
      IF (NESH.EQ.0 .OR. IST.EQ.3) GOTO 1030
      IL=NSH+1
      IM=NSH+2
      NESH(IL)=NTNFP
290   NESH(IM)=NTNF+NTA
      IF(NESH(IM).EQ.NP) NESH(IM)=1
      NSH=NSH+2
 1030 CONTINUE
      NTNF=NTNF+NTA-1                                                  GMJ00610
295   1040 CONTINUE                                                    GMJ00620
      IF ((ISTF.EQ.1).AND.(IST1.EQ.1)) NTYP(1)=5                       GMJ00630
      IF ((ISTF.EQ.1).AND.(IST1.EQ.2)) NTYP(1)=3                       GMJ00640
      IF ((ISTF.EQ.2).AND.(IST1.EQ.1)) NTYP(1)=4                       GMJ00650
      IF ((ISTF.EQ.2).AND.(IST1.EQ.2)) NTYP(1)=2                       GMJ00660
300   IF(NFS1.EQ.0) GOTO 6625
      IF(IST1.EQ.3 .AND. ISTF.EQ.1) NTYP(1)=7
      IF(IST1.EQ.3 .AND. ISTF.EQ.2) NTYP(1)=8
      IF(IST1.EQ.1 .AND. ISTF.EQ.3) NTYP(1)=9
      IF(IST1.EQ.2 .AND. ISTF.EQ.3) NTYP(1)=10
305   C
      DO 201 I=NFS1,NFSL
      PHI(I)=Y(I)
      IF(IFLGD.EQ.0) GOTO 220
      IF(NTYP(I).EQ.7 .OR.NTYP(I).EQ.8) GOTO 230
310   W(I,1)=W(I,2)*PHIN(I)
      GOTO 220
  230 W(I,1)=W(I,2)*PHINL(I)
  220 IF(NTYP(I).NE.8 .AND. NTYP(I).NE.10) GOTO 210
      IF(NTYP(I).EQ.8) PHINL(I)=0.
315   IF(NTYP(I).EQ.10) PHIN(I)=0.
      PHIP(I)=PHI(I)
  210 CONTINUE
  201 CONTINUE
      C   READ IN INITIAL RECHARGE RATE, STREAMBED POSITION, STREAM HEAD
320   C   FOR CASES WHICH MAY BECOME HYDRAULICALLY CONNECTED
      IF(IFLGD.GE.0) GOTO 204
      NRB=NFS1-IFLGD-1
      READ(ICR,*) B
      WRITE(ILP,3950) B
325   BO=B                                                             CAL
 3950 FORMAT(* INITIAL STREAMBED HYDRAULIC IMPEDANCE, B =*,F10.5)
      READ(ICR,*) H2,IFH
      WRITE(ILP,4000) H2,IFH
 4000 FORMAT(* INITIAL STREAM HEAD = *,F10.5,*  VARIES WITH TIME*,
330  1* (YES=1/NO=0) =*,I2)
      H2N=H2
      READ(ICR,*) HC
      WRITE(ILP,4150) HC
 4150 FORMAT(* SUCTION HEAD IN UNSAT ZONE (MODULUS) = *,F10.5)
335   READ(ICR,*) (YM(I),I=NFS1,NRB)
      BEDINV=YM(NFS1)
      IF(ICAL.LT.4) GOTO 900
      C   READ DISCHARGE FOR FIRST DAY AND CALCULATE STREAM STAGE (H2N)  CAL
      TREAD=0.                                                         CAL
340   CALL STREAM(HA,HP,H2,TIME)                                       CAL
      WRITE(ILP,920) Q(ISN),H2                                         CAL
  920 FORMAT(* INITIAL DISCHARGE (ML/DAY)=*,F10.3,*   INITIAL STREAM*, CAL
```

```
      1* HEAD (M) =*,F10.3)                                                    CAL
       H2N=H2                                                                  CAL
345   C   STREAMBED IMPEDANCE IS TEMPERATURE DEPENDENT IF ICAL.GE.7.           CAL
       IF(ICAL.LT.7) GOTO 900                                                  CAL
       BTEMP=TCOMP(TEMPX)                                                      CAL
       WRITE(ILP,4600) BTEMPP,BTEMP,B,TEMPX                                    CAL
 4600 FORMAT(* CURRENT AND NEXT TEMP CORRECTION*,2F10.3,*   NEW B =*,          CAL
350     1F10.3,*   NEXT TEMP =*,F6.1)                                          CAL
       BTEMPP=BTEMP                                                            CAL
  900 WONE=(H2+HC-YM(NFS1))/B
      WRITE(ILP,4050) WONE
 4050 FORMAT(* INITIAL RECHARGE RATE AT CENTRELINE =*,F10.5)
355   WRITE(ILP,4100)
 4100 FORMAT(*   NODE       X            Y          YMAX          PHI*,
     1*           W*)
      DO 4200 I=NFS1,NRB
      IF(YM(I).GT.H2) GOTO 4200
360   W(I,1)=W(I,2)=(H2+HC-YM(I))/B
 4200 WRITE(ILP,4300) I,X(I),Y(I),YM(I),PHI(I),W(I,1)
 4300 FORMAT(1X,I5,5(2X,F10.5))
      BB=1./B
      WRITE(ILP,4400) BB
365  4400 FORMAT(* LEAKAGE FACTOR (K/B)= *,F10.5)
      EL=YM(NFS1)-PHI(NFS1)
      H2HC=H2+HC
      TD=EL-H2HC*ALOG((H2HC+EL)/H2HC)
      WRITE(ILP,4450) TD
370  4450 FORMAT(* INITIAL RECHARGE DELAY TIME = *,F10.5)
  204 CONTINUE
      NFS=NFS1
      IF(IFLGA.EQ.0) GOTO 6627
      WRITE(ILP,6629)
375   WRITE(ILP,6630) (I,PHI(I),I=NFS1,NFSL)
 6629 FORMAT(1X,* INITIAL FREE SURFACE POSITIONS*)
 6625 IF(IFLGA.GE.2) CALL CHECK
 6627 ITIME=1
C
380  C   FOR ALL THESIS RUNS INTNL=0 AS NO INTERNAL SOLUTIONS REQUIRED.
C    CODE FOR INTERNAL SOLNS (SEE EQN 4.9) IS NOT LISTED.
       READ(ICR,*) INTNL
       WRITE(ILP,130) INTNL
  130 FORMAT(* NO OF INTERNAL POINTS FOR SOLVING*,/,I5)
385    IF(INTNL.EQ.0) GO TO 6100
       WRITE(ILP,131)
  131 FORMAT(*  I    XINT(I)     YINT(I)*)
       DO 6660 I=1,INTNL
       READ(ICR,*) XINT(I),YINT(I)
390  6660 WRITE(ILP,*) I,XINT(I),YINT(I)
 6100 IF(IFLGB.LE.1) GOTO 6130
C    SPECIFY NODES AT WHICH HEAD WILL CHANGE
C    WARNING - IF REDUCING HEAD OF FREE SURFACE NODE, NEW HEAD
C              MUST EXCEED Y CO-ORD OF ADJACENT UNDERLYING
395  C              SPECIFIED HEAD NODE AT CURRENT TIME STEP.
C              TO MAKE LARGE HEAD CHANGES, USE SMALL TIME STEPS
C              AND SHIFTING NODES.
       READ(ICR,*) (NPSET(I),I=1,IFLGB)
       WRITE(ILP,6105) (NPSET(I),I=1,IFLGB)
```

```
      6105 FORMAT(1X,*BDY NODES AT WHICH HEAD WILL CHANGE*,/,2015)
           NSETFS=0
           IF(NFS1.EQ.0) GOTO 6230
           DO 6120 I=1,IFLGB
           J=NTYP(NPSET(I))
           IF(J.EQ.8 .OR. J.EQ.10) NSETFS=NPSET(I)
      6120 CONTINUE
           IF(NSETFS.NE.0) WRITE(ILP,6121) NSETFS
      6121 FORMAT(1X,*NODE *,I3,* IS ALSO A FREE SURFACE NODE*)
      C    LIST PROCESSED INPUT IF REQUIRED
      6130 IF (IFLGA.EQ.0) GO TO 6200                                    GMJ00670
           IF(IFLGA.LE.2) GOTO 9850
      C    HISTORY TRACKING OF POTENTIAL AT UP TO 12 NODES
           WRITE(ILP,9600) IFLGA
      9600 FORMAT(/* HYDROGRAPHS FOR *,I4,* NODES*)
           READ(ICR,*) (NH(I),I=1,IFLGA)
           WRITE(ILP,9800) (NH(I),I=1,IFLGA)
      9800 FORMAT(* NODES *,12(I3))
      9850 CONTINUE
           WRITE(ILP,133)
       133 FORMAT(1X,///,* LIST PROCESSED INPUT*,/)
           WRITE (ILP,6140)                                              GMJ00740
      6140 FORMAT (1X,*   NODE     X          Y       NTYP    PHI*
          1,*        PHIN       PHINL*)
           DO 1050 IJ=1,N                                                GMJ00760
           WRITE (ILP,7692) IJ,X(IJ),Y(IJ),                              GMJ00770
          $     NTYP(IJ),PHI(IJ),PHIN(IJ),PHINL(IJ)                     GMJ00780
      7692 FORMAT(1X,I7,2F10.5,I4,3F10.5)
      1050 CONTINUE                                                      GMJ00790
           NSHSEG=NSH/2
           WRITE(ILP,1060) NSHSEG
      1060 FORMAT(* NO OF SEGMENTS WHOSE NODES MAY BE SHIFTED*,/,I5)
      6200 CONTINUE
           IF(ICAL.LT.2) GOTO 6400
      C    INTERPOLATE BETWEEN NODES FOR OBSERVATION BORE POSITIONS      CAL
           CALL INTERP                                                   CAL
           WRITE(ILP,6350)                                               CAL
           DO 6300 I=1,NBOR                                              CAL
           K=NDB1(I)                                                     CAL
           ZMO(I)=PHI(K)+CINT(I)*(PHI(K+1)-PHI(K))                       CAL
      6300 WRITE(ILP,6360) NB(I),ZMO(I)                                  CAL
      6350 FORMAT(* INITIAL OBS BORE WATER TABLE ELEVATIONS (MODEL)*)    CAL
      6360 FORMAT(1X,I10,F10.3)                                          CAL
      6400 IF(TFACT.LT.11.) GOTO 2020
           IF(TFACT.GT.99.) GOTO 6500
           NTHIME=INT(TFACT+0.00001)
           READ(ICR,*) (THIME(I),I=1,NTHIME)
           GOTO 6540
      6500 NTHIME=IDL-ID1+6                                              CAL
           THIME(1)=0.01*TRATIO                                          CAL
           THIME(2)=0.02*TRATIO                                          CAL
           THIME(3)=0.05*TRATIO                                          CAL
           THIME(4)=0.10*TRATIO                                          CAL
           THIME(5)=0.20*TRATIO                                          CAL
           THIME(6)=0.50*TRATIO                                          CAL
           DO 6520 I=7,NTHIME                                            CAL
      6520 THIME(I)=FLOAT(I-6)*TRATIO                                    CAL
```

```
      DLTT=THIME(1)                                                    CAL
      TIME=DLTT                                                        CAL
 6540 WRITE(ILP,7700)
460   7700 FORMAT(/* SPECIFIED SOLUTION TIMES (TFACT>11) *)
      WRITE(ILP,6630) (I,THIME(I),I=1,NTHIME)
C
C   START OF TIME LOOP
C
465   2020 CONTINUE
      IF(ICAL.LT.4) GOTO 950
      IF(ITIME.EQ.1) GOTO 970                                          CAL
      IF((TIME-TREAD).LT.(TRATIO*0.9999)) GOTO 960                     CAL
      TREAD=TREAD+TRATIO                                               CAL
470    970 H2=H2N                                                      CAL
      CALL STREAM(HA,HP,H2N,TIME)                                      CAL
      IF(ICAL.LT.7) GOTO 990                                           CAL
      BTEMP=TCOMP(TEMPX)                                               CAL
      B=BO*BTEMPP                                                      CAL
475   WRITE(ILP,4600) BTEMPP,BTEMP,B,TEMPX                             CAL
      BTEMPP=BTEMP                                                     CAL
      990 CONTINUE                                                     CAL
      IF((ND+ID1).LE.IDL) GOTO 2021                                    CAL
      ITIME=ITIME-1                                                    CAL
480   GOTO 9000                                                        CAL
      960 H2=H2N                                                       CAL
      H2N=HP+(HA-HP)*(TIME-TREAD)/TRATIO                               CAL
      WRITE(6,980) H2N
      980 FORMAT(* H2N = *,F10.3)
485   GOTO 2021                                                        CAL
      950 IF(IFH.EQ.0) GOTO 2021
      IF(ITIME.EQ.1) GOTO 2021
      H2=H2N
C   READ NEW VALUE OF STREAM HEAD
490   READ(ICR,*) H2N
      700 CONTINUE
      WRITE(ILP,4500) TIME,H2N
      4500 FORMAT(* AT TIME = *,F10.5,* STREAM HEAD = *,F10.5)
      GOTO 2025
495   2021 IF(IFLGD.LE.1) GOTO 2025
C   RESET RECHARGE RATE AT EACH TIME STEP IF REQUIRED
      NRN=NFS1+IFLGD-1
      IF(EOF(ICR)) 2025,2024
      2024 CONTINUE
500   DO 2023 I=NFS1,NRN
      2023 W(I,1)=W(I,2)
      READ(ICR,*) (W(I,2),I=NFS1,NRN)
      WRITE(ILP,5050) TIME
      WRITE(ILP,*) (I,W(I,2),I=NFS1,NFSL)
505   5050 FORMAT(* RECHARGE AT TIME=*,F8.4)
      2025 CONTINUE
      IF(IFLGD.GE.0) GOTO 2034
      DO 2032 I=NFS1,NRB
      2032 W(I,1)=W(I,2)
510   2034 CONTINUE
C   ITERATE WITHIN TIME STEP
C   RESET BDY HEAD AT EACH TIME STEP IF REQUIRED
      IF(IFLGB.LE.1) GOTO 2028
```

```
      C     IF(ITIME.EQ.1) GOTO 2028
            DO 2027 I=1,IFLGB
 2027 PHI(NPSET(I))=PHISET
      C   IF HEAD RESET FOR A FREE SURFACE NODE, RESET Y CO-ORDINATE ALSO
            IF(NSETFS.NE.0) Y(NSETFS)=PHI(NSETFS)
            WRITE(ILP,5060) TIME,PHISET,IFLGB
 5060 FORMAT(* AT TIME = *,F12.6,*  HEAD RESET TO *,F12.5,*  FOR *,I5,
           1* NODES*)
 2028 CONTINUE
      C   SET UP INITIAL VALUES
            DO 8010 I=1,N
            PHIO(I)=PHI(I)
            PHINO(I)=PHIN(I)
 8010 PHINLO(I)=PHINL(I)
      C   SET PREVIOUS END POINTS FOR SHIFTING SEGMENTS
            IF(NSH.EQ.0) GOTO 208
            DO 203 I=1,NSH
            J=NESH(I)
            XP(I)=X(J)
  203 YP(I)=Y(J)
  208 CONTINUE
            ILOOP=0
            IF(ITIME.GT.1) GOTO 2030
            IF(NFS1.EQ.0)  GOTO 2030
            IF(IFLGC.NE.0) ILOOP=-1
            GOTO 2030
      C
      C   2ND ITERATION LOOP
      C
      C   SET FREE SURFACE AT HALF TIME STEP POSITION
 2070 THETA3=0.5
            IF(NFS.GT.NFSL) GOTO 8004
            DO 8002 I=NFS,NFSL
 8002 Y(I)=PHIO(I)+THETA3*(PHI(I)-PHIO(I))
            IF(NNSB.GE.NNSBP) GOTO 8004
      C   FALLING SEEPAGE FACE
            NFC3=NFS1+NNSBP-1
            DO 8110 I=NFS,NFC3
            IF(Y(I).GT.YM(I)) Y(I)=YM(I)
            IF(NTYP(I).EQ.7) NTYP(I)=11
            IF(NTYP(I).EQ.6) NTYP(I)=12
            IF(NTYP(I).EQ.9) NTYP(I)=13
            NNSB=NNSB+1
 8110 CONTINUE
 8004 CONTINUE
      C   SHIFT SHIFTING NODES TO NEW X,Y POSITIONS AND INTERPOLATE
      C   FOR NEW VALUES OF PHI AND PHIN
            IF(NSH.GT.0) CALL SHIFT
      C   RESTORE INITIAL PHI,PHIN,PHINL VALUES
            IF(ILOOP.EQ.2) GOTO 2030
            DO 8003 I=1,N
            PHI(I)=PHIO(I)
            PHIN(I)=PHINO(I)
            PHINL(I)=PHINLO(I)
 8003 CONTINUE
      C
      C   PERFORM BOUNDARY INTEGRATION AND SOLVE FOR UNKNOWNS
```

```
 2030 CONTINUE
      CALL INTG                                                    GMJ00800
      CALL ASSMBL                                                  GMJ00810
      CALL FDCOMP                                                  GMJ00820
      CALL FSOLVE                                                  GMJ00830
      CALL SURT                                                    GMJ00840
C
C   CHECK FOR HYDRAULIC CONNECTION
      IF(NFS1.EQ.0) GOTO 400
      IF(IFLGD.GE.0) GOTO 3800
      NNSBP=NNSB
      IUNSAT=0
      DO 3100 I=NFS1,NRB
      NTMP=NTYP(I)
      IF(PHI(I).LT.YM(I)) GOTO 3200
      Y(I)=YM(I)
      ZZ=H2N
      IF(YM(I).GT.H2N) ZZ=YM(I)
      W(I,2)=(ZZ-PHI(I))/B
      IF(NTMP.GE.11) GOTO 3000
      IF(NTMP.EQ.7) NTYP(I)=11
      IF(NTMP.EQ.6) NTYP(I)=12
      IF(NTMP.EQ.9) NTYP(I)=13
      NNSB=NNSB+1
      GOTO 3000
 3200 Y(I)=PHI(I)
      W(I,2)=(H2N-YM(I)+HC)/B
      YMHC=YM(I)-HC
      IF(PHI(I).LE.YMHC)  GOTO 3300
      W(I,2)=(H2N-PHI(I))/B
      IUNSAT=1
 3300 IF(YM(I).GT.H2N) W(I,2)=0.
      IF(NTMP.LE.10) GOTO 3000
      IF(NTMP.EQ.11) NTYP(I)=7
      IF(NTMP.EQ.12) NTYP(I)=6
      IF(NTMP.EQ.13) NTYP(I)=9
      NNSB=NNSB-1
 3000 CONTINUE
 3100 CONTINUE
      IF(NNSB.EQ.NNSBP) GOTO 3460
C   PREVENT MULTIPLE FREE SURFACE FORMATION WHEN H2N>YM, HC>0 AND
C   FREE SURFACE IS DECLINING THROUGH A FLAT BOTTOMMED STREAMBED.
      NRBM=NRB-1
      DO 3400 J=1,NRBM
      I=NRB-J
      IF(NTYP(I).LE.10 .AND. NTYP(I+1).GT.10) GOTO 3420
      GOTO 3400
 3420 WRITE(ILP,3440)
      WRITE(ILP,4100)
      WRITE(ILP,4300) I,X(I),Y(I),YM(I),PHI(I),W(I,2)
 3440 FORMAT(/* INSTABILITY CORRECTION APPLIED TO A STREAMBED NODE*,/)
      IF(NTYP(I).EQ.7) NTYP(I)=11
      IF(NTYP(I).EQ.6) NTYP(I)=12
      IF(NTYP(I).EQ.9) NTYP(I)=13
      NNSB=NNSB+1
      Y(I)=YM(I)
      PHI(I)=YM(I)
```

```
            W(I,2)=0.
            IF(H2N.GT.YM(I)) W(I,2)=(H2N-YM(I))/B
630        3400 CONTINUE
           3460 CONTINUE
            NFS=NFS1+NNSB
           3800 CONTINUE
          C   MOVE FREE SURFACE
635         IF(NFS.GT.NFSL) GOTO 400
            DO 202 I=NFS,NFSL
            Y(I)=PHI(I)
           202 CONTINUE
          C
640       C
          C   PRINT RESULTS
           400 WRITE(ILP,6700) ITIME,TIME,ILOOP
          6700 FORMAT(//* STEP NO *,I4,10X,*SOLUTION AT TIME =*,F13.6,
              1*   ILOOP =*,I3)
645         WRITE (ILP,6620)                                              GMJ00850
           6620 FORMAT (/1X,*POTENTIAL*)                                  GMJ00860
            WRITE (ILP,6630) (I,PHI(I),I=1,N)                             GMJ00870
           6630 FORMAT (5(1X,I4,1X,1PE13.6))                              GMJ00880
            WRITE (ILP,6640)                                              GMJ00890
650        6640 FORMAT (1X,*NORMAL DERIVATIVE OF THE POTENTIAL*)          GMJ00900
            NTNF=0                                                        GMJ00910
            DO 1690 KK=1,NSEG                                             GMJ00920
            NEP=NPSG(KK)                                                  GMJ00930
            MMT=NTNF+1                                                    GMJ00940
655         ITMP(1)=MMT                                                   GMJ00950
            TMP(1)=PHIN(MMT)                                              GMJ00960
            DO 1695 MM=2,NEP                                              GMJ00970
            MMT=NTNF+MM                                                   GMJ00980
            IF (MMT.EQ.NP) MMT=1                                          GMJ00990
660         ITMP(MM)=MMT                                                  GMJ01000
            TMP(MM)=PHIN(MMT)                                             GMJ01010
           1695 CONTINUE                                                  GMJ01020
            WRITE (ILP,6650) KK,(ITMP(I),TMP(I),I=1,NEP)                  GMJ01030
           6650 FORMAT (1X,I4,/,5(1X,I4,1X,1PE13.6))                      GMJ01040
665         NTNF=NTNF+NEP-1                                               GMJ01050
           1690 CONTINUE                                                  GMJ01060
            IF(NFS1.EQ.0) GOTO 300
            IF(NFS.GT.NFSL) GOTO 3500
            WRITE(ILP,1710)
670        1710 FORMAT(//,*         FREE SURFACE POSITIONS*)
            WRITE(ILP,6630)(I,Y(I),I=NFS,NFSL)
           3500 IF(IFLGD.GE.0) GOTO 3900
            WRITE(ILP,3600) NNSB
           3600 FORMAT(/,* NO OF STREAMBED NODES = *,I4)
675         WRITE(ILP,4100)
            DO 3700 I=NFS1,NRB
           3700 WRITE(ILP,4300) I,X(I),Y(I),YM(I),PHI(I),W(I,2)
           3900 CONTINUE
          C   END OF 2ND ITERATION LOOP
680       C
          C   ILOOP =-1 ..ITIME=1,IFLGC=1,1ST ITERATION,SETS THETA=1 IN SORT
          C         = 0 ..1ST ITERATION AT ALL OTHER TIMES
          C         = 1 ..2ND ITERATION (Y(K+1)-Y(K))>THRESH
          C         = 2 ..ITIME=1,IFLGC=1,2ND ITERATION,SETS DLTT=0 IN ASSMBL
```

```
685         C                AND SORT
                IF(ILOOP.GE.1) GOTO 300
                IF(NFS1.EQ.0)  GOTO 300
                IF(NFS.GT.NFSL) GOTO 300
                IF(ILOOP.EQ.0) GOTO 200
690             IF(IFLGA.GE.2) CALL CHECK
            C   ILOOP=-1   INITIAL PHIN AND PHINL NOW USED IN MASS BALANCE
                ILOOP=2
                GOTO 2070
              200 CONTINUE
695         C
            C   TEST FREE SURFACE MOVEMENT TO SEE IF 2ND ITERATION IS REQUIRED
                DO 8000 I=NFS1,NFSL
                IF(PHI(I).EQ.0.) GOTO 8001
                TEST=ABS((PHI(I)-PHIO(I))/PHI(I))
700             IF(TEST.GT.THRESH) GOTO 8001
             8000 CONTINUE
                IF(NNSB.NE.NNSBP) GOTO 8001
                IF(IUNSAT.EQ.1) GOTO 8001
                GOTO 300
705          8001 ILOOP=1
                GOTO 2070
              300 CONTINUE
            C
            C   CALC INTERNAL SOLUTIONS AND MASS BALANCE THEN INCREMENT TIME
710             IF(INTNL.EQ.0) GOTO 6675
                WRITE(ILP,7696)
             7696 FORMAT(//*        INTERNAL SOLUTIONS*/*     X-COORD*,*     Y-COORD*,
                1* POTENTIAL     X-VEL     Y-VEL*/)
            C       DO 6670 I=1,INTNL
715         C6670 CALL INTRNL(XINT(I),YINT(I))
             6675 IF(IFLGA.GE.2) CALL CHECK
                IF(IFLGA.LE.2 .AND. ICAL.LT.2) GOTO 9900
                IF(ITIME.GT.NHIS) GOTO 9900
                TM(ITIME)=TIME
720             IF(NBOR.EQ.0) GOTO 9905                                          CAL
            C   CALCULATE FREE SURFACE ELEVATION AT OBSERVATION BORES            CAL
                DO 9920 I=1,NBOR                                                 CAL
                J=NDBJ(I)                                                        CAL
             9920 P(I+3,ITIME)=PHI(J)+CINT(I)*(PHI(J+1)-PHI(J))                  CAL
725             GOTO 9900                                                       CAL
             9905 CONTINUE                                                      CAL
                DO 9910 I=1,IFLGA
             9910 P(I+3,ITIME)=PHI(NH(I))
             9900 CONTINUE
730             IF(TIME.GE.TMLMT) GOTO 9000
                IF(ITIME.EQ.NHIS) GOTO 9000
                IF(TFACT.GE.11.) GOTO 7200
                IF(TFACT.GT.9.9) GOTO 6800
                GOTO 7100
735          7200 IF(ITIME.EQ.NTHIME) GOTO 9000
                ITIME=ITIME+1
                TIME=THIME(ITIME)
                DLTT=TIME-THIME(ITIME-1)
                GOTO 2020
740          7100 CONTINUE
                DLTT=DLTT*TFACT
```

```
            TIME=TIME+DLTT
            ITIME=ITIME+1
            GOTO 2020
C   BASE 10 LOGARITHMIC TIME STEPS (MULTIPLES OF 2,5,10 OF
C   INITIAL DLTT)
 6800 ITF=ITIME/3.
            IT=ITIME-ITF*3-1
            IF(IT) 6810,6810,6820
 6810 TT=2.*TIME
            GOTO 6900
 6820 TT=2.5*TIME
 6900 DLTT=TT-TIME
            TIME=TT
            ITIME=ITIME+1
            GOTO 2020
C   END OF TIME LOOP
 9000 CONTINUE
            IF(NBOR.NE.0) GOTO 9400                                    CAL
            IF(IFLGA.LE.2) STOP
            NHCOL=IFLGA+3
            WRITE(ILP,9100) (NH(I),I=1,IFLGA)
 9100 FORMAT(1X,///,* HISTORY OF POTENTIALS AT SELECTED NODES*,/
           1*   TIME   *,16X,12(I4,4X))
            IF(ITIME.GT.NHIS) ITIME=NHIS
            GOTO 9500                                                  CAL
 9400 NHCOL=NBOR+3                                                     CAL
            WRITE(ILP,9450) (NB(I),I=1,NBOR)                           CAL
 9450 FORMAT(///,* HISTORY OF POTENTIALS AT SELECTED BORES*,/          CAL
           1*   TIME   NET INFLUX  % ERROR CON.NO.*,12I8)              CAL
            WRITE(ILP,9460) (SL(I),I=1,NBOR)                           CAL
 9460 FORMAT(37X,12F8.1)                                               CAL
 9500 CONTINUE                                                         CAL
            DO 9200 J=1,ITIME
 9200 WRITE(ILP,9300) TM(J),(P(I,J),I=1,NHCOL)
 9300 FORMAT(1X,F12.4,12F8.3)
            IF(ICAL.EQ.2) CALL MATCH(ITIME)                            CAL
            IF(ICAL.EQ.2) CALL COMPARE                                 CAL
            IF(ICAL.GE.3) CALL BOREFIT(ITIME,1)                        CAL
            IF(ICAL.NE.3) GOTO 9700                                    CAL
            TRT=TRATIO                                                 CAL
            FAC=-0.6                                                   CAL
            DO 9750 I=1,6                                              CAL
            FAC=FAC+0.2                                                CAL
            IF(I.EQ.3 .OR. I.EQ.5) FAC=FAC-0.1                         CAL
            TRATIO=TRT+TRT*FAC                                         CAL
            WRITE(ILP,9760) TRATIO,FAC                                 CAL
 9760 FORMAT(//* TIME RATIO , FACTOR *,2F10.3,/)                       CAL
            REWIND 8                                                   CAL
            CALL BOREFIT(ITIME,0)                                      CAL
 9750 CONTINUE                                                         CAL
 9700 CONTINUE                                                         CAL
            IF(ICAL.GE.4) CALL FLOWFIT(ITIME,ICAL,PERM)                CAL
            STOP
            END                                                      GMJ01080
```

```
      C
      C
      C-----------------------------------------------------------------
            SUBROUTINE CNRANG                                          GMJ01090
      C
      C   CALCULATE CORNER ANGLES FOR BOUNDARY NODES
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                              GMJ01110
           $     IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,         GMJ00090
           $     NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                     GMJ01140
           $     CORANG(44),FCORN(44)                                  GMJ01150
            DO 1010 II=1,N                                             GMJ01160
            IIM=II-1                                                   GMJ01170
            IF (IIM.EQ.0) IIM=N                                        GMJ01180
            IIP=II+1                                                   GMJ01190
            IF (IIP.EQ.NP) IIP=1                                       GMJ01200
            XII=X(II)                                                  GMJ01210
            YII=Y(II)                                                  GMJ01220
            XV1=X(IIM)-XII                                             GMJ01230
            YV1=Y(IIM)-YII                                             GMJ01240
            XV2=X(IIP)-XII                                             GMJ01250
            YV2=Y(IIP)-YII                                             GMJ01260
            XYMG= SQRT(((XV1*XV1)+(YV1*YV1))*((XV2*XV2)+(YV2*YV2)))    GMJ01270
            CTMP=((XV1*XV2)+(YV1*YV2))/XYMG                            GMJ01280
            STMP=((XV1*YV2)-(XV2*YV1))/XYMG                            GMJ01290
            ETMP= ATAN2(STMP,CTMP)                                     GMJ01300
            IF (ETMP.LT.0.0) ETMP=ETMP+6.283185                        GMJ01310
            CORANG(II)=ETMP                                            GMJ01320
      1010  CONTINUE                                                   GMJ01330
            IF (IFLGB.NE.1) GO TO 9999                                 GMJ01340
            WRITE (ILP,6010)                                           GMJ01350
      6010  FORMAT (/1X,*NODAL ANGLES*)                                GMJ01360
            WRITE (ILP,*) (I,CORANG(I),I=1,N)                          GMJ01370
      9999  RETURN                                                     GMJ01380
            END                                                        GMJ01390
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE ASSMBL                                          GMJ01400
      C
      C     ASSEMBLE MATRIX RN (N*N) AND VECTOR RHS (N*1)
      C     WHERE     RN * U = RHS
      C     AND  U (N*1) IS THE VECTOR OF UNKNOWN PHI,PHIN AND PHINL VALUES
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                              GMJ01420
           $      IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,       GMJ00090
           $      NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                     GMJ01460
           $      CORANG(44),FCORN(44)                                GMJ01470
            COMMON /SOL/ RN(44,44),RHS(44),IPVT(44)                    GMJ01480
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
            COMMON /BED/ WONE,B,H2,H2N,YM(22)
            REAL PHINT(66)                                             GMJ01490
            THETA=0.5
      C     CALCULATE FREE SURFACE ANGLES
            DO 75 I=1,N
            NTMP=NTYP(I)
            IF(NTMP.GE.6 .AND. NTMP.LE.10) GOTO 80
         75 CONTINUE
            GOTO 85
         80 CALL BETA
         85 CONTINUE
            DO 1100 JJ=1,N                                             GMJ01500
            NTMP=NTYP(JJ)                                              GMJ01510
            IF ((NTMP.EQ.1).OR.(NTMP.EQ.5)) GO TO 2010                 GMJ01520
            DO 1020 II=1,N                                             GMJ01530
            TMPA=RN(II,JJ)                                             GMJ01540
            RN(II,JJ)=-RLN(II,JJ)                                      GMJ01550
            RLN(II,JJ)=-TMPA                                           GMJ01560
            IF(NTMP.GE.11) GOTO 1040
            IF(NTMP.NE.6 .AND. NTMP.NE.7 .AND. NTMP.NE.9) GOTO 1020
            IF(ILOOP.EQ.2) GOTO 1020
            DTHC=DLTT*THETA2/CBTA(JJ)
            RN(II,JJ)=RN(II,JJ)+DTHC*RLN(II,JJ)
            GOTO 1020
      1040  RN(II,JJ)=RN(II,JJ)+B*RLN(II,JJ)
      1020  CONTINUE                                                   GMJ01570
            PHINT(JJ)=PHI(JJ)                                          GMJ01580
            GO TO 1100                                                 GMJ01590
      2010  PHINT(JJ)=PHIN(JJ)                                         GMJ01600
            IF (NTMP.EQ.5) PHINT(JJ)=0.0                               GMJ01610
      1100  CONTINUE                                                   GMJ01620
            DO 1440 IN=1,N                                             GMJ01630
            ACC=FCORN(IN)                                              GMJ01640
            DO 1450 KN=1,N                                             GMJ01650
            NTMP=NTYP(KN)
            IF(NTMP.GE.11) GOTO 1442
            IF(NTMP.GE.6) GOTO 1430
            ACC=ACC+(RLN(IN,KN)*PHINT(KN))                             GMJ01660
            GOTO 1450
      1430  ACC=ACC+RLN(IN,KN)*PHINT(KN)
```

```
      IF(NTMP.EQ.8 .OR.NTMP.EQ.10)GOTO 1445
      IF(ILOOP.EQ.2) GOTO 1445
60    WTERM=(THETA*W(KN,2)+(1.-THETA)*W(KN,1))
      ACC=ACC+RLN(IN,KN)*DLTT*WTERM
      DTH=DLTT*(1.-THETA2)/CBTA(KN)
      TMPB=PHIN(KN)
      IF(NTMP.EQ.7) TMPB=PHINL(KN)
65    ACC=ACC-RLN(IN,KN)*DTH*TMPB
      GOTO 1445
 1442 ZZ=H2N
      IF(YM(KN).GT.H2N) ZZ=YM(KN)
      ACC=ACC+RLN(IN,KN)*ZZ
70    1445 CONTINUE
      1450 CONTINUE                                                    GMJ01670
      RHS(IN)=ACC                                                      GMJ01680
      1440 CONTINUE                                                    GMJ01690
      IF (IFLGB.NE.1) GO TO 9999                                       GMJ01700
75    WRITE (ILP,6490)                                                 GMJ01710
      6490 FORMAT (/1X,*:RN-MOD:  FROM ASSMBL*)                        GMJ01720
      DO 1660 II=1,N                                                   GMJ01730
      WRITE (ILP,*) (RN(II,J),J=1,N)                                   GMJ01740
      1660 CONTINUE                                                    GMJ01750
80    WRITE (ILP,6020)                                                 GMJ01760
      6020 FORMAT (/1X,*:PHINT:  FROM ASSMBL*)                         GMJ01770
      WRITE (ILP,*) (PHINT(I),I=1,N)                                   GMJ01780
      WRITE (ILP,6570)                                                 GMJ01790
      6570 FORMAT (/1X,*:RIGHT-HAND-SIDE:  FROM ASSMBL*)               GMJ01800
85    WRITE (ILP,*) (RHS(I),I=1,N)                                     GMJ01810
      9999 RETURN                                                      GMJ01820
      END                                                              GMJ01830
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE SORT                                                   GMJ01840
      C
      C     SORTS OUT EACH UNKNOWN AS EITHER A PHI, PHIN OR PHINL
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                     GMJ01860
           $      IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,               GMJ00090
           $      NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                            GMJ01890
           $      CORANG(44),FCORN(44)                                       GMJ01900
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /SOL/ RN(44,44),RHS(44),IPVT(44)                           GMJ01920
            COMMON /BED/ WONE,B,H2,H2N,YM(22)
            THE=0.5
            THETA=THETA2
      C     FULLY IMPLICIT SOLUTION FOR FREE SURFACE AT TIME=0
            IF(ILOOP.GE.0) GOTO 20
            THETA=1.
         20 CONTINUE
            DO 1000 JJ=1,N                                                    GMJ01930
            ATMP=RHS(JJ)                                                      GMJ01940
            NTMP=NTYP(JJ)                                                     GMJ01950
            IF(NTMP.EQ.12) PHINL(JJ)=ATMP
            IF ((NTMP.EQ.1).OR.(NTMP.EQ.5)) PHI(JJ)=ATMP                      GMJ01960
            IF ((NTMP.EQ.2).OR.(NTMP.EQ.4)) PHIN(JJ)=ATMP                     GMJ01970
            IF (NTMP.EQ.3 .OR. NTMP.EQ.6) PHINL(JJ)=ATMP
            IF ((NTMP.EQ.1).OR.(NTMP.EQ.2)) PHINL(JJ)=PHIN(JJ)                GMJ01990
            IF(NTMP.LT.6) GOTO 1000
            IF(NTMP.EQ.8) GOTO 100
            IF(NTMP.EQ.10) GOTO 200
            IF(NTMP.EQ.11) GOTO 2000
            IF(ILOOP.EQ.2) GOTO 50
      C     ILOOP=2 FREEZES FREE SURFACE FOR 2ND ITERATION AT TIME=0
            TMP=PHIN(JJ)
            IF(NTMP.EQ.7) TMP=PHINL(JJ)
            DTC=DLTT/CBTA(JJ)
            WTERM=DLTT*(THE*W(JJ,2)+(1.-THE)*W(JJ,1))
            PHI(JJ)=PHI(JJ)-DTC*(THETA*ATMP+(1.-THETA)*TMP)+WTERM
         50 IF(NTMP.EQ.7) GOTO 200
        100 PHIN(JJ)=ATMP
            GOTO 1000
        200 PHINL(JJ)=ATMP
            GOTO 1000
       2000 CONTINUE
            TMP=PHIN(JJ)
            IF(NTMP.EQ.11) TMP=PHINL(JJ)
      C     ADJUST STREAMBED POTENTIALS
            ZZ=H2N
            IF(YM(JJ).GT.H2N) ZZ=YM(JJ)
            PHI(JJ)=ZZ-B*ATMP
            IF(NTMP.EQ.11) GOTO 1050
            PHIN(JJ)=ATMP
            GOTO 1000
       1050 PHINL(JJ)=ATMP
       1000 CONTINUE
```

```
 9999 RETURN                                                    GMJ02010
      END                                                       GMJ02020
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE INTG                                                   GMJ02030
      C
      C    INTEGRATES AROUND BOUNDARY TO PRODUCE MATRICES RN AND RLN
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                     GMJ02050
           $      IFLGA,IFLGB,IFLGC,THETA2,GRAV,OTMP,TIME,DLTT,               GMJ00090
           $      NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                            GMJ02080
           $      CORANG(44),FCORN(44)                                       GMJ02090
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /SOL/ RN(44,44),RHS(44),IPVT(44)                           GMJ02110
            DO 1130 KL=1,N                                                    GMJ02120
            DO 1140 KK=1,N                                                    GMJ02130
            RN(KK,KL)=0.0                                                     GMJ02140
            RLN(KK,KL)=0.0                                                    GMJ02150
       1140 CONTINUE                                                          GMJ02160
       1130 CONTINUE                                                          GMJ02170
            DO 1010 II=1,N                                                    GMJ02180
            ACFB=0.0                                                          GMJ02190
            DO 1020 JJ=1,N                                                    GMJ02200
            JJP=JJ+1                                                          GMJ02210
            IF (JJP.EQ.NP) JJP=1                                              GMJ02220
            XJPJ=X(JJP)-X(JJ)                                                 GMJ02230
            YJPJ=Y(JJP)-Y(JJ)                                                 GMJ02240
            XJI=X(JJ)-X(II)                                                   GMJ02250
            YJI=Y(JJ)-Y(II)                                                   GMJ02260
            XJPI=X(JJP)-X(II)                                                 GMJ02270
            YJPI=Y(JJP)-Y(II)                                                 GMJ02280
            XIA=0.0                                                           GMJ02290
            XIB=0.0                                                           GMJ02300
            ETAL=0.0                                                          GMJ02310
            ATN=0.0                                                           GMJ02320
            BTN=0.0                                                           GMJ02330
            ASQ=0.0                                                           GMJ02340
            BSQ=0.0                                                           GMJ02350
            ALN=0.0                                                           GMJ02360
            BLN=0.0                                                           GMJ02370
            R1= SQRT((XJPJ*XJPJ)+(YJPJ*YJPJ))                                 GMJ02380
            IF (II.NE.JJ) GO TO 2020                                          GMJ02390
            XIB= SQRT((XJPI*XJPI)+(YJPI*YJPI))                                GMJ02400
            BSQ=XIB*XIB                                                       GMJ02410
            BLN=ALOG(BSQ)                                                     GMJ02420
            GO TO 2040                                                        GMJ02430
       2020 IF (II.NE.JJP) GO TO 2030                                         GMJ02440
            XIA=- SQRT((XJI*XJI)+(YJI*YJI))                                   GMJ02450
            ASQ=XIA*XIA                                                       GMJ02460
            ALN=ALOG(ASQ)                                                     GMJ02470
            GO TO 2040                                                        GMJ02480
       2030 CO=XJPJ/R1                                                        GMJ02490
            SI=YJPJ/R1                                                        GMJ02500
            XIA=(YJI*SI)+(XJI*CO)                                             GMJ02510
            XIB=(YJPI*SI)+(XJPI*CO)                                           GMJ02520
            ETAL= ABS((XJI*SI)-(YJI*CO))                                      GMJ02530
            IF (ETAL.LT.0.00001) GO TO 2010                                   GMJ02540
```

```
                ATN= ATAN(XIA/ETAL)                                              GMJ02550
                BTN= ATAN(XIB/ETAL)                                              GMJ02560
60          2010 ASQ=(XIA*XIA)+(ETAL*ETAL)                                       GMJ02570
                BSQ=(XIB*XIB)+(ETAL*ETAL)                                        GMJ02580
                ALN=ALOG(ASQ)                                                    GMJ02590
                BLN=ALOG(BSQ)                                                    GMJ02600
            2040 XIBMA=XIB-XIA                                                   GMJ02610
65              SIGNRN=-(XJI*YJPJ)+(XJPJ*YJI)                                    GMJ02620
                TMPA=(0.5*ETAL*(BLN-ALN))/XIBMA                                  GMJ02630
                TMPB=(BTN-ATN)/XIBMA                                             GMJ02640
                PJ=(XIB*TMPB)-TMPA                                               GMJ02650
                PJ= SIGN(PJ,SIGNRN)                                              GMJ02660
70              PJP=TMPA-(XIA*TMPB)                                              GMJ02670
                PJP= SIGN(PJP,SIGNRN)                                            GMJ02680
                ONE=(BSQ*(BLN-1.0))-(ASQ*(ALN-1.0))                              GMJ02690
                TWO=(XIB*BLN)-(XIA*ALN)-(2.0*XIBMA)+(2.0*ETAL*(BTN-ATN))         GMJ02700
                PNJ=((XIB*TWO)-(ONE/2.0))/(2.0*XIBMA)                            GMJ02710
75              PNJP=((ONE/2.0)-(XIA*TWO))/(2.0*XIBMA)                           GMJ02720
                RN(II,JJ)=RN(II,JJ)+PJ                                           GMJ02730
                RN(II,JJP)=RN(II,JJP)+PJP                                        GMJ02740
                MTMP=NTYP(JJ)                                                    GMJ02750
                NTMP=NTYP(JJP)                                                   GMJ02760
80              IF (MTMP.EQ.5) GO TO 2540                                        GMJ02770
                IF (NTMP.EQ.5) GO TO 2630                                        GMJ02780
                IF(MTMP.EQ.13) GOTO 2540
                IF (MTMP.EQ.4 .OR. MTMP.EQ.8 .OR. MTMP.EQ.9) GO TO 2540
                IF (NTMP.EQ.3 .OR. NTMP.EQ.7 .OR. NTMP.EQ.10) GO TO 2630
85              IF(NTMP.EQ.11) GOTO 2630
                RLN(II,JJ)=RLN(II,JJ)+PNJ                                        GMJ02810
                RLN(II,JJP)=RLN(II,JJP)+PNJP                                     GMJ02820
                GO TO 1020                                                       GMJ02830
            2540 ACFB=ACFB+(PHINL(JJ)*PNJ)                                       GMJ02840
90              RLN(II,JJP)=RLN(II,JJP)+PNJP                                     GMJ02850
                GO TO 1020                                                       GMJ02860
            2630 RLN(II,JJ)=RLN(II,JJ)+PNJ                                       GMJ02870
                ACFB=ACFB+(PHIN(JJP)*PNJP)                                       GMJ02880
            1020 CONTINUE                                                        GMJ02890
95              FCORN(II)=ACFB                                                   GMJ02900
            1010 CONTINUE                                                        GMJ02910
                CALL CNRANG                                                      GMJ02920
                DO 1310 II=1,N                                                   GMJ02930
                RN(II,II)=RN(II,II)-CORANG(II)                                   GMJ02940
100         1310 CONTINUE                                                        GMJ02950
                IF (IFLGB.NE.1) GO TO 9999                                       GMJ02960
                WRITE (ILP,6110)                                                 GMJ02970
            6110 FORMAT (/1X,*:RN:  FROM INTG*)                                  GMJ02980
                DO 1810 I=1,N                                                    GMJ02990
105             WRITE (ILP,*) (RN(I,J),J=1,N)                                    GMJ03000
            1810 CONTINUE                                                        GMJ03010
                WRITE (ILP,6120)                                                 GMJ03020
            6120 FORMAT (/1X,*:RLN:  FROM INTG*)                                 GMJ03030
                DO 1820 I=1,N                                                    GMJ03040
110             WRITE (ILP,*) (RLN(I,J),J=1,N)                                   GMJ03050
            1820 CONTINUE                                                        GMJ03060
                WRITE (ILP,6130)                                                 GMJ03070
            6130 FORMAT (/1X,*:FCORN:  FROM INTG*)                               GMJ03080
                WRITE (ILP,*) (I,FCORN(I),I=1,N)                                 GMJ03090
```

```
115          9999 RETURN                                                    GMJ03100
                  END                                                       GMJ03110
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE FDCOMP                                              GMJ03120
      C
      C     REARRANGE MATRIX RN READY FOR SOLVING
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                  GMJ03140
           $      IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,             GMJ00090
           $      NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /SOL/ A(44,44),RHS(44),IPVT(44)                         GMJ03170
            DO 35 K=1,NM                                                   GMJ03180
            KP1=K+1                                                        GMJ03190
            M=K                                                            GMJ03200
            AAMAX= ABS(A(K,K))                                             GMJ03210
            DO 15 I=KP1,N                                                  GMJ03220
            IF ( ABS(A(I,K)).LT.AAMAX) GO TO 15                            GMJ03230
            M=I                                                            GMJ03240
            AAMAX= ABS(A(I,K))                                             GMJ03250
         15 CONTINUE                                                       GMJ03260
            IPVT(K)=M                                                      GMJ03270
            T=A(M,K)                                                       GMJ03280
            A(M,K)=A(K,K)                                                  GMJ03290
            A(K,K)=T                                                       GMJ03300
            DO 20 I=KP1,N                                                  GMJ03310
            A(I,K)=-A(I,K)/T                                               GMJ03320
         20 CONTINUE                                                       GMJ03330
            DO 30 J=KP1,N                                                  GMJ03340
            T=A(M,J)                                                       GMJ03350
            A(M,J)=A(K,J)                                                  GMJ03360
            A(K,J)=T                                                       GMJ03370
            DO 25 I=KP1,N                                                  GMJ03380
            A(I,J)=A(I,J)+(A(I,K)*T)                                       GMJ03390
         25 CONTINUE                                                       GMJ03400
         30 CONTINUE                                                       GMJ03410
         35 CONTINUE                                                       GMJ03420
            RETURN                                                         GMJ03430
            END                                                            GMJ03440
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE FSOLVE                                                   GMJ03450
      C
      C     SOLVE SIMULTANEOUS LINEAR EQUATIONS
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                       GMJ03470
           $     IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,                  GMJ00090
           $     NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /SOL/ A(44,44),RHS(44),IPVT(44)                              GMJ03500
            DO 20 K=1,NM                                                        GMJ03510
            KP1=K+1                                                             GMJ03520
            M=IPVT(K)                                                           GMJ03530
            T=RHS(M)                                                            GMJ03540
            RHS(M)=RHS(K)                                                       GMJ03550
            RHS(K)=T                                                            GMJ03560
            DO 10 I=KP1,N                                                       GMJ03570
            RHS(I)=RHS(I)+(A(I,K)*T)                                            GMJ03580
         10 CONTINUE                                                            GMJ03590
         20 CONTINUE                                                            GMJ03600
            DO 40 KB=1,NM                                                       GMJ03610
            KM1=N-KB                                                            GMJ03620
            K=KM1+1                                                             GMJ03630
            RHS(K)=RHS(K)/A(K,K)                                                GMJ03640
            T=-RHS(K)                                                           GMJ03650
            DO 30 I=1,KM1                                                       GMJ03660
            RHS(I)=RHS(I)+(A(I,K)*T)                                            GMJ03670
         30 CONTINUE                                                            GMJ03680
         40 CONTINUE                                                            GMJ03690
            RHS(1)=RHS(1)/A(1,1)                                                GMJ03700
            RETURN                                                              GMJ03710
            END                                                                 GMJ03720
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE BETA
      C
      C   CALCULATE SLOPE OF FREE SURFACE
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                  GMJ03470
           $     IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,             GMJ00090
           $     NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),CORANG(44),FCORN(44)
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
      C  CALCULATE COSINE OF ANGLE OF SLOPE OF FREE SURFACE
            NFP=NFS1+1
            NLM=NFSL-1
            BETA1=ATAN((Y(NFP)-Y(NFS1))/(X(NFP)-X(NFS1)))
            CBTA(NFS1)=COS(BETA1)
            BETAL=ATAN((Y(NFSL)-Y(NLM))/(X(NFSL)-X(NLM)))
            CBTA(NFSL)=COS(BETAL)
      C   INTERMEDIATE POINTS
            DO 140 I=NFP,NLM
            YO=Y(I-1)-Y(I)
            XO=X(I-1)-X(I)
            Y1=Y(I+1)-Y(I)
            X1=X(I+1)-X(I)
            D=XO*X1*(X1-XO)
            B=ATAN((YO*X1**2-Y1*XO**2)/D)
            CBTA(I)=COS(B)
            IF(CBTA(I).LT.0.1) WRITE(6,150) I,B,CBTA(I)
          140 CONTINUE
          150 FORMAT(*  WARNING - COS BETA SMALL - *I,BETA,COS(BETA)*,
           $ I5,2F10.6)
            IF(IFLGB.NE.1) RETURN
            WRITE(ILP,200)
            WRITE(ILP,*) (I,CBTA(I),I=NFS1,NFSL)
          200 FORMAT(1X,/,* COSBETA FOR FREE SURFACE NODES*)
            RETURN
            END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE CHECK
      C
      C   CHECK RESULTS FOR CONSERVATION OF MASS
      C
            COMMON /HIS/ NHIS,P(15,65),TM(65),NH(15)
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                  GMJ00080
           $     IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,             GMJ00090
           $     NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                         GMJ00120
           $     CORANG(44),FCORN(44)                                      GMJ00130
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
            COMMON /BED/ WONE,B,H2,H2N,YM(22)
      C   CALCULATE MASS BALANCE AT EACH TIME STEP
            IF(ILOOP.LT.0) GOTO 450
            IF(ITIME.GT.0) GOTO 50
            TQI=TQO=TDW=TDE=TQS=0.
            IFW1=0
            IF(IFLGC.GE.1) IFW1=1
         50 CONTINUE
      C   VOLUME UNDER FREE SURFACE
            E=DW=0.
            IF(NFS1.EQ.0) GOTO 300
            NSP=NFS1+1
            NLM=NFSL-1
            E1=Y(NFS1)*0.5*(ABS(X(NSP)-X(NFS1)))
            EL=Y(NFSL)*0.5*(ABS(X(NFSL)-X(NLM)))
            DO 100 I=NSP,NLM
        100 E=E+Y(I)*0.5*(ABS(X(I+1)-X(I-1)))
            E=E+E1+EL
      C   RECHARGE TO FREE SURFACE
            IF(IFLGD.EQ.0) GOTO 300
            J=2
            NFSP=NFS+1
            IF(ITIME.EQ.0) J=1
            DW1=W(NFS,J)*0.5*(ABS(X(NFSP)-X(NFS)))
            DWL=W(NFSL,J)*0.5*(ABS(X(NFSL)-X(NLM)))
            DO 200 I=NFSP,NLM
        200 DW=DW+W(I,J)*0.5*(ABS(X(I+1)-X(I-1)))
            DW=DW+DW1+DWL
        300 IF(ITIME.NE.0) GOTO 400
            EP=E
            DWP=DW
            RETURN
        400 CONTINUE
            IF(IFW1.NE.1) GOTO 450
            DWP=DW
            IFW1=0
        450 CONTINUE
      C   FLUX ACROSS OTHER BOUNDARIES
            QI=QO=QS=QR=0.
            NFSM=NFS-1
            DO 600 I=1,N
            IF(I.GE.NFS .AND. I.LT.NFSL) GOTO 580
```

```
                IF(I.EQ.N) GOTO 510
                AL=SQRT((X(I+1)-X(I))**2+(Y(I+1)-Y(I))**2)
 60             QD=AL*0.5*(PHINL(I)+PHIN(I+1))
                IF(I.EQ.NFSM .AND. IFLGD.LT.0) QD=AL*0.5*(PHINL(I)+W(NFS,2))
                IF(I.GE.NFS1 .AND. I.LE.NFSM) QR=QR+QD
              520 IF(QD.LT.0.) GOTO 550
                QI=QI+QD
 65             GOTO 580
              550 QD=QD-QD
                GOTO 580
            C IF I=N
              510 AL=SQRT((X(1)-X(N))**2+(Y(1)-Y(N))**2)
 70             QD=AL*0.5*(PHINL(N)+PHIN(1))
                GOTO 520
              580 CONTINUE
              600 CONTINUE
            C   CALCULATE NON DIMENSIONAL STREAM DISCHARGE RATE (QS)
 75             IF(IFLGD.GE.0) GOTO 460
                IF(H2N.LE.YM(NFS1)) GOTO 460
                WB=2.
                A=(H2N-YM(NFS1))/WB
                C1=(A+A*A)**(5./3.)
 80             C2=(1.+2.82*A)**(-2./3.)
                QS=C1*C2
              460 CONTINUE
            C   UNIT BASE WIDTH AND 45 DEGREE BANK SLOPES ASSUMED ABOVE
                IF(ITIME.NE.1) GOTO 700
 85             QIP=QI
                QOP=QO
                QSP=QS
                IF(ILOOP.LT.0) RETURN
              700 CONTINUE
 90         C   FLUX DURING PERIOD
                DQI=0.5*DLTT*(QIP+QI)
                DQO=0.5*DLTT*(QOP+QO)
                DDW=0.5*DLTT*(DWP+DW)
                DQS=0.5*DLTT*(QSP+QS)
 95             DE=E-EP
                DLTS=DQI-DQO+DDW
                DER=DE-DLTS
                IF(DE.NE.0.) PDER=DER*100./DE
            C   TOTAL FLUX SINCE START
100             TQI=TQI+DQI
                TQO=TQO+DQO
                TDW=TDW+DDW
                TQS=TQS+DQS
                TDE=TDE+DE
105             TLTS=TQI-TQO+TDW
                TER=TDE-TLTS
                IF(TDE.NE.0.) PTER=TER*100./TDE
            C   SAVE PRESENT FLUX AS PREVIOUS FLUX
                WRITE(ILP,740)
110           740 FORMAT(1X,/,* DISCHARGE RATES ACCROSS BOUNDARIES AT END OF *,
                 1*PERIOD*,/,* RECHARGE     INFLOW     OUTFLOW*)
                WRITE(ILP,760) DW,QI,QO
              760 FORMAT(1X,3F11.6)
            C   COMPARE GROUNDWATER ACCRETION WITH STREAMFLOW
```

```
115                IF(IFLGD.GE.0) GOTO 810
                   RINST=DW+OR
                   RSINST=RINST/QS
                   RSTOT=TLTS/TQS
                   WRITE(ILP,820)
120              820 FORMAT(1X,/,* GW INTAKE AND STREAMFLOW RATE AT END*,
                    1* OF PERIOD AND TOTAL SINCE START*,/,
                    2* INTAKE(I)  STREAM(I)   IN/S(I)   INTAKE*,
                    3*(T)  STREAM(T)   IN/S(T)*)
                   WRITE(ILP,830) RINST,QS,RSINST,TLTS,TQS,RSTOT
125              830 FORMAT(1X,6F11.6)
                 810 CONTINUE
               C
               C   PRINT TIME HISTORY FOR NON-DIM DISCHARGE AND BANK STORAGE
                   IF(ITIME.GT.NHIS) GOTO 850
130                P(1,ITIME)=RINST
                   P(2,ITIME)=PTER
                   P(3,ITIME)=NFSM
                 850 CONTINUE
                   QIP=QI
135                QOP=QO
                   DWP=DW
                   QSP=QS
                   EP=E
                   WRITE(ILP,800)
140              800 FORMAT(1X,/,* CHECK MASS BALANCE (PERIOD//SINCE START)*,/,
                    1 * RECHARGE  +  INFLOW  -  OUTFLOW = (D.STO) .CF. *,
                    2 *D.PHREAT    ERROR    % ERROR*)
                   WRITE(ILP,900) DDW,DQI,DQO,DLTS,DE,DER,PDER
                   WRITE(ILP,900) TDW,TQI,TQO,TLTS,TDE,TER,PTER
145              900 FORMAT(1X,7F11.6)
                   WRITE(ILP,1000)
                1000 FORMAT(* ----------------------------------------------*,
                    1*----------------------------*,/)
                   RETURN
150                END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE SHIFT
      C
      C   SHIFT NODES ON SPECIFIED MOVING BOUNDARIES
      C
            COMMON /BLK/ KOR,KOW,ICR,ILP,                                  GMJ00080
           $     IFLGA,IFLGB,IFLGC,THETA2,GRAV,DTMP,TIME,DLTT,             GMJ00090
           $     NM,N,NP,NTYP(44),NPSG(8),ILOOP,THRESH
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),BETA(44),W(44,2)
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                         GMJ00120
           $     CORANG(44),FCORN(44)                                      GMJ00130
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
            COMMON /SHI/ XP(10),YP(10),NESH(10),NSH
            COMMON /QDI/ YO(44),XO(44),TPHI(44),TPHIN(44)
      C   SHIFT NODES ON SEGMENTS WHERE SHIFTING ALLOWED
            NSHSEG=NSH/2
            WRITE(ILP,50)
         50 FORMAT(1X,/,* NEW (X,Y) POSITIONS FOR SHIFTED NODES*)
            NP1=N+1
            DO 100 IS=1,NSHSEG
            J=2*IS
            I=J-1
            L=NESH(I)
            M=NESH(J)
            XL=X(M)-X(L)
            YL=Y(M)-Y(L)
            XI=XP(J)-XP(I)
            YI=YP(J)-YP(I)
            NS=L+1
            IF(M.EQ.1) M=N+1
            NF=M-1
            ML=M
            DO 175 IG=L,M
            YO(IG)=Y(IG)
            XO(IG)=X(IG)
            TPHI(IG)=0.
        175 TPHIN(IG)=0.
            IF(M.NE.NP1) GOTO 180
            YO(M)=Y(1)
            PHIN(M)=PHIN(1)
            PHI(M)=PHI(1)
            ML=1
        180 CONTINUE
            IF(XI.EQ.0.) GOTO 400
            IF(YI.EQ.0.) GOTO 500
            DO 200 K=NS,NF
            X(K)=X(L)+(X(K)-XP(I))*XL/XI
        200 Y(K)=Y(L)+(Y(K)-YP(I))*YL/YI
            GOTO 150
        400 DO 450 K=NS,NF
            X(K)=X(L)+(Y(K)-YP(I))*XL/YI
        450 Y(K)=Y(L)+(Y(K)-YP(I))*YL/YI
            GOTO 150
        500 DO 550 K=NS,NF
            X(K)=X(L)+(X(K)-XP(I))*XL/XI
```

```
      550 Y(K)=Y(L)+(X(K)-XP(I))*XL/XI
      150 CONTINUE
          IF(YL.LT.0.)GOTO 690
C   ADJUST PHI,PHIN FOR NODE NOS INCREASING IN +VE Y DIRECTION
          DO 600 II=NS,NF
          DO 610 JJ=NS,M
          IF(YO(JJ).GE.Y(II)) GOTO 620
      610 CONTINUE
      620 CALL QUAD(II,JJ,L,M)
      600 CONTINUE
          GOTO 800
C   ADJUST PHI,PHIN FOR NODE NOS INCREASING IN -VE Y DIRECTION
      690 DO 700 II=NS,NF
          DO 710 JJ=NS,M
          IF(YO(JJ).LE.Y(II)) GOTO 720
      710 CONTINUE
      720 CALL QUAD(II,JJ,L,M)
      700 CONTINUE
      800 WRITE(ILP,850)
      850 FORMAT(* NODE  X(INIT)     X(NEW)      Y(INIT)     Y(NEW)     *,
     1*PHI(INIT)   PHI(NEW)   PHIN(INIT)  PHIN(NEW)*)
      900 FORMAT(1X,I3,8F12.6)
          WRITE(ILP,950) L,XP(I),X(L),YP(I),Y(L),PHI(L),PHINL(L)
      950 FORMAT(1X,I3,4F12.6,12X,F12.6,12X,F12.6)
          DO 810 II=NS,NF
          WRITE(ILP,900) II,XO(II),X(II),YO(II),Y(II),PHI(II),TPHI(II),
     1PHIN(II),TPHIN(II)
          PHIN(II)=TPHIN(II)
      810 PHI(II)=TPHI(II)
          WRITE(ILP,950) ML,XP(J),X(ML),YP(J),Y(ML),PHI(ML),PHIN(ML)
      100 CONTINUE
          RETURN
          END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE QUAD(II,JJ,L,M)
      C
      C   QUAD GIVES QUADRATIC INTERPOLATION FOR PHI AND PHIN WHEN
      C   SHIFTING NODES   (CALLED FROM SHIFT)
      C
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                          GMJ00120
           $     CORANG(44),FCORN(44)                                       GMJ00130
            COMMON /QDI/ YO(44),XO(44),TPHI(44),TPHIN(44)
            K=0
            IF((JJ+1).GT.M) K=1
            X00=YO(JJ-1-K)
            X1=YO(JJ-K)
            X2=YO(JJ+1-K)
            X10=X1-X00
            X21=X2-X1
            X102=X1*X1-X00*X00
            X212=X2*X2-X1*X1
            D=X212*X10-X102*X21
            LL=0
            F0=PHI(JJ-1-K)
            F1=PHI(JJ-K)
            F2=PHI(JJ+1-K)
            GOTO 200
        100 F0=PHIN(JJ-1-K)
            IF((JJ-1-K).EQ.L) F0=PHINL(JJ-1-K)
            F1=PHIN(JJ-K)
            F2=PHIN(JJ+1-K)
        200 F10=F1-F0
            F21=F2-F1
            A=(-X21*F10+X10*F21)/D
            BE=(X212*F10-X102*F21)/D
            C=F1-A*X1*X1-BE*X1
            T=A*Y(II)*Y(II)+BE*Y(II)+C
            IF(LL.EQ.1) GOTO 300
            TPHI(II)=T
            LL=1
            GOTO 100
        300 TPHIN(II)=T
            RETURN
            END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE INTERP                                              CAL
      C
      C    INTERPOLATE NODAL POTENTIAL TO CORRESOND TO BORE POSITIONS
            COMMON /MAINE/ X(44),Y(44),RLN(44,44),                        GMJ00120
           $      CORANG(44),FCORN(44)                                    GMJ00130
            COMMON /BET/ NFS1,NFSL,ITIME,IFLGD,NFS
            COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,IDL,IDL,IPR,TEMPX
            WRITE(6,100)
        100 FORMAT(/* INTERP FOR I,TH BORE FROM NODES *)
            DO 10 I=1,NBOR
            S=ABS(SL(I))
            DO 20 J=NFS1,NFSL
            IF(X(J).GT.S) GOTO 30
         20 CONTINUE
            STOP
         30 NDB1(I)=J-1
            CINT(I)=(S-X(J-1))/(X(J)-X(J-1))
            WRITE(6,120) I,NB(I),SL(I),NDB1(I),J,X(J-1),X(J),CINT(I)
        120 FORMAT(I5,I10,F10.1,2I5,2F10.1,F10.4)
         10 CONTINUE
            RETURN
            END
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE MATCH(ITIME)                                          CAL2
      C
      C   SCANS MODEL HYDROGRAPHS AT OBS BORES TO FIND MATCH TIMES, TB(I)
            COMMON /PEZ/ PHI(44),PHIN(44),PHINL(44),CBTA(44),W(44,2)
            COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,ID1,IDL,IPR,TEMPX
            COMMON /HIS/ NHIS,P(15,65),TM(65),NH(15)
            IF(IPR.GE.1) WRITE(6,100)
            DO 10 I=1,MCH
            MSUM=0
            TB(I)=-1.
            DO 20 J=1,NBOR
            MSUM=MSUM+MB(J)
            IF(I.LE.MSUM) GOTO 30
         20 CONTINUE
         30 DP=ZP(I)-ZPO(J)
            ZMOD=ZMO(J)+DP
            JB=J+3
            DO 70 L=1,ITIME
            DM=P(JB,L)-ZMO(J)
            DPTDM=DP-DM
            IF(L.EQ.1) GOTO 80
            IF(DPTDM.LE.0. .AND. DDP.GT.0.) GOTO 60
            IF(DPTDM.GE.0. .AND. DDP.LT.0.) GOTO 60
            DDP=DPTDM
            GOTO 70
         80 DDP=DPTDM
            IF(DPTDM.GT.0.) GOTO 70
            TB(I)=TM(1)*DP/DM
            GOTO 90
         60 DT=DDP/(DDP-DPTDM)
            TB(I)=TM(L-1)+DT*(TM(L)-TM(L-1))
            GOTO 90
         70 CONTINUE
         90 IF(IPR.GE.1) WRITE(6,110) NB(J),I,ZP(I),DP,ZMOD,TB(I)
        100 FORMAT(/* SUMMARY OF MATCH POINT STATUS */,
           1/* BORE    MATCH PT  PROTO-Z   DELTA-Z   MODEL-Z  MATCH TIME*)
        110 FORMAT(1X,I10,I5,4F10.3)
         10 CONTINUE
            RETURN
            END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE COMPARE                                              CAL2
      C
      C   COMPARES TIMES OF PROTOTYPE AND MODEL MATCH POINTS
      C   (MODEL TIME/PROTO TIME  =  K/N) ST. K=HYD CONDUCTIVITY,
      C                                   N=EFFECTIVE POROSITY
            COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,ID1,IDL,IPR,TEMPX
            WRITE(6,100)
            S=S2=SP=SP2=SN=SN2=0.
            SB=SB2=SM=SM2=SBM=0.
            N=MCH
            NN=NP=0
            D1=FLOAT(ID1)-0.125
            DO 10 I=1,MCH
            MSUM=0
            DO 30 J=1,NBOR
            MSUM=MSUM+MB(J)
            IF(I.LE.MSUM) GOTO 40
         30 CONTINUE
         40 RELTP=TP(I)-D1
            IF(TB(I).GT.0.) GOTO 20
            N=N-1
            TRAT=-1.
            GOTO 10
         20 TRAT=TB(I)/RELTP
            S=S+TRAT
            S2=S2+TRAT*TRAT
            SB=SB+RELTP
            SM=SM+TB(I)
            SB2=SB2+RELTP*RELTP
            SM2=SM2+TB(I)*TB(I)
            SBM=SBM+RELTP*TB(I)
            IF(SL(J).GE.0.) GOTO 50
            NN=NN+1
            SN=SN+TRAT
            SN2=SN2+TRAT*TRAT
            GOTO 10
         50 NP=NP+1
            SP=SP+TRAT
            SP2=SP2+TRAT*TRAT
         10 WRITE(6,110) I,NB(J),SL(J),ZP(I),TP(I),RELTP,TB(I),TRAT
        100 FORMAT(/* TIME RATIO FOR MATCH POINTS*,/,* MATCH PT*
           1*  BORE  DISTANCE  PROTO-Z  PROTO-T   REL.P-T   MODEL-T   *,
           2*T-RATIO*)
        110 FORMAT(I5,I10,F8.1,5F10.3)
            TRX=TRSD=TRXP=SDP=TRXN=SDN=COV=CP=CN=-1.
            XB=XM=SDB=SDM=B=A=COD=SEE=SEA=SEB=TCEPT=0.
            FN=FLOAT(N)
            IF(N.GT.0) TRX=S/FN
            IF(N.GT.1) TRSD=SQRT((S2-S*S/FN)/(FN-1.))
            IF(N.GT.1) COV=TRSD/TRX
            IF(N.LE.2) GOTO 160
            XB=SB/FN
            XM=SM/FN
```

```
      FACTB=SB2-SB*XB
      FACTM=SM2-SM*XM
60    SDB=SQRT(FACTB/(FN-1.))
      SDM=SQRT(FACTM/(FN-1.))
      BN=SBM-XB*SM
      B=BN/FACTB
      A=XM-B*XB
65    COD=BN*BN/(FACTB*FACTM)
      SEE=SQRT((SM2-A*SM-B*SBM)/(FN-2.))
      SEA=SEE*SQRT(SB2/(FN*FACTB))
      SEB=SEE/FACTB
      TCEPT=-A/B
70  160 FNP=FLOAT(NP)
      IF(NP.GT.0) TRXP=SP/FNP
      IF(NP.GT.1) SDP=SQRT((SP2-SP*SP/FNP)/(FNP-1.))
      IF(NP.GT.1) CP=SDP/TRXP
      FNN=FLOAT(NN)
75    IF(NN.GT.0) TRXN=SN/FNN
      IF(NN.GT.1) SDN=SQRT((SN2-SN*SN/FNN)/(FNN-1.))
      IF(NN.GT.1) CN=SDN/TRXN
      WRITE(6,120)
  120 FORMAT(/,* TIME RATIO STATISTICS      MEAN     ST. DEV.  *,
80   1*   SD/MEAN    NO. MATCH PTS.*)
      WRITE(6,130) TRX,TRSD,COV,N
  130 FORMAT(*    ALL OBS BORES*,6X,3F11.3,8X,I6)
      WRITE(6,140) TRXP,SDP,CP,NP
  140 FORMAT(*    BORES ON RHS *,6X,3F11.3,8X,I6)
85    WRITE(6,150) TRXN,SDN,CN,NN
  150 FORMAT(*    BORES ON LHS *,6X,3F11.3,8X,I6)
      WRITE(6,170) N,XB,XM,SDB,SDM,B,A,TCEPT,COD,SEE,SEB,SEA
  170 FORMAT(/,* ALL BORES  -  MATCH TIME STATISTICS  -  REAL *,
     1*TIME AND MODEL TIME*,/,*      N    REAL-X     MODEL-X    *,
90   2*REAL-SD  MODEL-SD  B(=TRAT)      A       R INTCPT   COD*
     3* (1)   S.E.E.    S.E.B.    S.E.A.*,/,1X,I6,11F10.3)
      RETURN
      END
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE BOREFIT(ITIME,LPR)                                    CAL3,4
      C
      C   READ OBSERVATION BORE WATER LEVELS AND COMPARE WITH MODEL OUTPUT
      C
            COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NOB1(12),NBOR,ID1,IDL,IPR,TEMPX
            COMMON /CALQ/ ISN,Q(6),TRATIO,QLOSS(65),BEDINV,ND,QISN(65)
            COMMON /HIS/ NHIS,P(15,65),TM(65),NH(15)
            COMMON /PLT/ DY(65),AHD(65),AHDM(65)
            DIMENSION SP(12),SP2(12),SM(12),SM2(12),SD(12),SD2(12),SPM(12),
           1SEN(12),CHI2(12)
            D1=FLOAT(ID1)-0.125
            DL=FLOAT(IDL)+0.875
            MCH=0
            WRITE(6,220)
        220 FORMAT(/,* STATISTICS OF FIT OF MODEL O/P AGAINST BORE RECORDS*/)
            IF(IPR.LT.3) GOTO 340
            GLSCALE=0.15
            GLZERO=-0.3
            TSCALE=4.
            TZERO=FLOAT(ID1)
      C   INITIALIZE CALCOMP PLOT ON 25CM PLOTTER
            CALL PLOT25
            CALL XLIMIT(60.)
            CALL PAUPLOT(18HBLANK PAPER PLEASE,18)
            CALL SYMBOL(1.,18.,1.,16HBOX 39 PLS NCDPD,-90.,16)
            CALL PLOT(7.,3.,-1)
            CALL AXIS(0.,0.,10HDAY NUMBER,-10,12.,0.,TZERO,TSCALE,1)
            CALL AXIS(0.,0.,36HBORE WATER ELELATION CHANGE (METRES),36,21.,
           190.,GLZERO,GLSCALE,-1)
            CALL SYMBOL(20.,13.,0.28,4HBORE,0.,4)
            CALL SYMBOL(22.,13.,0.28,9HPROTOTYPE,0.,9)
            CALL SYMBOL(25.,13.,0.28,5HMODEL,0.,5)
        340 CONTINUE
            DO 900 I=1,NBOR
            L=1
            JB=I+3
      C   CALCULATE STATISTICS FOR PROTOTYPE (P),MODEL (M), AND RESIDUAL (D)
            SP(I)=SP2(I)=SM(I)=SM2(I)=0.
            SD(I)=SD2(I)=SPM(I)=SEN(I)=CHI2(I)=0.
        910 READ(8,920) NBR
        920 FORMAT(I10)
            IF(NBR.LT.NB(I))GOTO 910
            IF(IPR.EQ.1) WRITE(6,*) NBR
        930 READ(8,940) NBR,DYR
        940 FORMAT(I10,13X,F8.3)
            IF(DYR.LT.D1) GOTO 930
            BACKSPACE 8
            J=1
        980 READ(8,960) NBR,NDATE,DY(J),AHD(J)
        960 FORMAT(I10,1X,I6,6X,2F8.3)
            IF(NBR.NE.NB(I) .OR. DY(J).GT.DL) GOTO 990
            IF(AHD(J).GE.99.99) GOTO 980
            IF(LPR.GT.0) WRITE(6,970) J,NBR,NDATE,DY(J),AHD(J)
```

```
      970 FORMAT(I6,I10,1X,I6,6X,2F8.3)
          RELTP=DY(J)-D1
   60     TMOD=RELTP*TRATIO
      110 IF(TM(L).GE.TMOD) GOTO 120
          L=L+1
          IF(L.GT.ITIME) GOTO 990
          GOTO 110
   65 120 IF(L.NE.1) GOTO 130
          TD=TMOD/TM(1)
          ZM=ZMO(I)+TD*(P(JB,1)-ZMO(I))
          GOTO 140
      130 TD=(TMOD-TM(L-1))/(TM(L)-TM(L-1))
   70     ZM=P(JB,L-1)+TD*(P(JB,L)-P(JB,L-1))
      140 DP=AHD(J)-ZPO(I)
          DM=ZM-ZMO(I)
          D=DP-DM
          AHD(J)=DP
   75     AHDM(J)=DM
          DY(J)=RELTP
          SP(I)=SP(I)+DP
          SM(I)=SM(I)+DM
          SD(I)=SD(I)+D
   80     SP2(I)=SP2(I)+DP*DP
          SM2(I)=SM2(I)+DM*DM
          SD2(I)=SD2(I)+D*D
          SPM(I)=SPM(I)+DP*DM
          IF(DP.NE.0.) SEN(I)=SEN(I)+ABS(D/DP)
   85     IF(DM.EQ.0.) GOTO 150
          CHI2(I)=CHI2(I)+D*D/ABS(DM)
          GOTO 160
      150 IF(DP.NE.0.) CHI2(I)=CHI2(I)+D*D/ABS(DP)
      160 J=J+1
   90     GOTO 980
      990 MB(I)=J-1
          MCH=MCH+MB(I)
          MBB=J-1
        C   PERFORM LINE PRINTER PLOT IF REQUIRED
   95     IF(IPR.LT.2) GOTO 300
          WRITE(6,310) NB(I),TRATIO
      310 FORMAT(1H1,* BORE *,I10,*  +      MODEL  D  *,6X,*TRATIO = *,
         1F10.3,/)
          CALL LPLOT(MBB)
  100   C   PERFORM CALCOMP PLOT IF REQUIRED
          IF(IPR.LT.3) GOTO 300
        C   PLOT OBSERVED GROUNDWATER ELEVATION CHANGE
          DY(MBB+1)=TZERO
          DY(MBB+2)=TSCALE
  105     AHD(MBB+1)=GLZERO
          AHD(MBB+2)=GLSCALE
          DO 450 K=1,MBB
      450 DY(K)=DY(K)+TZERO
          CALL LINE(DY,AHD,MBB,1,-1,I)
  110   C   PLOT MODEL GROUNDWATER ELEVATION CHANGE
          DO 500 K=1,ITIME
          DY(K)=TM(K)/TRATIO + D1
          AHDM(K)=P(JB,K)-ZMO(I)
      500 WRITE(6,*) K,DY(K),AHDM(K)
```

```
115               DY(ITIME+1)=ID1
                  DY(ITIME+2)=TSCALE
                  AHDM(ITIME+1)=GLZERO
                  AHDM(ITIME+2)=GLSCALE
                  JPOINTS=I+20
120               CALL LINE(DY,AHDM,ITIME,1,JPOINTS,JPOINTS)
                  NUMB=NB(I)
                  YPRINT=12.6-FLOAT(I)*0.8
                  CALL NUMBER(18.7,YPRINT,0.28,NUMB,0.,3HI10)
                  CALL SYMBOL(23.,YPRINT,0.28,I,0.,-1)
125               CALL SYMBOL(25.5,YPRINT,0.28,JPOINTS,0.,-1)
                  YLINE=YPRINT+0.15
                  CALL PLOT(24.8,YLINE,3)
                  CALL PLOT(25.35,YLINE,2)
                  CALL PLOT(25.95,YLINE,3)
130               CALL PLOT(26.5,YLINE,2)
                  CALL PLOT(26.6,YLINE,3)
              300 IF(IPR.LT.1) GOTO 900
                  WRITE(6,320) NB(I),TRATIO
              320 FORMAT(/,* BORE *,I10,*      TRATIO =*,F10.3,/,
135              1*  INDEX    REL.P-T    DELTA.P-Z  DELTA.M-Z  DP.-DM.-Z*)
                  DO 400 K=1,MBB
                  DFK=AHD(K)-AHDM(K)
              400 WRITE(6,330) K,DY(K),AHD(K),AHDM(K),DFK
              330 FORMAT(I6,4F11.3)
140           900 CONTINUE
            C   COMPILE STATISTICS FOR EACH BORE AND COMBINE
                  SEN=TSP=TSM=TSD=TSP2=TSM2=TSD2=TSPM=TSENS=TCHI2=0.
                  WRITE(6,230) TRATIO
                  WRITE(6,240)
145               DO 200 I=1,NBOR
                  FN=FLOAT(MB(I))
                  SEN=SEN+FN
                  TSP=TSP+SP(I)
                  TSM=TSM+SM(I)
150               TSD=TSD+SD(I)
                  TSP2=TSP2+SP2(I)
                  TSM2=TSM2+SM2(I)
                  TSD2=TSD2+SD2(I)
                  TSPM=TSPM+SPM(I)
155               TSENS=TSENS+SEN(I)
                  TCHI2=TCHI2+CHI2(I)
                  IF(MB(I).GT.1) GOTO 210
                  WRITE(6,270) NB(I),MB(I)
              270 FORMAT(1X,I10,I4)
160               GOTO 200
              210 XP=SP(I)/FN
                  XM=SM(I)/FN
                  FACTP=SP2(I)-SP(I)*XP
                  FACTM=SM2(I)-SM(I)*XM
165               SDP=SQRT(FACTP/(FN-1.))
                  SDM=SQRT(FACTM/(FN-1.))
                  XD=SD(I)/FN
                  RMSS=SQRT(SD2(I)/FN)
                  BN=SPM(I)-XP*SM(I)
170               B=BN/FACTP
                  A=XM-B*XP
```

```
      COD=BN*BN/(FACTP*FACTM)
      SEE=SEA=SEB=-1.
      IF(MB(I).LE.2) GOTO 280
175   SEE=SQRT((SM2(I)-A*SM(I)-B*SPM(I))/(FN-2.))
      SEA=SEE*SQRT(SP2(I)/(FN*FACTP))
      SEB=SEE/FACTP
  280 SENS=SEN(I)*100./FN
      WRITE(6,250) NB(I),MB(I),XP,XM,SDP,SDM,XD,RMSS,B,A,COD,SEE
180  1,SEB,SEA,SENS,CHI2(I)
  250 FORMAT(1X,I10,I4,6F9.3,8F8.3)
  240 FORMAT(*   BORE        N     OBS-X     MOD-X    OBS-SD    MOD-SD *,
     1* RESID-X RES-RMSS    B (1)    A (0) COD (1)*,
     2*  S.E.E.  S.E.B.  S.E.A.    SENS     CHI2  *)
185   230 FORMAT(//,* TIME RATIO = *,F10.3,/)
  200 CONTINUE
      TXP=TSP/SFN
      TXM=TSM/SFN
      TFACTP=TSP2-TSP*TXP
190   TFACTM=TSM2-TSM*TXM
      TSDP=SQRT(TFACTP/(SFN-1.))
      TSDM=SQRT(TFACTM/(SFN-1.))
      TXD=TSD/SFN
      TRMSS=SQRT(TSD2/SFN)
195   TBN=TSPM-TXP*TSM
      TTB=TBN/TFACTP
      TA=TXM-TTB*TXP
      TCOD=TBN*TBN/(TFACTP*TFACTM)
      TSEE=TSEA=TSEB=-1.
200   IF(SFN.LE.2.) GOTO 290
      TSEE=SQRT((TSM2-TA*TSM-TTB*TSPM)/(SFN-2.))
      TSEA=TSEE*SQRT(TSP/(SFN*TFACTP))
      TSEB=TSEE/TFACTP
  290 TSENS=TSENS*100./SFN
205   WRITE(6,260) SFN,TXP,TXM,TSDP,TSDM,TXD,TRMSS,TTB,TA,TCOD,TSEE,
     1TSEB,TSEA,TSENS,TCHI2
  260 FORMAT(*    ALL  *,F7.0,F8.3,5F9.3,8F8.3)
      RETURN
      END
```

```
1          C
           C-----------------------------------------------------------------------
           C
                 SUBROUTINE STREAM(HA,HP,H2N,TIME)                                  CAL4
5          C
           C    STREAM CALCULATES STREAM HEAD AND DAILY DISCHARGE LOSS
                  COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
                 1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,IDL,IDL,IPR,TEMPX
                  COMMON /CALQ/ ISN,Q(6),TRATIO,QLOSS(65),BEDINV,ND,QISN(65),PLENG
10         C    DISCHARGE VALUE = -2  IF MISSING
           C                    = -1  IF DISCHARGE RATE EXCEEDS STATION RATING
           C    READ DAILY DISCHARGE FILE
                  H2=H2N
                  HP=HA
15                READN=0.
               10 READ(7,800) NDAY,NDATE,(Q(I),I=1,6),RAIN,RAINP,EVAP,TMEAN,TMAX
              800 FORMAT(I5,1X,I6,6F8.3,5F8.1)
                  READN=READN+1.
                  IF(NDAY.LT.ID1) GOTO 10
20                IF(IPR.GE.1) WRITE(6,850) NDAY,NDATE,(Q(I),I=1,6),RAIN,RAINP,EVAP,
                 1TMEAN,TMAX
              850 FORMAT(1X,I5,1X,I6,6F8.3,5F8.1)
                  TEMPX=TMEAN
                  IF(NDAY.GT.IDL) RETURN
25                ND=NDAY-ID1
                  IF(ND.NE.0) GOTO 20
                  IF(Q(ISN).LT.0.) Q(ISN)=0.
                  GOTO 200
               20 QISN(ND)=Q(ISN)
30                ISND=ISN+1
                  IF(Q(ISN).GE.0.) GOTO 100
                  IF(Q(ISN).EQ.-1.) GOTO 60
                  IREC=ISN
               30 IREC=IREC-1
35                IF(Q(IREC).GE.0.) GOTO 40
                  IF(IREC.EQ.1) GOTO 50
                  GOTO 30
               40 CALL CORRELQ(IREC)
                  GOTO 120
40         C     ASSUME H2N UNCHANGED
               50 QLOSS(ND)=-ABS(QLOSS(ND-1))
                  GOTO 400
           C     LOOK FOR STATION UPSTREAM WITHIN RATING (SET Q(ISN)=Q(IREC) )
               60 IREC=ISN
45             70 IREC=IREC-1
                  IF(Q(IREC).GT.0.) GOTO 80
                  IF(IREC.EQ.1) GOTO 90
                  GOTO 70
               80 Q(ISN)=Q(IREC)
50                GOTO 120
               90 Q(ISN)=120.
                  GOTO 120
           C     CHECK FOR RECORD AT D/S STATION
              100 IF(Q(ISND).GE.0.) GOTO 110
55                IREC=ISN
                  GOTO 120
              110 QLOSS(ND)=Q(ISN)-Q(ISND)
```

```
      IF(Q(ISN).GT.28. .OR. Q(ISND).GT.28.) QLOSS(ND)=-ABS(QLOSS(ND))
      IF(Q(ISN).EQ.0.  .OR. Q(ISND).EQ.0. ) QLOSS(ND)=-ABS(QLOSS(ND))
60    IF(RAIN.GT.0.1) QLOSS(ND)=-ABS(QLOSS(ND))
      GOTO 200
C   ESTIMATE QLOSS FROM DRY WEATHER CORRELATION
  120 ISN=ISN+1
      CALL CORRELQ(IREC)
65    ISN=ISN-1
      QLOSS(ND)=-ABS(Q(ISN)-Q(ISN+1))
C   CALC STREAM HEAD FROM DAILY DISCHARGE
C   EMPIRICAL RATING USED AS MANNINGS EQN INADEQUATE
  200 A=320.
70    C=2.88
      DEPTH=(Q(ISN)/A)**(1./C)
      H2N=BEDINV+DEPTH
      HA=H2N
      IF(ND.NE.0) GOTO 250
75    HP=HA
      RETURN
  250 CONTINUE
      WRITE(6,300) ND,NDAY,NDATE,H2N,QISN(ND),Q(ISN),Q(ISND),QLOSS(ND)
     1,RAIN
80  300 FORMAT(* INDEX   DAY NO.   DATE    H2N   QISN(ND) Q(ISN)  *
     1*Q(ISND)  QLOSS(ND)   RAIN*,/,I5,2I9,F8.3,4F9.3,F9.1)
  400 HA=H2N
      TND=FLOAT(ND)*TRATIO
      IF(TIME.GT.TND) GOTO 10
85    H2N=HP+(HA-HP)*(TIME-TND+READN*TRATIO)/(READN*TRATIO)
      WRITE(6,450) H2N,HP,H2,HA
  450 FORMAT(* H2N , HP , H2 , HA  =*,4F10.3)
      RETURN
      END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE CORRELQ(IREC)                                    CAL4
      C
      C   CALCULATES DAILY DISCHARGE AT STN ISN FROM DISCHARGE AT AN UPSTREAM
      C   STATION, ISND, USING MEAN DRY WEATHER CORRELATIONS
            COMMON /CALQ/ ISN,Q(6),TRATIO,QLOSS(65),BEDINV,ND,JISN(65),RLENG
            DIMENSION A(4),C(4)
            A(1)=.927
            C(1)=-1.28
            A(2)=.864
            C(2)=-2.79
            A(3)=.745
            C(3)=-1.65
            A(4)=.980
            C(4)=-2.22
            IR=IREC
            Q1=Q(IREC)
         10 Q2=Q1*A(IR-1)+C(IR-1)
            IR=IR+1
            Q1=Q2
            IF(IR.NE.ISN) GOTO 10
            IF(Q2.LT.0.) Q2=0.
            Q(ISN)=Q2
            RETURN
            END
```

```
      C
      C-----------------------------------------------------------------------
      C
            SUBROUTINE FLOWFIT(ITIME,ICAL,PERM)                                CAL4,5
      C
      C   COMPARES OBSERVED DAILY DISCHARGE LOSS WITH STREAMBED INTAKE
      C   RATE FROM MODEL O/P
      C
            COMMON /CAL/ MCH,NB(12),SL(12),MB(12),ZPO(12),TP(24),ZP(24),
           1ZMO(12),TB(24),CINT(12),NDB1(12),NBOR,ID1,IDL,IPR,TEMPX
            COMMON /CALQ/ ISN,Q(5),TRATIO,QLOSS(65),BEDINV,ND,QISN(65),RLENG
            COMMON /HIS/ NHIS,P(15,65),TM(65),NH(15)
            COMMON /PLT/ DY(65),AHD(65),AHDM(65)
      C
            IF(ICAL.LT.6) GOTO 40
      C   CREATE FILE (TAPE9) FOR ANALYSIS OF DISCHARGE RESIDUALS
            ISND=ISN+1
            NDD=IDL-ID1+2
            DO 30 I=1,NDD
         30 BACKSPACE 7
            READ(7,310) NDAY,NDATE,(Q(I),I=1,6),RAIN,RAINP,EVAP,TMEANP,TMAXP
        310 FORMAT(I5,1X,I6,6F8.3,5F8.1)
            WRITE(6,320) NDAY,NDATE,(Q(I),I=1,6),RAIN,RAINP,EVAP,TMEANP,TMAXP
        320 FORMAT(1X,I5,1X,6F8.3,5F8.1)
         40 CONTINUE
            SP=SM=SP2=SM2=SPM=SQRAT=SQRAT2=0.
            QMMAX=QPMAX=0.
            QMSUM=QDSUM=QUSUM=0.
            L=1
            N=M=0
            D1=FLOAT(ID1)-0.125
            DL=FLOAT(IDL)+0.875
            WRITE(6,160)
            WRITE(6,210)
            DO 10 I=1,ND
            RELTP=FLOAT(I)
            TMOD=RELTP*TRATIO
        110 IF(TM(L).GE.TMOD) GOTO 120
            L=L+1
            IF(L.GT.ITIME) GOTO 150
            GOTO 110
        120 IF(L.NE.1) GOTO 130
            QLS=P(1,1)
            GOTO 140
        130 TD=(TMOD-TM(L-1))/(TM(L)-TM(L-1))
            QLS=P(1,L-1)+TD*(P(1,L)-P(1,L-1))
        140 QL=QLS*RLENG*2.
            IF(ICAL.GE.5) QL=QL*PERM
      C   RECHARGE MAY BE LIMITED BY UPSTREAM DISCHARGE RATE
            QUS=QISN(I)*1000.
            QUSUM=QUSUM+QUS
            IF(ICAL.GE.5 .AND. QL.GT.QUS) QL=QUS
            QMSUM=QMSUM+QL
            QLOSS(I)=QLOSS(I)*1000.
            IF(ICAL.LT.6) GOTO 50
      C   WRITE TAPE9
            QPMOM=QLOSS(I)-QL
```

```
      IF(QLOSS(I).LT.0.) QPMQM=-QLOSS(I)-QL
      READ(7,310) NDAY,NDATE,(Q(J),J=1,6),RAIN,RAINP,EVAP,TMEAN,TMAX
60    Q(2)=Q(2)*1000.
      Q(ISND)=Q(ISND)*1000.
      WRITE(9,330) NDAY,NDATE,Q(2),QUS,Q(ISND),QLOSS(I),QL,QPMQM,RAIN,
     1RAINP,EVAP,TMEANP,TMAXP
  330 FORMAT(I5,1X,I6,6F8.0,5F8.1)
65    TMEANP=TMEAN
      TMAXP=TMAX
   50 CONTINUE
      M=M+1
      DY(M)=RELTP
70    AHDM(M)=QL
      IF(ABS(QL).GT.QMMAX) QMMAX=ABS(QL)
      AHD(M)=QLOSS(I)
      IF(ABS(QLOSS(I)).GT.QPMAX) QPMAX=ABS(QLOSS(I))
      IF(QLOSS(I).LE.0.) GOTO 20
75  C   CALCULATE STATISTICS OF FIT FOR RELIABLE DISCHARGE LOSS DATA ONLY
    C   QL = EXCHANGE FLOW FROM MODEL (AT X-SECTION * REACH LENGTH * 2 SIDES)
    C   QLOSS = EXCHANGE FLOW MEASURED (M*M*M/DAY)
    C   UNITS - QL - M*M.     QLOSS(I) - M*M*M/DAY   IE. K=QLOSS/QL - M/DAY.
      N=N+1
80    QRAT=0.
      IF(QL.NE.0.) QRAT=QLOSS(I)/QL
      SP=SP+QLOSS(I)
      SM=SM+QL
      SP2=SP2+QLOSS(I)*QLOSS(I)
85    SM2=SM2+QL*QL
      SPM=SPM+QLOSS(I)*QL
      SQRAT=SQRAT+QRAT
      SQRAT2=SQRAT2+QRAT*QRAT
      WRITE(6,220) I,RELTP,TMOD,QLS,QL,QLOSS(I),QUS,QRAT
90  210 FORMAT(4X,*I     REL.P-T     MODEL-T     INFLUX    MODEL-LOSS*,
     1*  PROTO-LOSS    Q (U/S)   LOSS-RATIO*)
  220 FORMAT(I5,7F12.3)
      GOTO 10
   20 QDSUM=QDSUM-QLOSS(I)
95    10 CONTINUE
      QPSUM=SP+QDSUM
      WRITE(6,300) SP,QDSUM,QPSUM,SM,QMSUM,QUSUM
  300 FORMAT(* TOTAL RECHARGE*/,* PROTO-MEAS  PROTO- EST   PROTO- TOT*,
     1*  MODEL-MEAS  MODEL- TOT  DISCHARGE U/S*,/,6F12.3)
100   IF(ICAL.LT.6) GOTO 150
    C   PRINT CONTENTS OF NEW FILE TAPE9
      REWIND 9
      WRITE(6,340)
  340 FORMAT(/,* TAPE9*,/* DAY NO  DATE  STN 03  STN U/S STN D/S*,
105   1* LOSS-P  LOSS-M LOSS P-M   RAIN   RAINP   EVAP   TMEANP *,
     2*  TMAXP *,/)
      DO 60 I=1,ND
      READ(9,330) NDAY,NDATE,Q(2),QUS,Q(ISND),QLOSS(I),QL,QPMQM,RAIN,
     1RAINP,EVAP,TMEANP,TMAXP
110    60 WRITE(6,350) NDAY,NDATE,Q(2),QUS,Q(ISND),QLOSS(I),QL,QPMQM,RAIN
     1,RAINP,EVAP,TMEANP,TMAXP
  350 FORMAT(1X,I5,1X,I6,6F8.0,5F8.1)
  150 IF(N.EQ.0) RETURN
      FN=FLOAT(N)
```

```
115               XP=SP/FN
                  XM=SM/FN
                  BN=SPM-XP*SM
                  B=BN/(SP2-XP*SP)
                  A=XM-B*XP
120               COD=BN*BN/((SP2-XP*SP)*(SM2-XM*SM))
                  XRAT=SQRAT/FN
                  SDRAT=SQRT((SQRAT2-SQRAT*XRAT)/(FN-1.))
                  SDX=SDRAT/XRAT
                  SEE=SEA=SEB=-1.
125               IF(N.LE.2) GOTO 190
                  SEE=SQRT((SM2-A*SM-B*SPM)/(FN-2.))
                  SEA=SEE*SQRT(SP2/(FN*(SP2-SP*XP)))
                  SEB=SEE/(SP2-SP*XP)
              190 WRITE(6,170)
130               WRITE(6,180) N,XP,XM,B,A,COD,SEE,SEB,SEA
              160 FORMAT(/* STATISTICS OF FIT OF MODEL O/P AGAINST*,
                 1* DISCHARGE LOSS RECORDS*/)
              170 FORMAT(*    N    OBS-X     MOD-X     B  (1)    A  (0) *,
                 1*  COD  (1)    S.E.E.    S.E.B.    S.E.A. *)
135           180 FORMAT(1X,I5,8F10.3)
                  EK=1./B
                  EN=EK/TRATIO
                  WRITE(6,200) EK,EN
              200 FORMAT(* EST FOR AQ. HYD. CONDUCTIVITY =*,F10.5,*   EFFECTIVE*,
140                1* POROSITY =*,F10.5)
                  WRITE(6,230) XRAT,SDRAT,SDX
              230 FORMAT(/,* QLOSS RATIO (PROTO/MODEL)  MEAN , SD , SD/MEAN =*,
                 13F10.5)
                  EKX=XRAT
145               IF(ICAL.GE.5) EKX=XRAT*PERM
                  NLOOP=0
              290 ENX=EKX/TRATIO
                  WRITE(6,200) EKX,ENX
                  SD=SD2=0.
150               DO 250 I=1,N
                  IF(QLOSS(I).LE.0.) GOTO 250
                  D=AHD(I)-AHDM(I)*XRAT
                  SD=SD+D
                  SD2=SD2+D*D
155           250 CONTINUE
                  XD=SD/FN
                  RMSS=SQRT(SD2/FN)
                  WRITE(6,260) RMSS,XD
              260 FORMAT(* FLOW RESIDUAL  RMSS=*,F12.5,10X,*MEAN=*,F12.5)
160               IF(ICAL.LT.5) GOTO 270
                  IF(NLOOP.EQ.1) GOTO 270
                  NLOOP=1
                  XRAT=1.
                  EKX=PERM
165               WRITE(6,280) PERM
              280 FORMAT(/* VALIDATION RESULTS WITH  K =*,F10.5)
                  GOTO 290
              270 IF(IPR.LT.2) RETURN
                  PMODMAX=XRAT*QMMAX
170               IF(PMODMAX.GT.QPMAX) QPMAX=PMODMAX
                  SCALE=5./QPMAX
```

```
                DO 240 I=1,M
                AHDM(I)=AHDM(I)*XRAT*SCALE
              240 AHD(I)=AHD(I)*SCALE
175             MBB=-M
                CALL LPLOT(MBB)
                IF(IPR.LT.3) RETURN
            C   PERFORM CALCOMP PLOT OF DISCHARGE LOSSES
                TSCALE=4.
180             TZERO=FLOAT(ID1)
                QSCALE=1000.
                CALL PLOT(35.,0.,-1)
                CALL AXIS(0.,0.,10HDAY NUMBER,-10,13.,0.,TZERO,TSCALE,1)
                CALL AXIS(0.,0.,34HDISCHARGE LOSS  (CUBIC METRES/DAY),34,14.
185            1,90.,0.,QSCALE,-1)
                CALL SYMBOL(4.,18.,0.28,13HU/S DISCHARGE,0.,13)
                CALL SYMBOL(4.,17.,0.28,14HPROTOTYPE LOSS,0.,14)
                CALL SYMBOL(4.,16.,0.28,13HDOUBTFUL LOSS,0.,13)
                CALL SYMBOL(4.,15.,0.28,10HMODEL LOSS,0.,10)
190         C   PLOT MODEL DISCHARGE LOSSES
                LPEN=3
                DO 370 I=1,M
                AHDM(I)=AHDM(I)/SCALE
                AHD(I)=AHD(I)/SCALE
195             DY(I)=DY(I)+TZERO
                DUMX=(DY(I)-TZERO)/TSCALE
                IF(AHDM(I).LT.0.) GOTO 360
                DUMY=AHDM(I)/QSCALE
                CALL PLOT(DUMX,DUMY,LPEN)
200             LPEN=2
                GOTO 370
              360 LPEN=3
              370 CONTINUE
                CALL PLOT(DUMX,DUMY,3)
205         C   PLOT PROTOTYPE DISCHARGE LOSSES
                DO 390 I=1,M
                DUMX=(DY(I)-TZERO)/TSCALE
                DUMY=AHD(I)/QSCALE
                IF(AHD(I).LT.0.) GOTO 380
210             CALL SYMBOL(DUMX,DUMY,0.28,0,0.,-1)
                GOTO 390
              380 DUMY=ABS(DUMY)
                CALL SYMBOL(DUMX,DUMY,0.28,25,0.,-1)
              390 CONTINUE
215         C   PLOT U/S DISCHARGE
                QSCALE=2000.
                DO 400 I=1,M
                DUMX=(DY(I)-TZERO)/TSCALE
                DUMY=QISN(I)*1000./QSCALE
220             IF(QISN(I).LT.0.) GOTO 400
                CALL SYMBOL(DUMX,DUMY,0.14,4,0.,-1)
              400 CONTINUE
                CALL AXIS(13.,0.,32HU/S DISCHARGE (CUBIC METRES/DAY),-32,14.,
               190.,9.,QSCALE,-1)
225             CALL SYMBOL(8.,18.,0.14,4,0.,-1)
                CALL SYMBOL(8.,17.,0.28,0,0.,-1)
                CALL SYMBOL(8.,16.,0.28,25,0.,-1)
                CALL PLOT(7.5,15.15,3)
```

```
                CALL PLOT(8.5,15.15,2)
230             CALL PLOT(8.6,15.15,3)
                RETURN
                END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE LPLOT(MBB)                                          CAL3,4
      C
      C   LPLOT GIVES LINE PRINTER PLOT OF OBSERVED AND MODEL VALUES
      C
            COMMON /PLT/ DY(65),AHD(65),AHDM(65)
            DIMENSION DUM(125),A(7)
            DATA A/1H ,1HO,1H+,1H*,1H-,1H=,1H1/
            DUM(1)=A(7)
            DO 100 I=2,125
          100 DUM(I)=A(1)
            IFQ=0
            IF(MBB.GT.0) GOTO 50
            IFQ=1
            MBB=-MBB
            WRITE(6,20)
            WRITE(6,21)
            WRITE(6,22)
            GOTO 60
           50 WRITE(6,15)
            WRITE(6,16)
            WRITE(6,17)
           15 FORMAT(60X,*METRES*)
           16 FORMAT(10X,*0*,19X,*1*,19X,*2*,19X,*3*,19X,*4*,19X,*5*,19X,*6*)
           17 FORMAT(10X,*=*,30(*...1*))
           18 FORMAT(10X,*1*)
           20 FORMAT(1H1,*  DATA +      MODEL 0*,20X,*EXCHANGE FLOW RATE  (AS*
             1* PERCENTAGE OF PEAK)*)
           21 FORMAT(10X,*0*,18X,*20*,18X,*40*,18X,*60*,18X,*80*,17X,*100*)
           22 FORMAT(10X,*=*,20(*....1*))
           60 IRP=0
            DO 1 I=1,MBB
            IR=INT(DY(I)+0.5)
            JM=INT(AHDM(I)*20.+1.5)
            JP=INT(AHD(I)*20.+1.5)
            8 IRP=IRP+1
            IF(IR-IRP) 5,6,7
            7 WRITE(6,18)
            WRITE(6,18)
            GOTO 8
            6 WRITE(6,18)
            5 IF(JM.GE.0) GOTO 31
            JM=-JM
            DUM(JM)=A(4)
            GOTO 32
           31 IF(JM.GT.125) JM=125
            DUM(JM)=A(2)
           32 IF(JP.GE.0) GOTO 33
            JP=-JP
            DUM(JP)=A(5)
            GOTO 34
           33 IF(JP.GT.125) JP=125
            DUM(JP)=A(3)
           34 IF(JP.EQ.JM) DUM(JP)=A(6)
            WRITE(6,40) DY(I),(DUM(J),J=1,125)
```

```
              40 FORMAT(F9.3,1X,125A1)
                 DUM(JM)=A(1)
60               DUM(JP)=A(1)
                 DUM(1)=A(7)
               1 CONTINUE
                 WRITE(6,18)
                 IF(IFQ.EQ.1) GOTO 70
65               WRITE(6,17)
                 WRITE(6,16)
                 RETURN
              70 WRITE(6,22)
                 WRITE(6,21)
70               RETURN
                 END
```

```
1
      C
      C-----------------------------------------------------------------
      C
5           FUNCTION TCOMP(TEMP)                                          CAL7
      C
      C    TCOMP GIVES ADJUSTMENT OF STREAMBED IMPEDANCE DUE TO TEMPERATURE
      C    DYNAMIC VISCOSITY OF WATER (PA-S) VERSUS TEMPERATURE FROM
      C      VENNARD,JK AND STREET,RL (1976) ELEMENTARY FLUID MECHANICS, 5TH
10    C      EDITION,JOHN WILEY AND SONS, NEW YORK. (APPENDIX 2, P705)
      C
           DIMENSION T(9),V(9)
           DATA (T(I),I=1,9)/0.,5.,10.,15.,20.,25.,30.,40.,50./
           DATA (V(I),I=1,9)/1.781,1.518,1.307,1.139,1.002,0.890,0.798,
15        10.653,0.547/
           I=2
         10 IF(TEMP.LE.T(I)) GOTO 20
           I=I+1
           GOTO 10
20       20 DV=V(I-1)+(V(I)-V(I-1))*(TEMP-T(I-1))/(T(I)-T(I-1))
           TCOMP=DV/0.982
      C    CALIBRATION PERIOD MEAN TEMP=20.9 DEGREES CENTIGRADE.
           RETURN
           END
```

DATA PROCESSING PROGRAMS

```
      PROGRAM BPL(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,TAPE7,
     1PLOT)
      COMMON TN(1280),VN(1280),IDS(140)
      DIMENSION T(4000),V(4000)
C     ---------------------------------------------------------------
C     BPL CONTROL PARAMETERS:
C     NBOR,IFTCON,IFPF
C     IDTYPE,NGRAPH,IFSEL
C     NBOR  = NUMBER OF BORES TO BE PROCESSED
C     IFTCON= 0 IRREGULAR TIME INTERVALS (AS PER DATA)
C           = 1 REGULAR TIME INTERVALS (BUT NOT FOR RIVER DISCHARGE)
C     IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C           = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C     BPL SELECTS DATA FOR PLOTTING FROM A BORE DATA
C     FILE  : SPECIFICATIONS ARE -
C      1) START AND FINISH DATES (INCLUSIVE)  (ID1,IM1,IY1,ID2,IM2,IY2)
C      2) MINIMUM TIME INTERVAL BETWEEN VALUES PLOTTED - DELTAT
C           IN HOURS  DEFAULT=0.
C      3) OPTION FOR OVER-RIDING DELTAT FOR CHANGES IN DATA VALUE
C           EXCEEDING SPECIFIED VALUE - DELTAV   DEFAULT=100.
C      4) DATA TYPE FOR PLOTTING (IFTCON = 0 )
C           IDTYPE=0  PLOT OF RAW DATA FILE (SUBJECT TO DELTAT,DELTAV)
C           IDTYPE=-1 CONVERTS BORE GL'S TO AHD VALUES
C           IDTYPE=-2 GIVE SMOOTHING ESTIMATES FOR GL DATA
C     BPL ALLOWS ANY NUMBER OF PLOTS ON ONE PAIR OF AXES
C     INPUT QUEUE:
C      NBOR,IFTCON,IFPF
C      IDTYPE,NGRAPH,IFSEL
C      ID1,IM1,IY1,ID2,IM2,IY2
C      IDS(1) = ID OF FIRST BORE FOR PROCESSING
C       IDS(2 TO NBOR)  -  IF IFSEL =1
C       DELTAT,DELTAV  -  IF IFTCON=0
C       TINT,STIME     -   "     "  =1
C       ALL 4          -  IF IFTCON=2
C       SCALES (SX,SY,ZY)  " NGRAPH=1
C
C     MOST USEFUL COMBINATIONS
C     IDTYPE  IFTCON   DELTAT,DELTAV   TINT,STIME   APPLICATION
C                        REQD           REQD
C       0        0          Y                      RAW DATA/VERIFY
C      -2        0          Y                      SMOOTHING ESTIMATES
C      -1        1                       Y         AHD AT REG INTERVALS
C
C     BPL USES THE FOLLOWING SET OF SUBROUTINES AND FUNCTIONS :
C     AHD  DATE  DECDAY  GRF  ICSMOU  MINMAX  SFOWN  SMOOTH.
C
C-----------------------------------------------------------------
C     INITIALIZE
      NMAX=4000
      NNMAX=1280
      DELTAT=0.
      DELTAV=100.
      NN=0
      I=0
      NG=0
      IFPLT=0
```

```
      C   READ IN CONTROL DATA
            READ(8,*) NBOR,IFTCON,IFPF
            WRITE(6,49) NBOR,IFTCON,IFPF
         49 FORMAT(1H1,1X,*NO OF BORES REQUIRED              *,I3,/,1X,
           1* CONSTANT TIME STEP REQUIRED  *,I1,/,1X,
           2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
           3* ( NO = 0 , YES = 1 )*/)
      C
            READ(8,*) IDTYPE,NGRAPH,IFSEL
            WRITE(6,510) IDTYPE,NGRAPH,IFSEL
        510 FORMAT(* IDTYPE =*,I4,*   NGRAPH =*,I4,*   IFSEL =*,I4)
            READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
            WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
         45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X
           2*END DATE       = *,3I2,/)
            READ(8,*) IDS(1)
            WRITE(6,530) IDS(1)
        530 FORMAT(1X,* FIRST BORE =*,I10)
            IF(IFSEL.LT.1) GOTO 550
            READ(8,*) (IDS(I),I=2,NBOR)
            WRITE(6,540) (I,IDS(I),I=2,NBOR)
        540 FORMAT(6X,I4,I10,/)
        550 CONTINUE
            IF(IFTCON-1) 41,42,39
         41 READ(8,*) DELTAT,DELTAV
            WRITE(6,52) DELTAT,DELTAV
         52 FORMAT(1X,* DELTAT         = *,F6.3,/,1X,
           1* DELTAV         = *,F6.3,/)
            GOTO 43
         42 READ(8,*) TINT,STIME
            WRITE(6,53) TINT,STIME
         53 FORMAT(1X,* TINT         =*,F6.3,/,1X,
           1* STIME        =*,F6.3,/)
            GOTO 43
         39 READ(8,*) DELTAT,DELTAV,TINT,STIME
            WRITE(6,52) DELTAT,DELTAV
            WRITE(6,53) TINT,STIME
      C    NOTE: TINT MUST BE GREATER THAN DELTAT
         43 CONTINUE
            TS=DECDAY(ID1,IM1,IY1,0,0)
            TF=DECDAY(ID2,IM2,IY2,0,0)
            ZX=TS-1.
            IF(ZX.LT.0.)ZX=0.
            IF(NGRAPH.LT.1)GOTO 54
      C
      C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
      C   FOR SINGLE AUTO SCALE SINGLE PLOT USE 0.,0.,0.
      C   SX IN DAYS/CM , SY IN UNITS/CM , ZY IN UNITS
            READ(8,*) SX,SY,ZY
         80 WRITE(6,3021) SX,ZX,SY,ZY
       3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING (0.,0.,0. =AUTOSET)*/,
           12X,*SX ZX SY ZY =*,4F12.4)
         54 CONTINUE
      C   LOOP THROUGH BORES
            DO 1000 NB=1,NBOR
      C   INITIALIZE WORKING ARRAYS
            DO 10 I=1,NNMAX
```

```
115               TN(I)=VN(I)=0.
                10 CONTINUE
                   DO 20 I=1,NMAX
                   T(I)=V(I)=0.
                20 CONTINUE
120             C   READ IN 1ST RECORD AND HUNT FOR FIRST REQUIRED BORE
                C   ECHO PRINT RAW DATA
                   TP=0.
                   NN=0
                   WRITE (6,55)
125                55 FORMAT(3X,*STATION   DATE  TIME  DAY NO.      VALUE*/)
                   IF(IFSEL.EQ.0 .AND. NB.GT.1) GOTO 40
                   I=1
                   46 READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
                   IF(EOF(5)) 48,47
130                48 STOP
                   47 IF(ISN-IDS(NB)) 46,51,105
                   40 I=0
                C   READ SUBSEQUENT RECORDS
                   50 I=I+1
135                   READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
                      IF(EOF(5)) 105,60
                   60 IF(I.EQ.1 .AND. IFSEL.EQ.0) IDS(NB)=ISN
                   51 T(I)=DECDAY(ID,IM,IY,IH,MIN)
                      IF(T(I).LT.TP) GOTO 105
140                   WRITE(6,90) ISN,ID,IM,IY,IH,MIN,T(I),V(I)
                   90 FORMAT(1X,I10,1X,3I2,1X,2I2,2X,F8.3,2X,F8.3)
                      NN=NN+1
                      IF(NN.GT.NMAX) STOP
                      TP=T(I)
145                   GOTO 50
                  100 FORMAT(I10,1X,3I2,1X,2I2,9X,F8.3)
                  105 CONTINUE
                      BACKSPACE 5
                      WRITE(6,120) NN
150                120 FORMAT(I8,* RECORDS READ*)
                      ISTNN=IDS(NB)
                      IF(IDTYPE.EQ.-2) GOTO 350
                C   LOCATE STARTING DATE
                      IF(IFTCON.EQ.1) GOTO 3000
155             C  NON CONSTANT TIME INTERVAL ALLOWED
                      DO 110 I=1,NN
                      IF(T(I)-TS)110,140,140
                  110 CONTINUE
                      WRITE(6,4030)
160              4030 FORMAT(*   START DATE IS AFTER LAST RECORD *)
                      STOP
                  140 IS=I
                      TN(1)=T(IS)
                      VN(1)=V(IS)
165                   JS=IS+1
                C   DELETE ITEMS OF DATA WHERE CHANGES IN TIME AND VALUE
                C   ARE LESS THAN DELTAT AND DELTAV RESPECTIVELY
                C   ALSO STOP AT SPECIFIED FINISH DATE
                      J=1
170                   DELDAY=DELTAT/24.
                      DO 150 I=JS,NN
```

```
      DT=T(I)-T(IS)
      IF(DT.GE.DELDAY)GOTO 180
      DV=ABS(V(I)-V(IS))
175   IF(DV.GE.DELTAV)GOTO 180
      IF(I.EQ.NN) GOTO 180
      TST=ABS(V(I+1)-V(I))
      IF(TST.GE.DELTAV) GOTO 180
      GOTO 150
180    180 IS=I
      J=J+1
      TN(J)=T(IS)
      VN(J)=V(IS)
185    150 CONTINUE
      NJ=J
      GOTO 4000
C CONSTANT TIME INTERVAL IS REQUIRED
C
 3000 TI=STIME/24.
190   TIN=TINT/24.
      SPAN=2.*TIN
      DO 3110 I=1,NN
      IF(T(I)-TI-TS) 3110,3140,3140
 3110 CONTINUE
195  3140 JS=I-1
      NJUMP=0
      IF(JS.NE.0)GOTO 3145
      JS=1
      TJUMP=TI+TS
200  3148 TJUMP=TJUMP+TIN
      NJUMP=NJUMP+1
      IF(TJUMP.GE.T(1)) GOTO 3145
      GOTO 3148
 3145 J=1
205   DO 3150 I=JS,NN
      IF(T(I)-TF-TI) 3160,3160,3170
 3160 CONTINUE
      TN(J)=(NJUMP+J-1)*TIN+TI+TS
      IF(T(I).GT.TN(J)) GOTO 3180
210   GOTO 3150
 3180 IF(I.GT.1) GOTO 3190
      TL=T(1)
      VL=VN(1)=V(1)
      GOTO 3150
215  3190 TL=T(I-1)
      VL=V(I-1)
      IF(T(I)-TL) 3150,3150,3195
 3195 VN(J)=VL+(V(I)-VL)*(TN(J)-TL)/(T(I)-TL)
      IF((T(I)-TL).LE.SPAN) GOTO 3198
220   VN(J)=-2.
      IF((INT(T(I))).EQ.(INT(TN(J)))) VN(J)=V(I)
 3198 J=J+1
      GOTO 3160
 3150 CONTINUE
225  3170 NJ=J-1
 4000 CONTINUE
      WRITE(6,4020) NN,NJ
 4020 FORMAT(I6,* RECORDS REDUCED TO *,I6,* RECORDS FOR PLOTTING*)
```

```
      C   ITEMS FOR PLOTTING NOW CORRESPOND TO (TN(J),VN(J)),J=1,...,NJ
230   C   MANIPULATE ACCORDING TO DATA TYPE REQUIRED
          IF(IDTYPE) 300,200,400
        200 CONTINUE
          GOTO 900
      C   BORE DATA -IDTYPE=-2 SMOOTHING ESTIMATES FOR GL DATA
235   C                      =-1 CONVERT GL TO AHD
        300 CONTINUE
          IF(IDTYPE.EQ.-2) GOTO 350
      C   CONVERT GL OF BORE TO AHD
      C   DEFAULT REF EL = 100.
240       AZ=AHD(ISTNN)
          WRITE(6,320) ISTNN,AZ
        320 FORMAT(/,2X,*REF ELEVATION FOR BORE *,I10,* = *,F7.3,
         1* METRES AHD*/)
          DO 310 I=1,NJ
245       VN(I)=AZ-VN(I)
        310 CONTINUE
          GOTO 900
      C   SMOOTHING ESTIMATES FOR BORE GL DATA
        350 CONTINUE
250       CALL MINMAX(V,NN,VMIN,VMAX)
          WRITE(6,372) VMIN,VMAX
        372 FORMAT(* VMIN,VMAX= *,2F10.3)
          IF(VMIN.LT.99.) GOTO 370
          WRITE(6,375) ISN
255     375 FORMAT(1X,* BORE*,I10,* DRY THROUGHOUT*)
          GOTO 1000
        370 DO 380 I=1,NN
          VN(I)=V(I)
          IF(VN(I).GT.99.) VN(I)=VMIN
260     380 TN(I)=T(I)
          IF(NGRAPH.EQ.0) GOTO 390
          IF(VMAX.GE.99.) CALL MINMAX(VN,NN,VMIN,VMAX)
          WRITE(6,372) VMIN,VMAX
          CALL SFOWN(VMIN,VMAX,10.,SY,ZY)
265       IF(VMIN.EQ.VMAX) GOTO 1000
          CALL PLOT25
          CALLXLIMIT(30.)
          WRITE(6,385) SY,ZY
        385 FORMAT(* SY,ZY =*,2F10.3)
270     390 CALL SMOOTH(NB,NN,NGRAPH,SX,SY,ZX,ZY,IFPF)
          GOTO 2500
        400 CONTINUE
      C   PRINT INPUT DATA AND DATA TO BE PLOTTED
      C
275     898 CONTINUE
          DO 899 I=1,NJ
          TN(I)=T(I)
          VN(I)=V(I)
        899 CONTINUE
280   C   PRINT INPUT DATA AND DATA TO BE PLOTTED
      C
        900 CONTINUE
          DO 2420 I=1,NJ
          IF(IFTCON.EQ.1 .AND. VN(I).LT.-0.01) GOTO 2420
285       NDAY=INT(TN(I))
```

```
                HRS=(TN(I)-FLOAT(NDAY))*24.
                IH=INT(HRS)
                TMIN=(HRS-FLOAT(IH))*60.+0.5
                MIN=INT(TMIN)
290             CALL DATE(ID,IM,IY,NDAY)
                WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I)
                IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I)
           2420 CONTINUE
           2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
295        2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
           2500 CONTINUE
           1000 CONTINUE
                IF(NGRAPH.GT.0) NG=NBOR
                WRITE(6,2005) NG
300        2005 FORMAT(/,2X,I2,* GRAPH/S TO BE PLOTTED*/)
                STOP
                END
```

```
      PROGRAM BAHD(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,TAPE7,
     1PLOT,TAPE9)
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
      COMMON IDS(140)
      DIMENSION T(1280),V(1280)
C     ------------------------------------------------------------
C     BAHD CONTROL PARAMETERS:
C     NBOR,IFTCON,IFPF
C     IDTYPE,NGRAPH,IFSEL
C     NBOR   = NUMBER OF BORES TO BE PROCESSED
C     IFTCON= 0 IRREGULAR TIME INTERVALS (AS PER DATA)
C           = 1 REGULAR TIME INTERVALS (BUT NOT FOR RIVER DISCHARGE)
C     IFPF   = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C           = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C     SPECIFICATIONS ARE -
C      1) START AND FINISH DATES (INCLUSIVE)  (ID1,IM1,IY1,ID2,IM2,IY2)
C      2) MINIMUM TIME INTERVAL BETWEEN VALUES PLOTTED - DELTAT
C           IN HOURS  DEFAULT=0.
C      3) OPTION FOR OVER-RIDING DELTAT FOR CHANGES IN DATA VALUE
C           EXCEEDING SPECIFIED VALUE - DELTAV   DEFAULT=100.
C      4) DATA TYPE FOR PLOTTING (IFTCON = 0 )
C           IDTYPE=0  PLOT OF RAW DATA FILE (SUBJECT TO DELTAT,DELTAV)
C           IDTYPE=-1 CONVERTS BORE GL'S TO AHD VALUES
C           IDTYPE=-2 GIVE SMOOTHING ESTIMATES FOR GL DATA
C     BAHD ALLOWS ANY NUMBER OF PLOTS ON ONE PAIR OF AXES
C     INPUT QUEUE:
C      NBOR,IFTCON,IFPF
C      IDTYPE,NGRAPH,IFSEL
C      ID1,IM1,IY1,ID2,IM2,IY2
C      IDS(1) = ID OF FIRST BORE FOR PROCESSING
C       IDS(2 TO NBOR)  -  IF IFSEL =1
C       DELTAT,DELTAV  -  IF IFTCON=0
C       TINT,STIME     -   "    "  =1
C       ALL 4          -  IF IFTCON=2
C       SCALES (SX,SY,ZY)  " NGRAPH=1
C
C     MOST USEFUL COMBINATION
C     IDTYPE  IFTCON  NGRAPH   IFPF        APPLICATION
C       2       0       0       1          CONVERTS GROUNDWATER DEPTH FILE
C                                          (GLFILE) TO GROUNDWATER ELEVATION
C                                          FILE (AHDFILE).
C
C     BAHD USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C     DATE  DECDAY  GRF  IREF  READREF  SETUP.
C
C-----------------------------------------------------------------
C     INITIALIZE
        NMAX=1280
        DELTAT=0.
        DELTAV=100.
        NN=0
        I=0
        NG=0
C     READ IN CONTROL DATA
        READ(8,*) NBOR,IFTCON,IFPF
        WRITE(6,49) NBOR,IFTCON,IFPF
```

```
     49 FORMAT(1H1,1X,*NO OF BORES REQUIRED              *,I3,/,1X,
       1* CONSTANT TIME STEP REQUIRED  *,I1,/,1X,
       2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
       3* ( NO = 0 , YES = 1 )*/)
C
      READ(8,*) IDTYPE,NGRAPH,IFSEL
      WRITE(6,510) IDTYPE,NGRAPH,IFSEL
  510 FORMAT(* IDTYPE =*,I4,*   NGRAPH =*,I4,*   IFSEL =*,I4)
      READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
      WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
   45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X
     2*END DATE      = *,3I2,/)
      READ(8,*) IDS(1)
      WRITE(6,530) IDS(1)
  530 FORMAT(1X,* FIRST BORE =*,I10)
      IF(IFSEL.LT.1) GOTO 550
      READ(8,*) (IDS(I),I=2,NBOR)
      WRITE(6,540) (I,IDS(I),I=2,NBOR)
  540 FORMAT(6X,I4,I10,/)
  550 CONTINUE
      IF(IFTCON-1) 41,42,39
   41 READ(8,*) DELTAT,DELTAV
      WRITE(6,52) DELTAT,DELTAV
   52 FORMAT(1X,* DELTAT         = *,F6.3,/,1X,
     1* DELTAV         = *,F6.3,/)
      GOTO 43
   42 READ(8,*) TINT,STIME
      WRITE(6,53) TINT,STIME
   53 FORMAT(1X,* TINT          =*,F6.3,/,1X,
     1* STIME         =*,F6.3,/)
      GOTO 43
   39 READ(8,*) DELTAT,DELTAV,TINT,STIME
      WRITE(6,52) DELTAT,DELTAV
      WRITE(6,53) TINT,STIME
C    NOTE: TINT MUST BE GREATER THAN DELTAT
   43 CONTINUE
      TS=DECDAY(ID1,IM1,IY1,0,0)
      TF=DECDAY(ID2,IM2,IY2,0,0)
      ZX=TS-1.
      IF(ZX.LT.0.)ZX=0.
      IF(NGRAPH.LT.1)GOTO 54
C
C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
C   SX IN DAYS/CM , SY IN UNITS/CM , ZY IN UNITS
      READ(8,*) SX,SY,ZY
      WRITE(6,3021) SX,ZX,SY,ZY
 3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
     12X,*SX ZX SY ZY =*,4F12.4)
      CALL SETUP(SX,ZX,SY,ZY)
      CALL READREF
   54 CONTINUE
C   LOOP THROUGH BORES
      DO 1000 NB=1,NBOR
C   INITIALIZE WORKING ARRAYS
      DO 20 I=1,NMAX
      T(I)=V(I)=0.
   20 CONTINUE
```

```
      C   READ IN 1ST RECORD AND HUNT FOR FIRST REQUIRED BORE
      C   ECHO PRINT RAW DATA
            TP=0.
            VP=0.
            NN=0
            IF(NGRAPH.EQ.0) GOTO 30
            TSYM=-400.
            GAPMAX=50.
         30 CONTINUE
            WRITE (6,55)
         55 FORMAT(3X,*STATION   DATE  TIME  DAY NO.      VALUE*/)
            IF(IFSEL.EQ.0 .AND. NB.GT.1) GOTO 40
            I=1
         46 READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
            IF(EOF(5)) 48,47
         48 STOP
         47 IF(ISN-IDS(NB)) 46,51,105
         40 I=0
      C   READ SUBSEQUENT RECORDS
         50 I=I+1
            READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
            IF(EOF(5)) 105,60
         60 IF(I.EQ.1 .AND. IFSEL.EQ.0) IDS(NB)=ISN
         51 T(I)=DECDAY(ID,IM,IY,IH,MIN)
            IF(I.EQ.1 .AND. NGRAPH.NE.0) CALL GRF(NB)
            IF(T(I).LT.TP) GOTO 105
            WRITE(6,90) ISN,ID,IM,IY,IH,MIN,T(I),V(I)
         90 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
            NN=NN+1
            IF(NN.GT.NMAX) STOP
            GAP=T(I)-TP
            IF(NGRAPH.EQ.0) GOTO 230
      C   PLOT LINE IF GAP IN DATA NOT EXCESSIVE
            GAP=T(I)-TP
            Y=17.8-(T(I)-ZX)/SX
            X=6.+(V(I)-ZY)/SY
            IF(V(I).GT.99.) X=6.+FLOAT(NB)*0.2
            IF(GAP.GT.GAPMAX) GOTO 210
            IF(I.EQ.1) GOTO 210
            IF(V(I).GT.99.) GOTO 210
            IF(VP.GT.99.) GOTO 210
            IF((T(I)-TSYM).LT.365.) GOTO 220
            CALL SYMBOL(X,Y,0.2,NB,-90.,-2)
            TSYM=T(I)
            TP=T(I)
            VP=V(I)
            GOTO 50
         220 CALL PLOT(X,Y,2)
         230 TP=T(I)
            VP=V(I)
            GOTO 50
      C   GAP IN RECORD.  DO NOT CONNECT POINTS.
         210 CALL SYMBOL(X,Y,0.2,NB,-90.,-1)
            TSYM=T(I)
            TP=T(I)
            VP=V(I)
            GOTO 50
```

```
      100 FORMAT(I10,1X,3I2,1X,2I2,9X,F8.3)
      105 CONTINUE
          IF(NGRAPH.EQ.0) GOTO 110
          IF(VP.GT.99.) GOTO 110
          XF=6.+(VP-ZY)/SY
          YF=17.8-(TP-ZX)/SX
          CALL SYMBOL(XF,YF,0.2,NB,-90.,-1)
      110 CONTINUE
          BACKSPACE 5
          WRITE(6,120) NN
      120 FORMAT(I8,* RECORDS READ*)
          ISTNN=IDS(NB)
          NJ=NN
C     PRINT INPUT DATA AND DATA TO BE PLOTTED
C
          IF(IDTYPE.EQ.0) GOTO 1000
          IF(IDTYPE.NE.2) GOTO 1500
          IF(NB.EQ.1) CALL READREF
          IRF=IREF(ISTNN)
          HT=EL(IRF)
          WRITE(6,1220) NB,IRF,ISTNN,HT
     1220 FORMAT(* CY NO=*,I6,*  REF NO=*,I6,*  BORE =*,I10,*  EL=*,F6.2)
          DO 1200 I=1,NJ
          IF(V(I).GT.99.) GOTO 1200
          V(I)=HT-V(I)
     1200 CONTINUE
     1500 CONTINUE
          DO 2420 I=1,NJ
          NDAY=INT(T(I))
          HRS=(T(I)-FLOAT(NDAY))*24.
          IH=INT(HRS)
          TMIN=(HRS-FLOAT(IH))*60.+0.5
          MIN=INT(TMIN)
          CALL DATE(ID,IM,IY,NDAY)
          WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
          IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
     2420 CONTINUE
     2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
     2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
     1000 CONTINUE
          IF(NGRAPH.GT.0) NG=NBOR
          WRITE(6,2005) NG
     2005 FORMAT(/,2X,I2,* GRAPH/S TO BE PLOTTED*/)
          STOP
          END
```

```
      PROGRAM BPLOT(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,TAPE7,
     1PLOT,TAPE9)
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
      COMMON IDS(140)
      DIMENSION T(1280),V(1280)
C     ------------------------------------------------------------
C     BPLOT CONTROL PARAMETERS:
C     NBOR,IFTCON,IFPF
C     IDTYPE,NGRAPH,IFSEL
C     NBOR  = NUMBER OF BORES TO BE PROCESSED
C     IFTCON= 0 IRREGULAR TIME INTERVALS (AS PER DATA)
C           = 1 REGULAR TIME INTERVALS (BUT NOT FOR RIVER DISCHARGE)
C     IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C           = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C     BPLOT SELECTS DATA FOR PLOTTING FROM A BORE DATA
C     FILE  : SPECIFICATIONS ARE -
C      1) START AND FINISH DATES (INCLUSIVE)  (ID1,IM1,IY1,ID2,IM2,IY2)
C      2) MINIMUM TIME INTERVAL BETWEEN VALUES PLOTTED - DELTAT
C           IN HOURS  DEFAULT=0.
C      3) OPTION FOR OVER-RIDING DELTAT FOR CHANGES IN DATA VALUE
C           EXCEEDING SPECIFIED VALUE - DELTAV   DEFAULT=100.
C      4) DATA TYPE FOR PLOTTING (IFTCON = 0 )
C           IDTYPE=0  PLOT OF RAW DATA FILE (SUBJECT TO DELTAT,DELTAV)
C           IDTYPE=-1 CONVERTS BORE GL'S TO AHD VALUES
C           IDTYPE=-2 GIVE SMOOTHING ESTIMATES FOR GL DATA
C     BPLOT ALLOWS ANY NUMBER OF PLOTS ON ONE PAIR OF AXES
C     INPUT QUEUE:
C      NBOR,IFTCON,IFPF
C      IDTYPE,NGRAPH,IFSEL
C      ID1,IM1,IY1,ID2,IM2,IY2
C      IDS(1) = ID OF FIRST BORE FOR PROCESSING
C       IDS(2 TO NBOR)  -  IF IFSEL =1
C       DELTAT,DELTAV   -  IF IFTCON=0
C       TINT,STIME      -   "     "  =1
C       ALL 4           -  IF IFTCON=2
C       SCALES (SX,SY,ZY)  " NGRAPH=1
C
C     MOST USEFUL COMBINATION
C     IDTYPE  IFTCON  NGRAPH  IFPF      APPLICATIONS
C       0       0       1      0        PLOTS GROUNDWATER ELEVATIONS
C                                       ON STANDARD A4 SIZE CALCOMP
C                                       GRAPHS.
C
C     BPLOT USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C     DATE  DECDAY  GRF  IREF  READREF  SETUP.
C
C-----------------------------------------------------------------
C     INITIALIZE
        NMAX=1280
        DELTAT=0.
        DELTAV=100.
        NN=0
        I=0
        NG=0
C     READ IN CONTROL DATA
        READ(8,*) NBOR,IFTCON,IFPF
```

```
      WRITE(6,49) NBOR,IFTCON,IFPF
   49 FORMAT(1H1,1X,*NO OF BORES REQUIRED              *,I3,/,1X,
     1* CONSTANT TIME STEP REQUIRED  *,I1,/,1X,
     2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
     3* ( NO = 0 , YES = 1 )*/)
C
      READ(8,*) IDTYPE,NGRAPH,IFSEL
      WRITE(6,510) IDTYPE,NGRAPH,IFSEL
  510 FORMAT(* IDTYPE =*,I4,*   NGRAPH =*,I4,*   IFSEL =*,I4)
      READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
      WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
   45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X
     2*END DATE       = *,3I2,/)
      READ(8,*) IDS(1)
      WRITE(6,530) IDS(1)
  530 FORMAT(1X,* FIRST BORE =*,I10)
      IF(IFSEL.LT.1) GOTO 550
      READ(8,*) (IDS(I),I=2,NBOR)
      WRITE(6,540) (I,IDS(I),I=2,NBOR)
  540 FORMAT(6X,I4,I10,/)
  550 CONTINUE
      IF(IFTCON-1) 41,42,39
   41 READ(8,*) DELTAT,DELTAV
      WRITE(6,52) DELTAT,DELTAV
   52 FORMAT(1X,* DELTAT         = *,F6.3,/,1X,
     1* DELTAV         = *,F6.3,/)
      GOTO 43
   42 READ(8,*) TINT,STIME
      WRITE(6,53) TINT,STIME
   53 FORMAT(1X,* TINT         =*,F6.3,/,1X,
     1* STIME        =*,F6.3,/)
      GOTO 43
   39 READ(8,*) DELTAT,DELTAV,TINT,STIME
      WRITE(6,52) DELTAT,DELTAV
      WRITE(6,53) TINT,STIME
C     NOTE: TINT MUST BE GREATER THAN DELTAT
   43 CONTINUE
      TS=DECDAY(ID1,IM1,IY1,0,0)
      TF=DECDAY(ID2,IM2,IY2,0,0)
      ZX=TS-1.
      IF(ZX.LT.0.)ZX=0.
      IF(NGRAPH.LT.1)GOTO 54
C
C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
C   SX IN DAYS/CM , SY IN UNITS/CM , ZY IN UNITS
      READ(8,*) SX,SY,ZY
      WRITE(6,3021) SX,ZX,SY,ZY
 3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
     12X,*SX ZX SY ZY =*,4F12.4)
      CALL SETUP(SX,ZX,SY,ZY)
      CALL READREF
   54 CONTINUE
C   LOOP THROUGH BORES
      DO 1000 NB=1,NBOR
C   INITIALIZE WORKING ARRAYS
      DO 20 I=1,NMAX
      T(I)=V(I)=0.
```

```
115            20 CONTINUE
           C    READ IN 1ST RECORD AND HUNT FOR FIRST REQUIRED BOR
           C    ECHO PRINT RAW DATA
                TP=0.
                VP=0.
120             NN=0
                IF(NGRAPH.EQ.0) GOTO 30
                TSYM=-400.
                GAPMAX=50.
             30 CONTINUE
125             WRITE (6,55)
             55 FORMAT(3X,*STATION   DATE  TIME  DAY NO.      VALUE*/)
                IF(IFSEL.EQ.0 .AND. NB.GT.1) GOTO 40
                I=1
             46 READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
130             IF(EOF(5)) 48,47
             48 STOP
             47 IF(ISN-IDS(NB)) 46,51,105
             40 I=0
           C    READ SUBSEQUENT RECORDS
135          50 I=I+1
                READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
                IF(EOF(5)) 105,60
             60 IF(I.EQ.1 .AND. IFSEL.EQ.0) IDS(NB)=ISN
             51 T(I)=DECDAY(ID,IM,IY,IH,MIN)
140             IF(I.EQ.1 .AND. NGRAPH.NE.0) CALL GRF(NB)
                IF(T(I).LT.TP) GOTO 105
                WRITE(6,90) ISN,ID,IM,IY,IH,MIN,T(I),V(I)
             90 FORMAT(1X,I10,1X,3I2,1X,2I2,2X,F8.3,2X,F8.3)
                NN=NN+1
145             IF(NN.GT.NMAX) STOP
                GAP=T(I)-TP
                IF(NGRAPH.EQ.0) GOTO 230
           C    PLOT LINE IF GAP IN DATA NOT EXCESSIVE
                GAP=T(I)-TP
150             Y=17.8-(T(I)-ZX)/SX
                X=6.+(V(I)-ZY)/SY
                IF(V(I).GT.99.) X=6.+FLOAT(NB)*0.2
                IF(GAP.GT.GAPMAX) GOTO 210
                IF(I.EQ.1) GOTO 210
155             IF(V(I).GT.99.) GOTO 210
                IF(VP.GT.99.) GOTO 210
                IF((T(I)-TSYM).LT.365.) GOTO 220
                CALL SYMBOL(X,Y,0.2,NB,-90.,-2)
                TSYM=T(I)
160             TP=T(I)
                VP=V(I)
                GOTO 50
            220 CALL PLOT(X,Y,2)
            230 TP=T(I)
165             VP=V(I)
                GOTO 50
           C    GAP IN RECORD.  DO NOT CONNECT POINTS.
            210 CALL SYMBOL(X,Y,0.2,NB,-90.,-1)
                TSYM=T(I)
170             TP=T(I)
                VP=V(I)
```

```
            GOTO 50
          100 FORMAT(I10,1X,3I2,1X,2I2,9X,F8.3)
          105 CONTINUE
175           IF(NGRAPH.EQ.0) GOTO 110
              IF(VP.GT.99.) GOTO 110
              XF=6.+(VP-ZY)/SY
              YF=17.8-(TP-ZX)/SX
              CALL SYMBOL(XF,YF,0.2,NB,-90.,-1)
180       110 CONTINUE
              BACKSPACE 5
              WRITE(6,120) NN
          120 FORMAT(I8,* RECORDS READ*)
              ISTNN=IDS(NB)
185           NJ=NN
        C   PRINT INPUT DATA AND DATA TO BE PLOTTED
        C
              IF(IDTYPE.EQ.0) GOTO 1000
              DO 2420 I=1,NJ
190           NDAY=INT(T(I))
              HRS=(T(I)-FLOAT(NDAY))*24.
              IH=INT(HRS)
              TMIN=(HRS-FLOAT(IH))*60.+0.5
              MIN=INT(TMIN)
195           CALL DATE(ID,IM,IY,NDAY)
              WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
              IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
         2420 CONTINUE
         2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
200      2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
         1000 CONTINUE
              IF(NGRAPH.GT.0) NG=NBOR
              WRITE(6,2005) NG
         2005 FORMAT(/,2X,I2,* GRAPH/S TO BE PLOTTED*/)
205           STOP
              END
```

```
      PROGRAM BSTAT(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,TAPE7,
     1PLOT,TAPE9)
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
      COMMON IDS(140)
      DIMENSION T(1280),V(1280)
C     ------------------------------------------------------------
C    BSTAT CONTROL PARAMETERS:
C    NBOR,IFTCON,IFPF
C    IDTYPE,NGRAPH,IFSEL
C    NBOR  = NUMBER OF BORES TO BE PROCESSED
C    IFTCON= 0 IRREGULAR TIME INTERVALS (AS PER DATA)
C          = 1 REGULAR TIME INTERVALS (BUT NOT FOR RIVER DISCHARGE)
C    IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C          = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C    BSTAT COMPUTES STATISTICS OF GROUNDWATER LEVEL OBSERVATIONS
C    AT ALL OBS BORES AND TABULATES THESE
C
C    FILE  : SPECIFICATIONS ARE -
C     1) START AND FINISH DATES (INCLUSIVE)  (ID1,IM1,IY1,ID2,IM2,IY2)
C     2) MINIMUM TIME INTERVAL BETWEEN VALUES PLOTTED - DELTAT
C          IN HOURS  DEFAULT=0.
C     3) OPTION FOR OVER-RIDING DELTAT FOR CHANGES IN DATA VALUE
C          EXCEEDING SPECIFIED VALUE - DELTAV   DEFAULT=100.
C     4) DATA TYPE FOR PLOTTING (IFTCON = 0 )
C          IDTYPE=0  PLOT OF RAW DATA FILE (SUBJECT TO DELTAT,DELTAV)
C          IDTYPE=-1 CONVERTS BORE GL'S TO AHD VALUES
C          IDTYPE=-2 GIVE SMOOTHING ESTIMATES FOR GL DATA
C    INPUT QUEUE:
C     NBOR,IFTCON,IFPF
C     IDTYPE,NGRAPH,IFSEL
C     ID1,IM1,IY1,ID2,IM2,IY2
C     IDS(1) = ID OF FIRST BORE FOR PROCESSING
C      IDS(2 TO NBOR)  -  IF IFSEL =1
C      DELTAT,DELTAV  -  IF IFTCON=0
C      TINT,STIME     -   "     "  =1
C      ALL 4          -  IF IFTCON=2
C      SCALES (SX,SY,ZY)  " NGRAPH=1
C
C    MOST USEFUL COMBINATION
C    IDTYPE   IFTCON  NGRAPH          APPLICATION
C      1        0       0             PRINT TABLE OF GROUNDWATER
C                                     OBSERVATION STATISTICS FOR ALL
C                                     OBS BORES.
C
C    BSTAT USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C    DATE  DECDAY  GRF  IREF  MINMAX  READREF  SETUP.
C
C-------------------------------------------------------------------
C    INITIALIZE
       NMAX=1280
       DELTAT=0.
       DELTAV=100.
       NN=0
       I=0
       NG=0
       NRDG=0
```

```
      C   READ IN CONTROL DATA
            READ(8,*) NBOR,IFTCON,IFPF
            WRITE(6,49) NBOR,IFTCON,IFPF
         49 FORMAT(1H1,1X,*NO OF BORES REQUIRED              *,I3,/,1X,
           1* CONSTANT TIME STEP REQUIRED  *,I1,/,1X,
           2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
           3* ( NO = 0 , YES = 1 )*/)
      C
            READ(8,*) IDTYPE,NGRAPH,IFSEL
            WRITE(6,510) IDTYPE,NGRAPH,IFSEL
        510 FORMAT(* IDTYPE =*,I4,*   NGRAPH =*,I4,*   IFSEL =*,I4)
            READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
            WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
         45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X
           2*END DATE      = *,3I2,/)
            READ(8,*) IDS(1)
            WRITE(6,530) IDS(1)
        530 FORMAT(1X,* FIRST BORE =*,I10)
            IF(IFSEL.LT.1) GOTO 550
            READ(8,*) (IDS(I),I=2,NBOR)
            WRITE(6,540) (I,IDS(I),I=2,NBOR)
        540 FORMAT(6X,I4,I10,/)
        550 CONTINUE
            IF(IFTCON-1) 41,42,39
         41 READ(8,*) DELTAT,DELTAV
            WRITE(6,52) DELTAT,DELTAV
         52 FORMAT(1X,* DELTAT        = *,F6.3,/,1X,
           1* DELTAV        = *,F6.3,/)
            GOTO 43
         42 READ(8,*) TINT,STIME
            WRITE(6,53) TINT,STIME
         53 FORMAT(1X,* TINT        =*,F6.3,/,1X,
           1* STIME       =*,F6.3,/)
            GOTO 43
         39 READ(8,*) DELTAT,DELTAV,TINT,STIME
            WRITE(6,52) DELTAT,DELTAV
            WRITE(6,53) TINT,STIME
      C     NOTE: TINT MUST BE GREATER THAN DELTAT
         43 CONTINUE
            TS=DECDAY(ID1,IM1,IY1,0,0)
            TF=DECDAY(ID2,IM2,IY2,0,0)
            ZX=TS-1.
            IF(ZX.LT.0.)ZX=0.
            IF(NGRAPH.LT.1)GOTO 54
      C
      C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
      C   SX IN DAYS/CM , SY IN UNITS/CM , ZY IN UNITS
            READ(8,*) SX,SY,ZY
            WRITE(6,3021) SX,ZX,SY,ZY
       3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
           12X,*SX ZX SY ZY =*,4F12.4)
            CALL SETUP(SX,ZX,SY,ZY)
         54 CONTINUE
            CALL READREF
            IF(IDTYPE.EQ.1) WRITE(6,3600)
       3600 FORMAT(1H1,///14X,*BORE NO  EAST    NORTH REF EL MEAN  SDEV  MIN*,
           1*   MAX  RANGE   N  DRY*,/)
```

```
115          C   LOOP THROUGH BORES
                   DO 1000 NB=1,NBOR
             C   INITIALIZE WORKING ARRAYS
                   DO 20 I=1,NMAX
                   T(I)=V(I)=0.
120             20 CONTINUE
             C   READ IN 1ST RECORD AND HUNT FOR FIRST REQUIRED BORE
             C   ECHO PRINT RAW DATA
                   TP=0.
                   VP=0.
125                NN=0
                   IF(NGRAPH.EQ.0) GOTO 30
                   TSYM=-400.
                   GAPMAX=50.
                30 CONTINUE
130          C      WRITE (6,55)
             C  55 FORMAT(3X,*STATION   DATE  TIME  DAY NO.     VALUE*/)
                   IF(IFSEL.EQ.0 .AND. NB.GT.1) GOTO 40
                   I=1
                46 READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
135                IF(EOF(5)) 48,47
                48 STOP
                47 IF(ISN-IDS(NB)) 46,51,105
                40 I=0
             C   READ SUBSEQUENT RECORDS
140             50 I=I+1
                   READ(5,100) ISN,ID,IM,IY,IH,MIN,V(I)
                   IF(EOF(5)) 105,60
                60 IF(I.EQ.1 .AND. IFSEL.EQ.0) IDS(NB)=ISN
                51 T(I)=DECDAY(ID,IM,IY,IH,MIN)
145                IF(I.EQ.1 .AND. NGRAPH.NE.0) CALL GRF(NB)
                   IF(T(I).LT.TP) GOTO 105
             C      WRITE(6,90) ISN,ID,IM,IY,IH,MIN,T(I),V(I)
             C  90 FORMAT(1X,I10,1X,3I2,1X,2I2,2X,F8.3,2X,F8.3)
                   NN=NN+1
150                IF(NN.GT.NMAX) STOP
                   GAP=T(I)-TP
                   IF(NGRAPH.EQ.0) GOTO 230
                   GAP=T(I)-TP
                   Y=17.8-(T(I)-ZX)/SX
155                X=6.+(V(I)-ZY)/SY
                   IF(V(I).GT.99.) X=6.+FLOAT(NB)*0.2
                   IF(GAP.GT.GAPMAX) GOTO 210
                   IF(I.EQ.1) GOTO 210
                   IF(V(I).GT.99.) GOTO 210
160                IF(VP.GT.99.) GOTO 210
                   IF((T(I)-TSYM).LT.365.) GOTO 220
                   CALL SYMBOL(X,Y,0.2,NB,-90.,-2)
                   TSYM=T(I)
                   TP=T(I)
165                VP=V(I)
                   GOTO 50
                220 CALL PLOT(X,Y,2)
                230 TP=T(I)
                   VP=V(I)
170                GOTO 50
             C   GAP IN RECORD.  DO NOT CONNECT POINTS.
```

```
  210 CALL SYMBOL(X,Y,0.2,NB,-90.,-1)
      TSYM=T(I)
      TP=T(I)
175   VP=V(I)
      GOTO 50
  100 FORMAT(I10,1X,3I2,1X,2I2,9X,F8.3)
  105 CONTINUE
C     WRITE(6,90) ISN,ID,IM,IY,IH,MIN,T(I),V(I)
180   IF(NGRAPH.EQ.0) GOTO 110
      IF(VP.GT.99.) GOTO 110
      XF=6.+(VP-ZY)/SY
      YF=17.8-(TP-ZX)/SX
      CALL SYMBOL(XF,YF,0.2,NB,-90.,-1)
185   110 CONTINUE
      BACKSPACE 5
C     WRITE(6,120) NN
C 120 FORMAT(I8,* RECORDS READ*)
      ISTNN=IDS(NB)
190   NJ=NN
C   PRINT INPUT DATA AND DATA TO BE PLOTTED
C
      IF(IDTYPE.EQ.0) GOTO 1000
      IF(IDTYPE.EQ.1) GOTO 900
195   DO 2420 I=1,NJ
      NDAY=INT(T(I))
      HRS=(T(I)-FLOAT(NDAY))*24.
      IH=INT(HRS)
      TMIN=(HRS-FLOAT(IH))*60.+0.5
200   MIN=INT(TMIN)
      CALL DATE(ID,IM,IY,NDAY)
      WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
      IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,T(I),V(I)
 2420 CONTINUE
205  2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
 2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
      GOTO 1000
  900 CONTINUE
C   PRINT BORE STATISTICS
210   VS=0.
      VS2=0.
      NDRY=0
      SDEV=0.
      DO 3100 I=1,NJ
215   IF(V(I).LT.99.) GOTO 3200
      NDRY=NDRY+1
      GOTO 3100
 3200 VS=VS+V(I)
      VS2=VS2+V(I)*V(I)
220  3100 CONTINUE
      NRDG=NRDG+NJ
      FN=FLOAT(NJ-NDRY)
      VBAR=VS/FN
      IF(FN.LE.1) GOTO 3400
225   VAR=(VS2-VS*VS/FN)/(FN-1.)
      SDEV=SQRT(VAR)
 3400 CALL MINMAX(V,NJ,VMIN,VMAX)
      RNG=VMAX-VMIN
```

```
                RON=0.
230             IF(SDEV.GT.0.) RON=RNG/SDEV
                IQN=IDS(NB)
                IRF=IREF(IQN)
                LE=INT(CE(IRF))
                LN=INT(CN(IRF))
235             IF(NB.EQ.55) WRITE(6,3600)
                IF(NB.EQ.110) WRITE(6,3600)
                WRITE(6,3500) IQN,LE,LN,EL(IRF),VBAR,SDEV,VMIN,VMAX,RNG,
               1NJ,NDRY,RON
           3500 FORMAT(10X,I10,2I8,6F6.2,I5,I3,15X,F6.2)
240        1000 CONTINUE
                WRITE(6,3700) NRDG
           3700 FORMAT(1H1,//,* TOTAL NO OF BORE READINGS =*,I8)
                IF(NGRAPH.GT.0) NG=NBOR
                WRITE(6,2005) NG
245        2005 FORMAT(/,2X,I2,* GRAPH/S TO BE PLOTTED*/)
                STOP
                END
```

```
      PROGRAM BORFL(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5,TAPE7,TAPE9)
      COMMON T(1280),V(1280),VN(140,16),K(300),IR(140),NT(16),
     1NDT(16),W(1280)
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
C     ------------------------------------------------------------
C     BORFL INTERPOLATES BETWEEN OBSERVED GROUNDWATER ELEVATIONS
C     TO FIND GW ELEVATIONS ON SPECIFIED DATES FOR ALL BORES
C     (USED IN PRODUCING MANUAL CONTOUR PLOTS OF THE FREE SURFACE
C     ON THOSE DATES)
C
C     NO CONTROL PARAMETERS TO BE READ IN (ALL SPECIFIED BELOW)
C
C     BORFL USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C     CONST  DATE  IREF  READREF.
C
C     ------------------------------------------------------------
      NTIN=90
      NFEND=0
      MAX=14
      IDTYPE=1
C     DO 1 J=1,16
C     DO 2 I=1,140
C   2 VN(I,J)=0.
C   1 NT(J)=NDT(J)=0
      DO 4 I=1,140
    4 K(I)=0
      CALL READREF
C     SPECIFY DATES BY DAY NUMBER
      NT(1)=90
      NT(2)=181
      NT(3)=273
      NT(4)=365
      NT(5)=456
      NT(6)=547
      NT(7)=639
      NT(8)=731
      NT(9)=821
      NT(10)=912
      NT(11)=1004
      NT(12)=1096
      NT(13)=1186
      NT(14)=1277
      NC=0
    3 NC=NC+1
C   READ BORE DATA
      TP=0.
      J=1
    5 READ(5,400) ISN,ID,IM,IY,T(J),V(J)
  400 FORMAT(I10,1X,3I2,6X,2F8.3)
      IF(EOF(5)) 7,6
    6 IF(J.EQ.1) IR(NC)=ISN
      IF(T(J).LT.TP) GOTO 8
C     WRITE(6,500) ISN,ID,IM,IY,T(J),V(J)
C 500 FORMAT(1X,I10,1X,3I2,6X,2F8.3)
      TP=T(J)
      J=J+1
      GOTO 5
```

```
      7 IFLEND=1
      8 NN=J-1
60      BACKSPACE 5
C     FIND AHD ON SELECTED DAYS
        CALL CONST(NTIN,NFEND,NC,MAX,NN,NJ,ITYPE)
        KS=K(2*NC-1)
        KSM=KS-1
65      NIT=NJ-KSM
        WRITE(6,*) NN,# ITEMS REDUCED TO #,NIT,# FIRST,LAST = #,KS,NJ
        WRITE(6,*) # FIRST,LAST =#,NT(KS),NT(NJ),VN(NC,KS),VN(NC,NJ)
        IF(IDTYPE.EQ.1) WRITE(6,600) (VN(NC,I),I=1,MAX)
  600 FORMAT(1X,16F6.2)
70      IF(IFLEND.EQ.1) GOTO 100
        GOTO 3
C     FINISHED READING IN DATA
  100 CONTINUE
        WRITE(6,700)
75      WRITE(6,110) (NT(I),I=1,MAX)
        WRITE(6,115) (NDT(I),I=1,MAX)
  110 FORMAT(30X,14I7)
  115 FORMAT(31X,14I7)
        WRITE(7,110) (NT(I),I=1,MAX)
80      WRITE(7,115) (NDT(I),I=1,MAX)
        DO 140 I=1,NC
        M=IREF(IR(I))
        IF(I.NE.55 .AND. I.NE.110) GOTO 150
        WRITE(6,700)
85  700 FORMAT(1H1)
        WRITE(6,110) (NT(N),N=1,14)
        WRITE(6,115) (NDT(N),N=1,14)
  150 CONTINUE
        WRITE(6,130) IVS(M),EL(M),CE(M),CN(M),(VN(I,J),J=1,MAX)
90      WRITE(7,130) IVS(M),EL(M),CE(M),CN(M),(VN(I,J),J=1,MAX)
  130 FORMAT(1X,I10,F6.2,2F7.0,14F7.2)
        TST=FLOAT(I)/10.
        NST=INT(TST)
        QST=FLOAT(NST)
95      IF(QST.EQ.TST) WRITE(6,750)
  750 FORMAT(1X)
  140 CONTINUE
  200 CONTINUE
        STOP
100     END
```

```
      PROGRAM QPL(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,
     1PLOT,TAPE7)
      COMMON TN(1300),VN(1300),VMN(1300),VMX(1300),TITLE(16)
      DIMENSION T(200),V(200),ISTN(8)
      DIMENSION DM(12),MN(12),MT(12),DM3(4),VMNM(12),VMXM(12)
      DATA MT/0,31,59,90,120,151,181,212,243,273,304,334/
      DATA MN/31,28,31,30,31,30,31,31,30,31,30,31/
C     ------------------------------------------------------------
C     QPL CALCULATES AND PLOTS DAILY RIVER DISCHARGE
C
C     QPL CONTROL PARAMETERS:
C     NSTN,IFPLT,IFPF
C     NSTN  = NUMBER OF STATIONS
C     IFPLT = 0 PLOT NOT REQUIRED
C           = 1 PLOT IS REQUIRED
C           = 2 PLOT AND INPUT DATA PRINTOUT REQD
C     IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C           = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C     QPL SELECTS DATA FOR PLOTTING FROM A RAW DATA
C     FILE  : SPECIFICATIONS ARE -
C     INPUT QUEUE:
C      NSTN,IFPLT,IFPF
C      ID1,IM1,IY1,ID2,IM2,IY2  (FIRST AND LAST DATES REQD)
C      ISTN(I),TITLE(I)  (I5,5X,2A10,/)  (I=1,...,NSTN)
C     (SX,SY,ZY) (* AXIS SCALES AND Y-ZERO)  (IFPLT = 1 ONLY)
C
C     QPL USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C     DATE  DECDAY  FLORATE  GRAPH  MINMAXL  RATING  SETUP.
C
C-----------------------------------------------------------------------
      NNMAX=200
      NWMAX=1300
      NDT=1277
      NG=0
      TIN=1.
      STIME=9./24.
C
C     READ IN CONTROL DATA
      READ(8,*) NSTN,IFPLT,IFPF
      WRITE(6,51) NSTN,IFPLT,IFPF
   51 FORMAT(1H1,1X,*NO OF STNS REQUIRED              *,I1,/,1X,
     1* PLOT OF RESULTS REQUIRED  *,I1,/,1X,
     2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
     3* ( NO = 0 , YES = 1 )*/)
C
      READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
      WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
   45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X,
     2*END DATE       = *,3I2,/)
      WRITE(6,53) TIN,STIME
   53 FORMAT(1X,* TIN          =*,F6.3,/,1X,
     1* STIME        =*,F6.3,/)
      TS=DECDAY(ID1,IM1,IY1,0,0)
      TF=DECDAY(ID2,IM2,IY2,24,0)
      ZX=0.
      NCT=20
```

```
      DO 40 I=1,NSTN
      J=2*I-1
60    K=J+1
      READ(8,2004) ISTN(I),TITLE(J),TITLE(K)
 2004 FORMAT(I5,5X,2A10)
   40 WRITE(6,2006) ISTN(I),TITLE(J),TITLE(K)
 2006 FORMAT(1X,I5,5X,2A10)
65    IF(IFPLT.LT.1) GOTO 50
C
C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
      READ(8,*) SX,SY,ZY
      WRITE(6,3021) SX,ZX,SY,ZY
70  3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
     12X,*SX ZX SY ZY =*,4F12.4)
   50 CONTINUE
C   CYCLE THROUGH STATIONS
C
75    DO 1000 NCY=1,NSTN
C
      ISTNN=ISTN(NCY)
      I=2*NCY-1
      TITLE(1)=TITLE(I)
80    TITLE(2)=TITLE(I+1)
      IF(IFPLT.GE.1) CALL SETUP(SX,ZX,SY,ZY,MN)
C   SELECT GAUGING STATION RATING
      CALL RATING(ISTNN,NR,A1,B1,H12,A2,B2,H23,A3,B3)
      WRITE(6,46) ISTNN,NR,H12,A1,B1,H12,H23,A2,B2,H23,A3,B3
85   46 FORMAT(1H1,1X,*RATING FOR STATION *,I2,5X,*NO. OF SEGMENTS IS *,
     1I1,//,5X,* H = 1.000 TO *,F5.3,5X,*A =*,F15.1,9X,*B =*,F8.3,/,
     25X,* H = *,F5.3,* TO *,F5.3,5X,*A =*,F15.1,9X,*B =*,F8.3,/,
     35X,* H           > *,F5.3,5X,*A =*,F15.1,9X,*B=*,F8.3,//)
C   INITIALIZE WORKING ARRAYS
90    DO 10 I=1,NWMAX
      TN(I)=VN(I)=0.
   10 CONTINUE
      DO 20 I=1,NNMAX
      T(I)=V(I)=0.
95   20 CONTINUE
      TMIN=0.+STIME
      TMAX=TMIN+TIN
      VS=VF=-2.
      KDGOOD=0
100   KOUNT=1
      I=1
      J=0
      ICLOB=0
      ND1=0
105   WRITE (6,55)
   55 FORMAT(3X,*STATION   DATE  TIME  DAY ND.     VALUE*/)
C   READ FIRST RECORD
      READ(5,100) STNN,ID,IM,IY,IH,MIN,VV
      T(1)=DECDAY(ID,IM,IY,IH,MIN)
110   V(1)=FLORATE(A1,B1,H12,A2,B2,H23,A3,B3,VV)
      WRITE(6,90) STNN,ID,IM,IY,IH,MIN,T(I),VV,V(I)
   90 FORMAT(1X,A10,1X,3I2,1X,2I2,3F10.3)
      IF(T(1).LE.TMAX) GOTO 4010
C   FILL TO START OF RECORD WITH -2
```

```
115         C             -2 = NO RECORD
            C             -1 = FLOWRATE EXCEEDS STATION RATING
                  ND1=INT(T(1))
                  J=ND1
                  TMIN=TMIN+TIN*FLOAT(ND1)
120               TMAX=TMIN+TIN
                  DO 4008 KB=1,ND1
                  TN(KB)=FLOAT(KB)+STIME
                  VN(KB)=VMN(KB)=VMX(KB)=-2.
               4008 CONTINUE
125               TST1=(T(1)-TMIN)/TIN
                  IF(TST1.LT.0.25) GOTO 4010
                  ICLOB=2
               4010 S1=V(1)*(T(1)-TMIN)
                  VS=V(1)
130               VMIN=VMAX=V(1)
            C
               4000 I=I+1
                  IF(I.GT.NNMAX) GOTO 9000
                  MISS=0
135               KOUNT=KOUNT+1
            C   READ SUBSEQUENT RECORDS
                  READ(5,100) STNN,ID,IM,IY,IH,MIN,VV
                  IF(EOF(5)) 4030,4020
               4020 T(I)=DECDAY(ID,IM,IY,IH,MIN)
140               IF(T(I).LT.T(I-1)) GOTO 4025
                  V(I)=FLGRATE(A1,B1,H12,A2,B2,H23,A3,B3,VV)
                  IF(IFPLT.EQ.2) WRITE(6,90) STNN,ID,IM,IY,IH,MIN,T(I),VV,V(I)
                  IF(T(I).GT.TMAX) GOTO 4011
                  S1=S1+(T(I)-T(I-1))*(V(I)+V(I-1))*0.5
145               IF(V(I).LT.-0.1 .OR. V(I-1).LT.-0.1) ICLOB=1
                  GOTO 4000
               4011 TSJUMP=TMAX+TIN
                  IF(T(I).GT.TSJUMP) GOTO 4040
                  VF=V(I-1)+(TMAX-T(I-1))*(V(I)-V(I-1))/(T(I)-T(I-1))
150               IF(V(I).LT.-0.1 .OR. V(I-1).LT.-0.1) ICLOB=1
               4044 J=J+1
                  S1=S1+(TMAX-T(I-1))*(VF+V(I-1))*0.5
                  VN(J)=S1*0.0864
                  IF(ICLOB.EQ.1) VN(J)=-1.
155               IF(ICLOB.EQ.2) VN(J)=-2.
                  IF(ICLOB.EQ.0) KDGOOD=KDGOOD+1
                  ICLOB=0
                  TN(J)=TMAX
            C
160               V(I+1)=VS
                  V(I+2)=VF
                  NJ=I+2
                  CALL MINMAXL(V,NJ,VMIN,VMAX)
                  VMN(J)=VMIN
165               VMX(J)=VMAX
                  IF(MISS.EQ.1) GOTO 4046
                  S1=(T(I)-TMAX)*(VF+V(I))*0.5
                  IF(V(I).LT.-0.1 .OR. V(I-1).LT.-0.1) ICLOB=1
                  TMIN=TMAX
170               TMAX=TMAX+TIN
                  T(1)=T(I)
```

```
      V(1)=V(I)
      I=1
      VS=VF
175   GOTO 4000
C   MISSING RECORD
 4040 TST3=(T(I-1)-TMIN)/TIN
      IF(TST3.LT.0.75) GOTO 4045
      MISS=1
180   VF=V(I-1)
      GOTO 4044
C
 4046 TMAX=TMAX+TIN
 4045 J=J+1
185   VN(J)=-2.
      VMN(J)=VMX(J)=-2.
      TN(J)=TMAX
      TMAX=TMAX+TIN
      IF(T(I).GT.TMAX) GOTO 4045
190   TST2=(TMAX-T(I))/TIN
      TMIN=TMAX-TIN
      IF(TST2.GT.0.75) GOTO 4050
      ICLOB=2
C
195  4050 S1=(T(I)-TMIN)*V(I)
      V(1)=V(I)
      T(1)=T(I)
      VS=V(I)
      I=1
200   GOTO 4000
C
 4025 IF(IFPLT.NE.2) WRITE(6,91) STNN,ID,IM,IY,IH,MIN,T(I),VV
   91 FORMAT(1X,A10,1X,3I2,1X,2I2,2F10.3)
 4030 JMAX=J
205 C   END OF DISCHARGE CALCULATION, INPUT DATA STATISTICS-
      KOUNT=KOUNT-1
      BACKSPACE 5
      PGD=FLOAT(KOUNT)/FLOAT(KDGOOD)
      PGOOD=100.*KDGOOD/(FLOAT(J)-FLOAT(ND1))
210   WRITE(6,4100) KOUNT,PGD,J,KDGOOD,PGOOD
 4100 FORMAT(1X,/,* RECORDS READ          =*,I6,
     1/,* MEAN NO REC/GOOD DAYS =*,F9.2,
     2/,* TOTAL NO OF DAYS      =*,I6,
     3/,* NO OF VALID DAYS      =*,I6,
215  4/,* % VALID DAYS          =*,F9.2,//)
C
C   FILL TO 30/6/82 WITH MISSING RECORD INDICATOR (-2)
 4055 IF(J.GE.NDT) GOTO 4060
      J=J+1
220   TN(J)=TN(J-1)+TIN
      VN(J)=VMN(J)=VMX(J)=-2.
      GOTO 4055
 4060 CONTINUE
C
225   100 FORMAT(A10,1X,3I2,1X,2I2,9X,F8.3)
      IH=0
      MIN=0
      DO 2430 I=1,NDT
```

```
                NDAY=INT(TN(I))
230             CALL DATE(ID,IM,IY,NDAY)
                WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I),VMN(I),VMX(I)
                IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I),
               1VMN(I),VMX(I)
 2430 CONTINUE
235  2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3,1X,F8.3,1X,F8.3)
 2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3,1X,F8.3,1X,F8.3)
C
C    CALC MONTHLY DISCHARGE AND MIN AND MAX DISCHARGE RATES
                NDP=0
240             NMTH=12
                DO 5000 NY=79,82
                IF(NY.EQ.82) NMTH=6
                IBADT=0
                SUMY=0.
245             NDP1=(NY-79)*365
                IF(NY.GT.80) NDP1=NDP1+1
                NTIT=2*NCY-1
                DO 6000 NM=1,NMTH
                IBAD=0
250             ND=MN(NM)
                IF(NY.EQ.80 .AND.NM.EQ.2) ND=ND+1
                SUM=0.
                DO 6010 J=1,ND
                K=NDP+J
255             V(J)=VMN(K)
                V(ND+J)=VMX(K)
                IF(VN(K).GE.-0.0001) GOTO 6020
                IBAD=IBAD+1
                GOTO 6010
260  6020 SUM=SUM+VN(K)
 6010 CONTINUE
                DM(NM)=SUM
                IF(IBAD.GT.0) DM(NM)=-SUM
                IBADT=IBADT+IBAD
265             NDP=NDP+ND
                NMNX=2*ND
                CALL MINMAXL(V,NMNX,VMIN,VMAX)
                VMNM(NM)=VMIN
                VMXM(NM)=VMAX
270  6000 CONTINUE
C
C    PRINT DISCHARGE TABLE
                WRITE(6,6100) TITLE(1),NY
 6100 FORMAT(1H1,//,45X,A10,5X,*19*,I2)
275             WRITE(6,6200)
 6200 FORMAT(40X,*DAILY DISCHARGE (MEGALITRES)*)
                WRITE(6,6300)
 6300 FORMAT(//,10X,*  DAY    JAN      FEB      MAR      APR      MAY      JUN*,
               1*      JUL      AUG      SEP      OCT      NOV      DEC*)
280             LEAP=0
                IF(NY.EQ.80) LEAP=1
                NROWS=28+LEAP
                DO 6400 I=1,NROWS
                K=NDP1+I
285             L=K+LEAP
```

```
      6400 WRITE(6,6500) I,VN(K),VN(K+31),(VN(L+MT(LL)),LL=3,NMTH)
      6500 FORMAT(11X,I2,1X,12F8.1)
           IF(LEAP.EQ.1) GOTO 6700
           I=29
290        K=NDP1+I
           WRITE(6,6600) I,VN(K),(VN(K+MT(LL)),LL=3,NMTH)
      6600 FORMAT(11X,I2,1X,F8.1,8X,10F8.1)
      6700 I=30
           K=NDP1+I
295        L=K+LEAP
           WRITE(6,6600) I,VN(K),(VN(L+MT(LL)),LL=3,NMTH)
           I=31
           K=NDP1+I
           L=K+LEAP
300        IF(NY.EQ.82) GOTO 6900
           WRITE(6,6800) I,VN(K),VN(L+MT(3)),VN(L+MT(5)),VN(L+MT(7)),
          1VN(L+MT(8)),VN(L+MT(10)),VN(L+MT(12))
      6800 FORMAT(11X,I2,1X,F8.1,3(8X,F8.1),F8.1,2(8X,F8.1))
           GOTO 7000
305   6900 WRITE(6,6950) I,VN(K),VN(L+MT(3)),VN(L+MT(5))
      6950 FORMAT(11X,I2,1X,F8.1,2(8X,F8.1))
      7000 WRITE(6,7100) (DM(I),I=1,NMTH)
      7100 FORMAT(/40X,*MONTHLY DISCHARGE ( - = PARTIAL SUM)*/,14X,
          112F8.1)
310        WRITE(6,7200) (VMNM(I),I=1,NMTH)
      7200 FORMAT(40X,*MIN AND MAX DISCHARGE RATE (LITRES/SEC)*/,14X
          112F8.1)
           WRITE(6,7220) (VMXM(I),I=1,NMTH)
      7220 FORMAT(14X,12F8.1)
315   C
           IFBADY=0
           N3M=4
           IF(NY.EQ.82) N3M=2
           DO 7300 I=1,N3M
320        SUM=0
           IFBAD=0
           DO 7320 J=1,3
           D=DM((I-1)*3+J)
           IF(D.GT.-0.0001)GOTO 7340
325        D=-D
           IFBAD=1
           IFBADY=1
      7340 SUM=SUM+D
      7320 CONTINUE
350        DM3(I)=SUM
           SUMY=SUMY+SUM
           IF(IFBAD.EQ.1) DM3(I)=-SUM
      7300 CONTINUE
           IF(IFBADY.EQ.1) SUMY=-SUMY
335   C
           WRITE(6,7400) (DM3(I),I=1,N3M)
      7400 FORMAT(40X,*3 MONTHLY DISCHARGE */,14X,4(16X,F8.1))
           WRITE(6,7500) IBADT,SUMY
      7500 FORMAT(29X,*NO OF DAYS DISCHARGE NOT CALCULATED =*,I5,10X,
          1*DISCHARGE FOR YEAR = *,F8.1)
340        WRITE(6,7600)
      7600 FORMAT(11X,*-1 = ABOVE STN RATING.*,5X,
```

```
          1*-2 = MISSING RECORD.*)
        C   CALL PLOTTING SUBROUTINE
345           IF(IFPLT) 2550,2550,2300
         2300 CONTINUE
              NJ=365
              IF(NY.EQ.80) NJ=366
              IF(NY.EQ.82) NJ=181
350           ZX=0.
              ZX=FLOAT(NDP1)
              CALL GRAPH(NCY,NJ,NY,NDP1,SX,ZX,SY,ZY)
         2550 CONTINUE
         5000 CONTINUE
355      1000 CONTINUE
              STOP
         9000 WRITE(6,9001) NNMAX,STNN,ID,IM,IY,IH,MIN,T(I-1),VV
         9001 FORMAT(1X,I4,* RECORDS/DAY -STOP*,A10,1X,3I2,1X,2I2,2F8.3)
              STOP
360           END
```

```
      PROGRAM EPL(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,
     1PLOT,TAPE7)
      COMMON TN(1300),VN(1300),VMN(1300),VMX(1300)
      DIMENSION T(100),V(100),TITLE(16),STN(8)
      DIMENSION DM(12),MN(12),MT(12),DM3(4),VMNM(12),VMXM(12)
      DATA MT/0,31,59,90,120,151,181,212,243,273,304,334/
      DATA MN/31,28,31,30,31,30,31,31,30,31,30,31/
C     ------------------------------------------------------
C    EPL TABULATES AND PLOTS MET DATA OF CUMMULATIVE TYPE
C    SUCH AS RAINFALL AND EVAPORATION
C
C    EPL CONTROL PARAMETERS:
C    NSTN,IFPLT,IFPF
C    NSTN  = NUMBER OF STATIONS
C    IFPLT = 0 PLOT NOT REQUIRED
C          = 1 PLOT IS REQUIRED
C          = 2 PLOT AND INPUT DATA PRINTOUT REQD
C    IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C          = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C    EPL SELECTS DATA FOR PLOTTING FROM A RAW DATA
C    FILE  : SPECIFICATIONS ARE -
C    INPUT QUEUE:
C     NSTN,IFPLT,IFPF
C     ID1,IM1,IY1,ID2,IM2,IY2  (FIRST AND LAST DATES REQD)
C     STN(I),TITLE(I)  (A10,2A10,/)  (I=1....NSTN)
C    (SX,SY,ZY) (* AXIS SCALES AND Y-ZERO)  (IFPLT = 1 ONLY)
C
C    EPL USES THE FOLLOWING SUBROUTINES AND FUNCTIONS :
C    DATE  DECDAY  GRAPH  MINMAXL  NDAYF  SETUP.
C
C-----------------------------------------------------------------
      NNMAX=100
      NWMAX=1300
      NDT=1277
      NG=0
      TIN=1.
      STIME=9./24.
      IH=9
      MIN=0
C
C    READ IN CONTROL DATA
      READ(8,*) NSTN,IFPLT,IFPF
      WRITE(6,51) NSTN,IFPLT,IFPF
   51 FORMAT(1H1,1X,*NO OF STNS REQUIRED              *,I1,/,1X,
     1* PLOT OF RESULTS REQUIRED  *,I1,/,1X,
     2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
     3* ( NO = 0 , YES = 1 )*/)
C
      READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
      WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
   45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X,
     2*END DATE       = *,3I2,/)
      WRITE(6,53) TIN,STIME
   53 FORMAT(1X,* TIN         =*,F6.3,/,1X,
     1* STIME       =*,F6.3,/)
      TS=DECDAY(ID1,IM1,IY1,0,0)
```

```
      TF=DECDAY(ID2,IM2,IY2,24,0)
      ZX=0.
      NCT=20
      DO 40 I=1,NSTN
      J=2*I-1
      K=J+1
      READ(8,2004) STN(I),TITLE(J),TITLE(K)
 2004 FORMAT(A10,2A10)
   40 WRITE(6,2006) STN(I),TITLE(J),TITLE(K)
 2006 FORMAT(1X,A10,2A10)
      IF(IFPLT.LT.1) GOTO 50
C
C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
      READ(8,*) SX,SY,ZY
      WRITE(6,3021) SX,ZX,SY,ZY
 3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
     12X,*SX ZX SY ZY =*,4F12.4)
   50 CONTINUE
C   CYCLE THROUGH STATIONS
C
      DO 1000 NCY=1,NSTN
C
      I=2*NCY-1
      TITLE(1)=TITLE(I)
      TITLE(2)=TITLE(I+1)
      IF(STN(NCY).EQ.10HTC         ) SY=SY*2.
      IF(IFPLT.GE.1) CALL SETUP(SX,ZX,SY,ZY,MN,TITLE)
C   INITIALIZE WORKING ARRAYS
      DO 10 I=1,NWMAX
      TN(I)=VN(I)=0.
   10 CONTINUE
      DO 20 I=1,NNMAX
      T(I)=V(I)=0.
   20 CONTINUE
      KOUNT=1
      MISS=0
      JS=1
C   READ FIRST RECORD
 7001 READ(5,100) STNN,ID,IM,IY,VV
      IF(EOF(5)) 1000,90
   90 IF(STNN.NE.STN(NCY)) GOTO 7001
  100 FORMAT(A10,1X,3I2,14X,F8.3)
      WRITE(6,101) STNN,ID,IM,IY,VV
  101 FORMAT(1X,A10,1X,3I2,14X,F8.3)
      J=NDAYF(ID,IM,IY)
      JP=J+1
      IF(J.GT.1) GOTO 7010
      IF(J.LT.1) GOTO 7001
      TN(J)=FLOAT(J)+STIME
      VN(J)=VV
      JS=J
C   READ SUBSEQUENT RECORDS
 7005 KOUNT=KOUNT+1
      READ(5,100) STNN,ID,IM,IY,VV
      IF(EOF(5)) 7004,7002
 7002 J=NDAYF(ID,IM,IY)
      IF(J.LE.JS) GOTO 7003
```

```
115               JSP=JS+1
                  IF(J.GT.JSP) GOTO 7010
                  TN(J)=FLOAT(J)+STIME
                  VN(J)=VV
                  JS=J
120               GOTO 7005
          C    MISSING RECORD -  SET VN(J)=-2.
           7010 TN(J)=FLOAT(J)+STIME
                  VN(J)=VV
                  JM=J-1
125               JS=J
                  DO 7000 I=JSP,JM
                  TN(I)=FLOAT(I)+STIME
           7000 VN(I)=-2.
                  MISS=MISS+JM-JSP+1
130               GOTO 7005
          C
          C    T(J) .LT. T(J-1)
           7003 J=JS
                  IF(IFPLT.NE.2) WRITE(6,102) STNN,ID,IM,IY,J,VV
135         102 FORMAT(1X,A10,1X,3I2,5X,I5,5X,F10.3)
          C    FINISH READING INPUT
           7004 JMAX=J
                  KOUNT=KOUNT-1
                  PVAL=100.*(FLOAT(J)-FLOAT(MISS))/(FLOAT(J))
140               WRITE(6,4100) KOUNT,J,MISS,PVAL
           4100 FORMAT(1X,/,* RECORDS READ           =*,I5,
                  1/,* TOTAL NO OF DAYS       =*,I6,
                  2/,* MISSING DAYS           =*,I6,
                  3/,* % VALID DAYS           =*,F9.2,//)
145               BACKSPACE 5
          C
          C    FILL TO 30/6/82 WITH MISSING RECORD INDICATOR (-2)
           4055 IF(J.GE.NDT) GOTO 4060
                  J=J+1
150               TN(J)=TN(J-1)+TIN
                  VN(J)=-2.
                  GOTO 4055
           4060 CONTINUE
          C
155               IH=9
                  MIN=0
                  DO 2430 I=1,NDT
                  NDAY=INT(TN(I))
                  CALL DATE(ID,IM,IY,NDAY)
160               WRITE(6,2401) STN(NCY),ID,IM,IY,IH,MIN,TN(I),VN(I)
                  IF(IFPF.GT.0) WRITE(7,2400) STN(NCY),ID,IM,IY,IH,MIN,TN(I),VN(I)
           2430 CONTINUE
           2400 FORMAT(A10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
           2401 FORMAT(1X,A10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
165       C
          C    CALC MONTHLY TOTALS AND MIN AND MAX VALUES
                  NDP=0
                  NMTH=12
                  DO 5000 NY=79,82
170               IF(NY.EQ.82) NMTH=6
                  IBADT=0
```

```
      SUMY=0.
      NDP1=(NY-79)*365
      IF(NY.GT.80) NDP1=NDP1+1
175   NTIT=2*NCY-1
      DO 6000 NM=1,NMTH
      IBAD=0
      ND=MN(NM)
      IF(NY.EQ.80 .AND.NM.EQ.2) ND=ND+1
180   SUM=0.
      DO 6010 J=1,ND
      K=NDP+J
      V(J)=VN(K)
      IF(VN(K).GE.-0.0001) GOTO 6020
185   IBAD=IBAD+1
      GOTO 6010
 6020 SUM=SUM+VN(K)
 6010 CONTINUE
      DM(NM)=SUM
190   IF(IBAD.GT.0) DM(NM)=-SUM
      IBADT=IBADT+IBAD
      NDP=NDP+ND
      NMNX=ND
      CALL MINMAXL(V,NMNX,VMIN,VMAX)
195   VMNM(NM)=VMIN
      VMXM(NM)=VMAX
 6000 CONTINUE
C
C   PRINT CALENDAR
200   WRITE(6,6100) TITLE(1),TITLE(2),NY
 6100 FORMAT(1H1,//,45X,2A10,5X,*19*,I2)
      WRITE(6,6200)
 6200 FORMAT(55X,*(MILLIMETRES)*)
      WRITE(6,6300)
205   6300 FORMAT(/,10X,*  DAY    JAN      FEB      MAR      APR      MAY      JUN*,
     1*      JUL      AUG      SEP      OCT      NOV      DEC*)
      LEAP=0
      IF(NY.EQ.80) LEAP=1
      NROWS=28+LEAP
210   DO 6400 I=1,NROWS
      K=NDP1+I
      L=K+LEAP
 6400 WRITE(6,6500) I,VN(K),VN(K+31),(VN(L+MT(LL)),LL=3,NMTH)
 6500 FORMAT(11X,I2,1X,12F8.1)
215   IF(LEAP.EQ.1) GOTO 6700
      I=29
      K=NDP1+I
      WRITE(6,6600) I,VN(K),(VN(K+MT(LL)),LL=3,NMTH)
 6600 FORMAT(11X,I2,1X,F8.1,8X,10F8.1)
220   6700 I=30
      K=NDP1+I
      L=K+LEAP
      WRITE(6,6600) I,VN(K),(VN(L+MT(LL)),LL=3,NMTH)
      I=31
225   K=NDP1+I
      L=K+LEAP
      IF(NY.EQ.82) GOTO 6900
      WRITE(6,6800) I,VN(K),VN(L+MT(3)),VN(L+MT(5)),VN(L+MT(7)),
```

```
     1VN(L+MT(8)),VN(L+MT(10)),VN(L+MT(12))
 6800 FORMAT(11X,I2,1X,F8.1,3(8X,F8.1),F8.1,2(8X,F8.1))
      GOTO 6990
 6900 WRITE(6,6950) I,VN(K),VN(L+MT(3)),VN(L+MT(5))
 6950 FORMAT(11X,I2,1X,F8.1,2(8X,F8.1))
 6990 WRITE(6,7100) (DM(I),I=1,NMTH)
 7100 FORMAT(/40X,*MONTHLY TOTAL ( - = PARTIAL SUM)*/,14X,
     112F8.1)
      WRITE(6,7200) (VMNM(I),I=1,NMTH)
 7200 FORMAT(40X,*MIN AND MAX DAILY FIGURE*/,14X
     112F8.1)
      WRITE(6,7220) (VMXM(I),I=1,NMTH)
 7220 FORMAT(14X,12F8.1)
C
      IFBADY=0
      N3M=4
      IF(NY.EQ.82) N3M=2
      DO 7300 I=1,N3M
      SUM=0
      IFBAD=0
      DO 7320 J=1,3
      D=DM((I-1)*3+J)
      IF(D.GT.-0.0001)GOTO 7340
      D=-D
      IFBAD=1
      IFBADY=1
 7340 SUM=SUM+D
 7320 CONTINUE
      DM3(I)=SUM
      SUMY=SUMY+SUM
      IF(IFBAD.EQ.1) DM3(I)=-SUM
 7300 CONTINUE
      IF(IFBADY.EQ.1) SUMY=-SUMY
C
      WRITE(6,7400) (DM3(I),I=1,N3M)
 7400 FORMAT(40X,*3 MONTHLY TOTAL     */,14X,4(16X,F8.1))
      WRITE(6,7500) IBADY,SUMY
 7500 FORMAT(33X,*NO OF DAYS RECORD NOT AVAILABLE =*,I5,14X,
     1*TOTAL FOR YEAR = *,F8.1)
      WRITE(6,7600)
 7600 FORMAT(11X,*-1 = ABOVE STN RATING.*,5X,
     1*-2 = MISSING RECORD.*)
C    CALL PLOTTING SUBROUTINE
      IF(IFPLT) 2550,2550,2300
 2300 CONTINUE
      NJ=365
      IF(NY.EQ.80) NJ=366
      IF(NY.EQ.82) NJ=181
      ZX=0.
      ZX=FLOAT(NDP1)
      CALL GRAPH(NCY,NJ,NY,NDP1,SX,ZX,SY,ZY)
 2550 CONTINUE
 5000 CONTINUE
 1000 CONTINUE
      STOP
      END
```

```
      PROGRAM EPLT(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE8,TAPE5,
     1PLOT,TAPE7)
      COMMON TN(1300),VN(1300),VMN(1300),VMX(1300)
      DIMENSION T(100),V(100),TITLE(24),ASTN(8)
      DIMENSION DM(12),MN(12),MT(12),DM3(4),VMNN(12),VMXX(12)
      DATA MT/0,31,59,90,120,151,181,212,243,273,304,334/
      DATA MN/31,28,31,30,31,30,31,31,30,31,30,31/
C     ------------------------------------------------------------
C    EPLT TABULATES AND PLOTS MET DATA OF THE NON-CUMMULATIVE TYPE
C    SUCH AS TEMPERATURE AND ATMOSPHERIC PRESSURE
C
C    EPLT CONTROL PARAMETERS:
C    NSTN,IFPLT,IFPF
C    NSTN  = NUMBER OF STATIONS
C    IFPLT = 0 PLOT NOT REQUIRED
C          = 1 PLOT IS REQUIRED
C          = 2 PLOT AND INPUT DATA PRINTOUT REQD
C    IFPF  = 0 PERMANENT FILE FOR OUTPUT IS NOT REQUIRED
C          = 1 ..    ..    ..    ..   .. IS REQUIRED
C
C    EPLT SELECTS DATA FOR PLOTTING FROM A RAW DATA
C    FILE  : SPECIFICATIONS ARE -
C    INPUT QUEUE:
C     NSTN,IFPLT,IFPF
C     ID1,IM1,IY1,ID2,IM2,IY2  (FIRST AND LAST DATES REQD)
C     ASTN(I),TITLE(I)  (I5,5X,2A10,/)  (I=1....NSTN)
C    (SX,SY,ZY) (* AXIS SCALES AND Y-ZERO)  (IFPLT = 1 ONLY)
C
C    EPLT USES THE FOLLOWING SUBROUTINES AND FUNCTIONS  :
C    DATE  DECDAY  GRAPH  MINMAXL  NDAYF  SETUP.
C
C-----------------------------------------------------------------
      NNMAX=100
      NWMAX=1300
      NDT=1277
      NG=0
      TIN=1.
      STIME=9./24.
      IH=9
      MIN=0
C
C    READ IN CONTROL DATA
      READ(8,*) NSTN,IFPLT,IFPF
      WRITE(6,51) NSTN,IFPLT,IFPF
   51 FORMAT(1H1,1X,*NO OF STNS REQUIRED              *,I1,/,1X,
     1* PLOT OF RESULTS REQUIRED  *,I1,/,1X,
     2* PERM FILE OF RESULTS REQUIRED *,I1,/,1X,
     3* ( NO = 0 , YES = 1 )*/)
C
      READ(8,*) IDTYPE
      WRITE(6,48) IDTYPE
   48 FORMAT(* IDTPE=*,I4)
      READ(8,*) ID1,IM1,IY1,ID2,IM2,IY2
      WRITE(6,45) ID1,IM1,IY1,ID2,IM2,IY2
   45 FORMAT(1X,*STARTING DATE = *,3I2,/,1X,
     2*END DATE      = *,3I2,/)
      WRITE(6,53) TIN,STIME
```

```
   53 FORMAT(1X,* TIN         =*,F6.3,/,1X,
     1* STIME       =*,F6.3,/)
      TS=DECDAY(ID1,IM1,IY1,0,0)
      TF=DECDAY(ID2,IM2,IY2,24.0)
      ZX=0.
      NCT=20
      DO 40 I=1,NSTN
      J=3*I-2
      K=J+1
      L=K+1
      READ(8,2004) ASTN(I),TITLE(J),TITLE(K),TITLE(L)
 2004 FORMAT(4A10)
   40 WRITE(6,2006) ASTN(I),TITLE(J),TITLE(K),TITLE(L)
 2006 FORMAT(1X,4A10)
      IF(IFPLT.LT.1) GOTO 50
C
C   SPECIFY SCALES FOR PLOTTING  SX, SY, ZY
      READ(8,*) SX,SY,ZY
      WRITE(6,3021) SX,ZX,SY,ZY
 3021 FORMAT(1X,* SCALES AND ORIGIN FOR PLOTTING */,
     12X,*SX ZX SY ZY =*,4F12.4)
   50 CONTINUE
C   CYCLE THROUGH STATIONS
C
      DO 1000 NCY=1,NSTN
C
      ISTNN=ASTN(NCY)
      I=3*NCY-2
      TITLE(1)=TITLE(I)
      TITLE(2)=TITLE(I+1)
      TITLE(3)=TITLE(I+2)
      IF(IFPLT.GE.1) CALL SETUP(SX,ZX,SY,ZY,MN,TITLE)
C   INITIALIZE WORKING ARRAYS
      DO 10 I=1,NWMAX
      TN(I)=VN(I)=0.
   10 CONTINUE
      DO 20 I=1,NNMAX
      T(I)=V(I)=0.
   20 CONTINUE
      KOUNT=1
      MISS=0
      JS=1
C   READ FIRST RECORD
 7001 READ(5,100) STNN,ID,IM,IY,VV
  100 FORMAT(A10,1X,3I2,14X,F8.3)
      IF(STNN.NE.ASTN(NCY)) GOTO 7001
      WRITE(6,101) STNN,ID,IM,IY,VV
  101 FORMAT(1X,A10,1X,3I2,14X,F8.3)
      J=NDAYF(ID,IM,IY)
      JP=J+1
      IF(J.GT.1) GOTO 7010
      IF(J.LT.1) GOTO 7001
      TN(J)=FLOAT(J)+STIME
      VN(J)=VV
      JS=J
C   READ SUBSEQUENT RECORDS
 7005 KOUNT=KOUNT+1
```

```
115               READ(5,100) STNN,ID,IM,IY,VV
                  IF(EOF(5)) 7004,7002
                7002 J=NDAYF(ID,IM,IY)
                  IF(J.LE.JS) GOTO 7003
                  JSP=JS+1
120               IF(J.GT.JSP) GOTO 7010
                  TN(J)=FLOAT(J)+STIME
                  VN(J)=VV
                  JS=J
                  GOTO 7005
125             C   MISSING RECORD -  SET VN(J)=-2.
                7010 TN(J)=FLOAT(J)+STIME
                  VN(J)=VV
                  JM=J-1
                  JS=J
130               DO 7000 I=JSP,JM
                  TN(I)=FLOAT(I)+STIME
                7000 VN(I)=-2.
                  MISS=MISS+JM-JSP+1
                  GOTO 7005
135             C
                C   T(J) .LT. T(J-1)
                7003 J=JS
                  IF(IFPLT.NE.2) WRITE(6,102) STNN,ID,IM,IY,J,VV
                 102 FORMAT(1X,A10,1X,3I2,5X,I5,5X,F10.3)
140             C   FINISH READING INPUT
                7004 JMAX=J
                  KOUNT=KOUNT-1
                  PVAL=100.*(FLOAT(J)-FLOAT(MISS))/(FLOAT(J))
                  WRITE(6,4100) KOUNT,J,MISS,PVAL
145               4100 FORMAT(1X,/,* RECORDS READ           =*,I6,
                    1/,* TOTAL NO OF DAYS       =*,I6,
                    2/,* MISSING DAYS           =*,I6,
                    3/,* % VALID DAYS           =*,F9.2,//)
                  BACKSPACE 5
150             C
                C   FILL TO 30/6/82 WITH MISSING RECORD INDICATOR (-2)
                4055 IF(J.GE.NDT) GOTO 4060
                  J=J+1
                  TN(J)=TN(J-1)+TIN
155               VN(J)=-2.
                  GOTO 4055
                4060 CONTINUE
                C
                  IH=9
160               MIN=0
                  DO 2430 I=1,NDT
                  NDAY=INT(TN(I))
                  CALL DATE(ID,IM,IY,NDAY)
                  WRITE(6,2401) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I)
165               IF(IFPF.GT.0) WRITE(7,2400) ISTNN,ID,IM,IY,IH,MIN,TN(I),VN(I)
                2430 CONTINUE
                2400 FORMAT(I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
                2401 FORMAT(1X,I10,1X,3I2,1X,2I2,1X,F8.3,F8.3)
                C
170             C   CALC MONTHLY MEAN AND MIN AND MAX DAILY VALUES
                  NDP=0
```

```
      NMTH=12
      DO 5000 NY=79,82
      IF(NY.EQ.82) NMTH=6
175   IBADT=0
      SUMY=0.
      NDP1=(NY-79)*365
      IF(NY.GT.80) NDP1=NDP1+1
      NTIT=2*NCY-1
180   DO 6000 NM=1,NMTH
      IBAD=0
      ND=MN(NM)
      IF(NY.EQ.80 .AND.NM.EQ.2) ND=ND+1
      SUM=0.
185   DO 6010 J=1,ND
      K=NDP+J
      V(J)=VN(K)
      IF(VN(K).GE.-0.0001) GOTO 6020
      IBAD=IBAD+1
190   GOTO 6010
 6020 SUM=SUM+VN(K)
 6010 CONTINUE
      DM(NM)=SUM/FLOAT(ND)
      IF(IBAD.GE.ND) GOTO 6015
195   IF(IBAD.GT.0) DM(NM)=SUM/(FLOAT(ND-IBAD))
 6015 IBADT=IBADT+IBAD
      NDP=NDP+ND
      NMNX=ND
      CALL MINMAXL(V,NMNX,VMIN,VMAX)
200   VMNM(NM)=VMIN
      VMXM(NM)=VMAX
 6000 CONTINUE
C
C   PRINT CALENDAR
205   WRITE(6,6100) TITLE(1),TITLE(2),TITLE(3),NY
 6100 FORMAT(1H1,//,40X,3A10,5X,*19*,I2)
      WRITE(6,6200)
 6200 FORMAT(56X,*(MILLIBARS)*)
      WRITE(6,6300)
210   6300 FORMAT(/,10X,*  DAY    JAN      FEB      MAR      APR      MAY      JUN*,
     1*      JUL      AUG      SEP      OCT      NOV      DEC*)
      LEAP=0
      IF(NY.EQ.80) LEAP=1
      NROWS=28+LEAP
215   DO 6400 I=1,NROWS
      K=NDP1+I
      L=K+LEAP
 6400 WRITE(6,6500) I,VN(K),VN(K+31),(VN(L+MT(LL)),LL=3,NMTH)
 6500 FORMAT(11X,I2,1X,12F8.1)
220   IF(LEAP.EQ.1) GOTO 6700
      I=29
      K=NDP1+I
      WRITE(6,6600) I,VN(K),(VN(K+MT(LL)),LL=3,NMTH)
 6600 FORMAT(11X,I2,1X,F8.1,8X,10F8.1)
225   6700 I=30
      K=NDP1+I
      L=K+LEAP
      WRITE(6,6600) I,VN(K),(VN(L+MT(LL)),LL=3,NMTH)
```

```
                I=31
230             K=NDP1+I
                L=K+LEAP
                IF(NY.EQ.82) GOTO 6900
                WRITE(6,6800) I,VN(K),VN(L+MT(3)),VN(L+MT(5)),VN(L+MT(7)),
               1VN(L+MT(8)),VN(L+MT(10)),VN(L+MT(12))
235          6800 FORMAT(11X,I2,1X,F8.1,3(8X,F8.1),F8.1,2(8X,F8.1))
                GOTO 6990
             6900 WRITE(6,6950) I,VN(K),VN(L+MT(3)),VN(L+MT(5))
             6950 FORMAT(11X,I2,1X,F8.1,2(8X,F8.1))
             6990 WRITE(6,7100) (DM(I),I=1,NMTH)
240          7100 FORMAT(/40X,*MONTHLY MEAN  ( - = PARTIAL SUM)*/,14X,
                112F8.1)
                WRITE(6,7200) (VMNM(I),I=1,NMTH)
             7200 FORMAT(40X,*MIN AND MAX DAILY MEAN*/,14X
                112F8.1)
245             WRITE(6,7220) (VMXM(I),I=1,NMTH)
             7220 FORMAT(14X,12F8.1)
           C
                IFBADY=0
                N3M=4
250             IF(NY.EQ.82) N3M=2
                DO 7300 I=1,N3M
                SUM=0
                IFBAD=0
                DO 7320 J=1,3
255             D=DM((I-1)*3+J)
                IF(D.GT.0.0001)GOTO 7340
                D=0.
                IFBAD=1
                IFBADY=1
260          7340 SUM=SUM+D
             7320 CONTINUE
                DM3(I)=SUM/3.
                SUMY=SUMY+SUM
                IF(IFBAD.EQ.1) DM3(I)=0.
265          7300 CONTINUE
                SUMY=SUMY/(3.*FLOAT(N3M))
                IF(IFBADY.EQ.1) SUMY=0.
           C
                WRITE(6,7400) (DM3(I),I=1,N3M)
270          7400 FORMAT(40X,*3 MONTHLY MEAN     */,14X,4(16X,F8.1))
                WRITE(6,7500) IBADT,SUMY
             7500 FORMAT(33X,*NO OF DAYS RECORD NOT AVAILABLE =*,I5,14X,
               1*MEAN  FOR YEAR = *,F8.1)
                WRITE(6,7600)
275          7600 FORMAT(11X,*-1 = ABOVE STN RATING.*,5X,
               1*-2 = MISSING RECORD.*)
           C   CALL PLOTTING SUBROUTINE
                IF(IFPLT) 2550,2550,2300
             2300 CONTINUE
280             NJ=365
                IF(NY.EQ.80) NJ=366
                IF(NY.EQ.82) NJ=181
                ZX=0.
                ZX=FLOAT(NDP1)
285             CALL GRAPH(NCY,NJ,NY,NDP1,SX,ZX,SY,ZY)
```

```
      2550 CONTINUE
      5000 CONTINUE
      1000 CONTINUE
           STOP
290        END
```

```
      FUNCTION AHD(ISTNN)
      DIMENSION N(124),A(124)
      N(1)=10
      A(1)=54.08
      N(2)=20
      A(2)=62.54
      N(3)=30
      A(3)=55.08
      N(4)=40
      A(4)=57.61
      N(5)=50
      A(5)=58.50
      N(6)=60
      A(6)=63.97
      N(7)=70
      A(7)=51.42
      N(8)=80
      A(8)=55.12
      N(9)=91
      A(9)=48.98
      N(10)=92
      A(10)=49.31
      N(11)=100
      A(11)=49.17
      N(12)=110
      A(12)=57.33
      N(13)=120
      A(13)=54.05
      N(14)=130
      A(14)=48.30
      N(15)=140
      A(15)=42.40
      N(16)=150
      A(16)=46.11
      N(17)=160
      A(17)=41.69
      N(18)=170
      A(18)=41.60
      N(19)=180
      A(19)=38.60
      N(20)=190
      A(20)=36.78
      N(21)=200
      A(21)=41.49
      N(22)=210
      A(22)=38.62
      N(23)=220
      A(23)=39.96
      N(24)=230
      A(24)=35.61
      N(25)=240
      A(25)=42.82
      N(26)=250
      A(26)=39.97
      N(27)=260
      A(27)=34.51
      N(28)=275
```

```
            A(28)=31.78
            N(29)=277
60          A(29)=31.80
            N(30)=285
            A(30)=31.28
            N(31)=287
            A(31)=31.30
65          N(32)=290
            A(32)=0.0
            N(33)=305
            A(33)=29.99
            N(34)=307
70          A(34)=29.91
            N(35)=315
            A(35)=26.90
            N(36)=317
            A(36)=26.90
75          N(37)=320
            A(37)=26.96
            N(38)=330
            A(38)=28.73
            N(39)=340
80          A(39)=25.06
            N(40)=355
            A(40)=26.24
            N(41)=357
            A(41)=26.24
85          N(42)=360
            A(42)=26.34
            N(43)=375
            A(43)=27.24
            N(44)=377
90          A(44)=27.26
            N(45)=380
            A(45)=24.13
            N(46)=390
            A(46)=22.82
95          N(47)=400
            A(47)=22.11
            N(48)=415
            A(48)=19.90
            N(49)=417
100         A(49)=20.07
            N(50)=420
            A(50)=18.31
            N(51)=430
            A(51)=18.42
105         N(52)=445
            A(52)=18.24
            N(53)=447
            A(53)=18.28
            N(54)=450
110         A(54)=18.89
            N(55)=465
            A(55)=18.46
            N(56)=467
            A(56)=18.51
```

```
115               N(57)=475
                  A(57)=17.14
                  N(58)=477
                  A(58)=17.13
                  N(59)=485
120               A(59)=16.55
                  N(60)=487
                  A(60)=16.63
                  N(61)=495
                  A(61)=17.11
125               N(62)=497
                  A(62)=17.16
                  N(63)=500
                  A(63)=15.74
                  N(64)=515
130               A(64)=7.39
                  N(65)=517
                  A(65)=7.37
                  N(66)=520
                  A(66)=9.78
135               N(67)=530
                  A(67)=8.34
            C
            C   FIND REFERENCE ELEVATION OF TOP OF BORE CASING
                  MN=124
140               DO 1 I=1,MN
                  IF(ISTNN.EQ.N(I)) GOTO 2
                1 CONTINUE
                  AHD=0.
                  RETURN
145               2 AHD=A(I)
                  RETURN
                  END
```

```
      SUBROUTINE CONST(NTIN,NFEND,NC,MAX,NN,NJ,ITYPE)
      COMMON T(1280),V(1280),VN(140,16),K(300),IR(140),NT(16),
     1NDT(16),W(1280)
C     NT(1)=NFEND
C     DO 15 I=2,MAX
C  15 NT(I)=NT(I-1)+NTIN
      DO 20 I=1,MAX
      MAY=NT(I)
      CALL DATE(ID,IM,IY,MAY)
      NDT(I)=ID*10000+IM*100+IY
   20 CONTINUE
      VU=0.
      ENDTOL=FLOAT(NTIN)/10.
C   FIND FIRST NT AFTER THE START OF DATA
      DO 300 I=1,MAX
      FNT=NT(I)+0.375
  300 IF(FNT.GT.T(1)) GOTO 320
      WRITE(6,*) #FIRST VALUE #,T(1),# AFTER LAST TIME INT #,NT(MAX)
  320 K(2*NC-1)=I
      IN=I
      L=LU=1
C   FIND FIRST DATA BEFORE AND AFTER EACH OF THE NT'S
  330 DO 400 J=L,NN
      FNT=NT(IN)+0.375
  400 IF(T(J).GT.FNT) GOTO 420
  500 CONTINUE
      ENDSPN=FNT-T(NN)
      IF(ENDSPN.LE.ENDTOL) GOTO 520
      K(2*NC)=IN-1
      NJ=IN-1
      RETURN
  520 VN(NC,IN)=V(NN)
      K(2*NC)=IN
      NJ=IN
      RETURN
  420 LU=J
      L=LU-1
      IF(ITYPE.EQ.2)GOTO 440
C   CHECK THAT V(LU) NOT DRY - ADVANCE LU
  340 IF(V(LU).LT.99.) GOTO 350
      LU=LU+1
      IF(LU.LE.NN) GOTO 340
      GOTO 500
C   CHECK THAT V(L) NOT DRY - REDUCE L
  350 IF(V(L).LT.99.) GOTO 380
      L=L-1
      IF(L.GE.1) GOTO 350
      VN(NC,IN)=-1.
      GOTO 460
C   CHECK THAT TIME STEP BETWEEN DATA IS NOT TOO LARGE
C   MAX TIME SPAN T(LU)-T(L) IS 2*TIME INTERVAL USED
  380 SPANMX=0.6*NTIN
      SPANU=T(LU)-FNT
      IF(SPANU.LT.SPANMX)GOTO 450
      SPANSV=NTIN*0.6
      SPANL=FNT-T(L)
      IF(SPANL.LT.SPANSV) GOTO 450
```

```
                VN(NC,IN)=-1.
                GOTO 460
60          C   PERFORM BORE CONSUMPTION CALCS USING WEIGHTED DAYS
              440 CONTINUE
            C     VL=VU
            C     IF(V(LU).EQ.V(L)) GOTO 441
            C     IFNT=FNT
65          C     IFNTM=IFNT-1
            C     INTL=T(L)
            C     INTLUM=T(LU)-1
            C     IF(IFNTM.LE.INTL) GOTO 441
            C     SUM1=0.
70          C   PREVENT W FROM HAVING NEGATIVE INDEX
            C     IF(INTL.GT.0) GOTO 446
            C     SUM1=-W(1)*(INTL+1)
            C     INTL=1
            C 446 CONTINUE
75          C     DO 442 I=INTL,IFNTM
            C 442 SUM1=SUM1+W(I)
            C     SUM2=SUM1
            C     DO 443 I=IFNT,INTLUM
            C 443 SUM2=SUM2+W(I)
80          C     VU=V(L)+(V(LU)-V(L))*(FNT-T(L))*SUM1/((T(LU)-T(L))*(SUM1+SUM2))
            C     GOTO 444
            C 441 VU=V(L)
            C 444 VN(NC,IN)=VU-VL
            C     IF(VS(NC,1).GT.0.0) VN(NC,IN)=VN(NC,IN)*4.546
85          C     GOTO 460
            C   SURVIVORS OF THE CHECKS ARE CALCULATED
              450 VN(NC,IN)=V(L)+(V(LU)-V(L))*(FNT-T(L))/(T(LU)-T(L))
              460 IN=IN+1
                  IF(IN.LE.MAX) GOTO 330
90                NJ=MAX
                  RETURN
                  END
```

```
      SUBROUTINE DATE(ID,IM,IY,NDAY)
      DIMENSION MN(13)
      DATA MN/0,31,59,90,120,151,181,212,243,273,304,334,365/
      ND=NDAY
      IY=79
      IF(ND.LE.365)GOTO 4
      ND=ND-365
      IY=IY+1
      IF(ND.LE.366) GOTO 2
      ND=ND-366
    3 IY=IY+1
      IF(ND.LE.365)GOTO 4
      ND=ND-365
      GOTO 3
    2 IF(ND.EQ.60) GOTO 7
      IF(ND.GE.61)ND=ND-1
    4 CONTINUE
      DO 5 I=2,13
      IF(ND.LE.MN(I)) GOTO 6
    5 CONTINUE
    6 IM=I-1
      ID=ND-MN(IM)
      RETURN
    7 ID=29
      IM=2
      RETURN
      END
```

```
      FUNCTION DECDAY(ID,IM,IY,IH,MIN)
      DIMENSION MN(12)
      DATA MN/0,31,59,90,120,151,181,212,243,273,304,334/
      NDY=365
      LEAPDY=0
      IF(IY-80) 1,2,3
    2 IF(IM.LE.2) GOTO 1
    3 LEAPDY=1
    1 NDAY=ID+MN(IM)+(IY-79)*365+LEAPDY
C   SET DECDAY=0 AT 0000 HRS 31/12/78
      DECDAY=NDAY+(MIN/60.+IH)/24.
      RETURN
      END
```

```
      FUNCTION FLORATE(A1,B1,H12,A2,B2,H23,A3,B3,V)
      IF(V.LE.1.000) GOTO 10
      IF(V.GT.H12) GOTO 1
      FLORATE=A1*(V-1.000)**B1
      RETURN
    1 IF(V.GT.H23) GOTO 2
      FLORATE=A2*(V-1.000)**B2
      RETURN
    2 FLORATE=A3*(V-1.000)**B3
      RETURN
   10 FLORATE=0.0
      RETURN
      END
```

```
      SUBROUTINE GRAPH(NCY,NJ,NY,NDP1,SX,ZX,SY,ZY)
      COMMON TN(1300),VN(1300),VMN(1300),VMX(1300),TITLE(15)
      DIMENSION T(370),V(370),TM(100)
C   GRAPH PLOTS GRAPHS ON ONE PAIR OF AXES
C   IT IS SELF SCALING - SCALES ARE COMMON TO ALL GRAPHS
C   INPUT - NO OF DATA PTS , ARRAYS CONTAINING COORD PTS, TITLES
C   ------------------------------------------------------------
      YLIM=5.2*SY
      ITRUNK=0
      DO 100 I=1,NJ
      K=NDP1+I
      T(I)=TN(K)
      V(I)=VN(K)
      IF(VN(K).LE.YLIM) GOTO 120
      ITRUNK=ITRUNK+1
      IF(ITRUNK.EQ.1) WRITE(6,300)
  300 FORMAT(1H1)
      WRITE(6,110) ITRUNK,TN(K),VN(K)
  110 FORMAT(1X,*ITRUNK*,I5,2F10.3)
      V(I)=YLIM
      TM(ITRUNK)=T(I)
  120 IF(VN(K).GT.-0.0001) GOTO 100
      V(I)=VN(K)*SY*0.15
  100 CONTINUE
C   NO SYMBOLS ON LINE
      J=0
C   RESET ORIGIN
      B=6.
      IF(NY.NE.79) GOTO 150
      XO=22.
      YO=18.3
      GOTO 180
  150 XO=-B
      YO=0.
      IF(NY.EQ.80) YO=0.05
      IF(NY.EQ.81) YO=-0.05
  180 CONTINUE
      CALL PLOT(XO,YO,-1)
      T(NJ+1)=ZX
      T(NJ+2)=-SX
      V(NJ+1)=ZY
      V(NJ+2)=SY
      CALL LINE(V,T,NJ,1,J,0)
C   IDENTIFY TRUNCATED DATA ON PLOT
      IF(ITRUNK.EQ.0) RETURN
      VM=5.2
      DO 200 I=1,ITRUNK
      TT=(ZX-TM(I))/SX
  200 CALL SYMBOL(VM,TT,0.2,18,-90.,-1)
      RETURN
      END
```

```
      SUBROUTINE GRF(NCY,NJ,ISN,K,SX,SY,ZX,ZY,T,V)
C     THIS VERSION OF GRF IS CALLED ONLY FROM PROGRAM BPL
      DIMENSION T(1280),V(1280)
      IF(K.GT.0) GOTO 60
C     RESET ORIGIN OF GRAPH
      XO=10.
      YO=0.
      YO=22.0
      CALL PLOT(XO,YO,-1)
   60 T(NJ+1)=ZX
      T(NJ+2)=-SX
      V(NJ+1)=ZY
      V(NJ+2)=SY
      IF(K.GE.1, GOTO 80
      CALL NUMBER(0.,2.2,0.3,ISN,0.,3HI10)
      CALL AXIS(0.,0.,2HGL,2,8.,0.,ZY,SY,-1)
      CALL AXIS(0.,0.,3HDAY,-3,17.,-90.,ZX,SX,1)
      CALL LINE(V,T,NJ,1,10,NCY)
C     PRINTS LINE OF DATA PTS WITH SYMBOL EACH 10 POINTS
      RETURN
   80 LSYM=K+2
      CALL LINE(V,T,NJ,1,-1,LSYM)
      RETURN
      END
```

```
      SUBROUTINE GRF(NB)
C     THIS VERSION OF GRF IS USED BY PROGRAMS BAHD,BPLOT AND BSTAT.
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
      COMMON IDS(140)
      DIMENSION T(1280),V(1280)
      ISTNN=IDS(NB)
      X=4.6-0.4*FLOAT(NB)
      IF(NB.NE.1) GOTO 50
      CALL SYMBOL(4.6,17.6,0.28,3HKEY,-90.,3)
      CALL SYMBOL(4.6,15.5,0.28,4HBORE,-90.,4)
      CALL SYMBOL(4.6,12.6,0.28,4HEAST,-90.,4)
      CALL SYMBOL(4.6,9.8,0.28,5HNORTH,-90.,5)
      CALL SYMBOL(4.6,7.0,0.28,9HCASING EL,-90.,9)
C     ENTER BORE KEY ON PLOT
   50 IRF=IREF(ISTNN)
      IS=IVS(IRF)
      IE=INT(CE(IRF))
      IN=INT(CN(IRF))
      HT=EL(IRF)
      CALL SYMBOL(X,17.2,0.28,NB,-90.,-1)
      CALL NUMBER(X,16.8,0.28,IS,-90.,3HI10)
      CALL NUMBER(X,13.8,0.28,IE,-90.,3HI10)
      CALL NUMBER(X,10.8,0.28,IN,-90.,3HI10)
      CALL NUMBER(X,7.8,0.28,HT,-90.,5HF10.2)
      RETURN
      END
```

```
      FUNCTION IREF(ISTNN)
      COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
C
C     FIND REFERENCE NO FOR BORE
      MN=140
      DO 1 I=1,MN
      IF(ISTNN.EQ.IVS(I)) GOTO 2
    1 CONTINUE
    2 IREF=I
      RETURN
      END
```

```
1                SUBROUTINE MINMAX(A,NJ,AMIN,AMAX)
                 DIMENSION A(1280)
                 AMIN=A(1)
                 AMAX=A(1)
5                DO 1 I=1,NJ
           C   FIND MIN AND MAX VALUES OF ARRAY A
                 IF(A(I).GT.AMAX) GOTO 2
                 IF(A(I).LT.AMIN) AMIN=A(I)
                 GOTO 1
10               2 AMAX=A(I)
                 1 CONTINUE
                 RETURN
                 END
```

```
      SUBROUTINE MINMAXL(A,NJ,AMIN,AMAX)
      DIMENSION A(100)
      AMIN=A(1)
      AMAX=A(1)
      K=NJ-2
C     K IS OUTSIDE PERIOD OF INTEREST
      DO 1 I=2,NJ
      IF(I.EQ.K) GOTO 1
C     FIND MIN AND MAX VALUES OF ARRAY A
      IF(A(I).GT.AMAX) GOTO 2
      IF(A(I).LT.AMIN) AMIN=A(I)
      GOTO 1
    2 AMAX=A(I)
    1 CONTINUE
      IF(AMIN.LT.-0.9999 .AND. AMIN.GT.-1.0001) GOTO 3
      RETURN
    3 AMIN=AMAX
      AMAX=-1.
      DO 4 I=2,NJ
      IF(A(I).LT.0.) GOTO 4
      IF(A(I).LT.AMIN) AMIN=A(I)
    4 CONTINUE
      RETURN
      END
```

```
      FUNCTION NDAYF(ID,IM,IY)
      DIMENSION MN(12)
      DATA MN/0,31,59,90,120,151,181,212,243,273,304,334/
      NDY=365
      LEAPDY=0
      IF(IY-80) 1,2,3
    2 IF(IM.LE.2) GOTO 1
    3 LEAPDY=1
    1 NDAYF=ID+MN(IM)+(IY-79)*365+LEAPDY
C   SET DECDAY=0 AT 0000 HRS 31/12/78
C     DECDAY=NDAYF+(MIN/60.+IH)/24.
      RETURN
      END
```

```
      SUBROUTINE RATING(ISTNN,NR,A1,B1,H12,A2,B2,H23,A3,B3)
C     SELECT APPROPRIATE RATING TABLE
C     RATING TABLES FOR GAUGING STATIONS
      NR=2
      A3=-1.
      B3=0.
      A2=-1.
      B2=0.
      H12=H23=1.6
      GOTO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,20),ISTNN
    1 A1=1855.
      B1=2.593
      H12=1.3
      RETURN
    2 A1=1738.
      B1=2.098
      H12=1.32
      RETURN
    3 A1=1249.
      B1=2.375
      A2=3040.
      B2=4.118
      H12=1.60
      H23=2.41
      RETURN
    4 A1=2381.
      B1=2.493
      H12=1.44
      RETURN
    5 A1=1669.
      B1=2.604
      A2=5501.
      B2=4.538
      H12=1.54
      H23=1.72
      RETURN
    6 A1=14374.
      B1=2.824
      H12=1.25
      RETURN
    7 A1=2054.
      B1=2.372
      A2=8409.
      B2=3.909
      H12=1.40
      H23=1.65
      RETURN
    8 A1=39137.
      B1=3.278
      H12=1.21
      RETURN
    9 A1=17513.
      B1=2.927
      H12=1.23
      RETURN
   10 A1=2123.
      B1=3.440
```

```
                H12=1.3
                WRITE(6,111) ISTNN
  111 FORMAT(1X,*WARNING - STN *,I2,* RATING IS POOR QUALITY*/)
                RETURN
   11 A1=12936.
                B1=2.731
                H12=1.32
                RETURN
   12 A1=1307.
                B1=2.408
                A2=2767.
                B2=3.539
                H12=1.51
                H23=1.85
                RETURN
   13 A1=7981.
                B1=2.434
                A2=9139.
                B2=2.456
                H12=1.22
C    A2,B2 USED FOR H.LT.H12 AFTER JULY 1980
                RETURN
   14 A1=393.
                B1=1.289
                H12=1.3
                WRITE(6,111) ISTNN
                RETURN
   15 A1=1297.
                B1=2.394
                A2=2669.
                B2=3.464
                H12=1.51
                H23=1.81
                RETURN
   20 A1=1.0
                B1=1.0
                NR=0
                H12=H23=9.999
                RETURN
                END
```

```
1                SUBROUTINE READREF
                 COMMON /REF/ IVS(140),EL(140),CE(140),CN(140)
           C    READ BORE COORDINATES
                 MN=140
5          C      CALL CONNEC(6)
                 DO 10 I=1,MN
                 READ(9,20) IVS(I),EL(I),CE(I),CN(I)
                 IF(EOF(9)) 30,10
              10 CONTINUE
10         C      WRITE(6,*) IVS(I),EL(I),CE(I),CN(I)
           C      CALL DISCON(6)
              20 FORMAT(I10,F10.2,2F10.0)
              30 IMAX=I-1
                 WRITE(6,40) IMAX
15            40 FORMAT(* TOTAL NO OF BORREFS = *,I6)
                 RETURN
                 END
```

```
      SUBROUTINE SETUP(SX,ZX,SY,ZY)
C     THIS VERSION OF SETUP IS USED BY PROGRAMS BAHD,BPLOT AND BSTAT.
C     SETUP INITIATES PLOTTING OF BORE DATA OVER 4 YEARS ON A4 SIZE
      CALL PLOT25
      XL=50.
      CALL XLIMIT(XL)
      CALL PAUPLOT(18HBLANK PAPER PLEASE,18)
      CALL SYMBOL(1.,18.,1.,16HNCDPD BOX 39 PLS,-90.,16)
      CALL PLOT(7.,0.,-1)
      CALL PLOT(0.,0.,3)
      CALL PLOT(28.,0.,2)
      CALL PLOT(28.,18.8,2)
      CALL PLOT(0.,18.8,2)
      CALL PLOT(0.,0.,2)
C     VERTICAL LINES
      Y=-1.4
      DY=2.4
      SV=6.
      SVP=5.8
      FV=28.
      YA=17.8
      DO 50 I=1,4
      Y=Y+DY
      CALL PLOT(SV,Y,3)
      CALL PLOT(FV,Y,2)
      Y=Y+DY
      CALL PLOT(FV,Y,3)
      CALL PLOT(SVP,Y,2)
   50 CONTINUE
      CALL PLOT(SV,YA,3)
      CALL PLOT(SV,1.,2)
C     SCALE VERTICAL AXES
      IA=0
      D=1.
      S=1.0
      T=0.8
      U=0.6
  150 X=5.
      CALL PLOT(SV,S,3)
      DO 100 I=1,22
      X=X+D
      CALL PLOT(X,S,3)
  100 CALL PLOT(X,T,2)
      X=5.9
      NGAP=5
      GAP=5.
      IF(SY.EQ.0.5) GAP=4.
      IF(SY.EQ.0.25) GAP=4.
      IF(GAP.EQ.4.) NGAP=6
      DO 200 I=1,NGAP
      VAL=ZY+GAP*SY*FLOAT(I-1)
      IV=INT(VAL)
      CALL NUMBER(X,U,0.3,IV,-90.,2HI2)
  200 X=X+GAP
      IA=IA+1
      S=17.8
      T=18.0
```

```
                U=18.5
                IF(IA.EQ.1) GOTO 150
60          C   LABEL HORIZ AXIS
                X1=17.4
                D=0.4
                DO 300 I=1,41
                CALL PLOT(6.,X1,3)
65              CALL PLOT(6.2,X1,2)
              300 X1=X1-D
                N1=1979
                N2=1980
                N3=1981
70              N4=1982
                CALL NUMBER(5.3,15.9,0.3,N1,-90.,2HI4)
                CALL NUMBER(5.3,11.1,0.3,N2,-90.,2HI4)
                CALL NUMBER(5.3,6.3,0.3,N3,-90.,2HI4)
                CALL NUMBER(5.3,2.6,0.3,N4,-90.,2HI4)
75          C   TITLES ON PAGE
                CALL SYMBOL(.3,12.5,.3,26HLITTLE PARA RECHARGE STUDY,-90.,26)
                CALL SYMBOL(.3,3.,0.3,3HFIG,-90.,3)
                CALL SYMBOL(0.9,14.7,.3,26HBORE WATER LEVEL ELEVATION,-90.,26)
                CALL SYMBOL(0.9,7.,0.3,12H(METRES AHD),-90.,12)
80              RETURN
                END
```

```
      SUBROUTINE SETUP(SX,ZX,SY,ZY,MN)
C   THIS VERSION OF SETUP IS CALLED BY PROGRAMS OPL,EPL AND EPLT.
      COMMON TN(1300),VN(1300),VMN(1300),VMX(1300),TITLE(15)
      DIMENSION MTH(6),MN(12)
      DATA MTH/10H  J    F  ,10H  M    A  ,10H  M    J  ,10H  J    A  ,
     110H  S    O  ,10H  N    D  /
C   SETUP INITIATES PLOTTING OF DAILY DATA OVER 4 YEARS ON A4 SIZE
      CALL PLOT25
      XL=50.
      CALL XLIMIT(XL)
      CALL PAUPLOT(18HBLANK PAPER PLEASE,18)
      CALL SYMBOL(1.,18.,1.,16HNCDPD BOX 39 PLS,-90.,16)
      CALL PLOT(7.,0.,-1)
      CALL PLOT(0.,0.,3)
      CALL PLOT(28.,0.,2)
      CALL PLOT(28.,19.,2)
      CALL PLOT(0.,19.,2)
      CALL PLOT(0.,0.,2)
      FY=0.
      FX=4.
      B=6.
      S=18.3
      DO 100 I=1,4
      TOP=FX+5.
      CALL PLOT(FX,0.,3)
      CALL PLOT(FX,S,2)
      CALL PLOT(TOP,S,2)
  100 FX=FX+8
C   WRITE DATES
      IYR=1982
      FX=0.
      FY=18.4
      DO 200 I=1,4
      FX=FX+6.
      CALL NUMBER(FX,FY,0.5,IYR,0.,2HI4)
      IYR=IYR-1
  200 CONTINUE
C   SCALE VERTICAL AXIS ON LEFT THEN RIGHT
      K=0
      IV=INT(SY+5.)
      FX=3.
      S=18.3
      T=18.1
      U=17.9
  450 D=1.
      DO 300 I=1,4
      FX=FX+D
      DO 400 J=1,5
      FX=FX+D
      CALL PLOT(FX,S,3)
  400 CALL PLOT(FX,T,2)
      ST=FX-0.15
  300 IF(K.EQ.0) CALL NUMBER(ST,U,0.3,IV,-90.,2HI2)
      K=K+1
      S=0.
      T=0.2
      FX=3.
```

```
      IF(K.LT.2) GOTO 450
C     LABEL MONTHS ON BOTTOM AXIS
      CALL SYMBOL(3.2,18.3,0.36,MTH,-90.,60)
C     SCALE BOTTOM THEN TOP AXIS
      J=0
      S=4.
      T=3.8
  600 Y=18.3
      DO 500 I=1,11
      Y=Y-FLOAT(MN(I))/20.
      CALL PLOT(S,Y,3)
      CALL PLOT(T,Y,2)
  500 CONTINUE
      S=28.
      T=27.8
      J=J+1
      IF(J.LT.2) GOTO 600
C     TITLES ON PAGE
      CALL SYMBOL(.5,14.5,.4,26HLITTLE PARA RECHARGE STUDY,-90.,26)
      CALL SYMBOL(.5,3.,0.4,3HFIG,-90.,3)
      CALL SYMBOL(1.5,14.7,.4,28HDAILY DISCHARGE (MEGALITRES),-90.,28)
      CALL SYMBOL(2.5,12.,.4,TITLE,-90.,20)
      RETURN
      END
```

```
      SUBROUTINE SFOWN(AMIN,AMAX,AL,SA,ZA)
      DIMENSION SN(5)
      DATA SN/1.,2.,2.5,5.,10./
C
C   GIVEN MIN AND MAX VALUES OF ARRAY A AND LENGTH OF AXIS,AL
C   FIND SCALE FACTOR,SA, AND ORIGIN VALUE ,ZA.
C
C   SCALE FACTOR
      NE=0
      AD=AMAX-AMIN
      SC=AD/AL
      IF(AD.NE.0.)GOTO 6
      WRITE(6,100)
  100 FORMAT(* STOP - GRAPH SHOWS NO CHANGE*)
      RETURN
    6 CONTINUE
      IF(SC.GT.10) GOTO 1
    7 CONTINUE
      IF(SC.LT.1.) GOTO 2
      DO 3 I=2,5
      IF(SN(I).GE.SC) GOTO4
    3 CONTINUE
    1 SC=SC/10.
      NE=NE+1
      GOTO 6
    2 SC=SC*10.
      NE=NE-1
      GOTO 7
    4 SA=SN(I)*10.**NE
C
C   ORIGIN  START WITH TEST FOR ZERO ORIGIN
      ZMIN=AMAX-AL*SA
      IF(ZMIN) 10,10,13
   10 IF(AMIN)13,11,11
   11 ZA=0.
      RETURN
C   SEARCH BETWEEN ZMIN AND AMIN FOR INTEGER
   13 AT=AMIN/SA
      MAT=AT
      IF(AMIN.LT.0.) MAT=AT-1.
      ZT=MAT*SA
      IF(ZT.GE.ZMIN) GOTO 20
      ZA=AMIN
      RETURN
   20 ZA=ZT
      RETURN
      END
```

```
      SUBROUTINE SMOOTH(NCY,NJ,NGRAPH,SX,SY,ZX,ZY,IFPF)
      COMMON TN(1280),VN(1280),IDS(140)
      DIMENSION A(1280),WK(2560),T(320)
C  FIRST ESTIMATE OF NO OF POINTS IN ERROR (LINEAR IINTERP)
      NJM=NJ-1
      C=0.1
  130 NER=0
      NSEP=-1
      NDER=0
      WRITE(6,*) #TRIGGER THRESHOLD (C) = #,C
      WRITE(6,200)
  200 FORMAT(* NON CONSEC  TRIGGER   I     TN(I)      VN(I)*,
     1*  LIN INTERP   DIFF*,/,*  TRIGGERS    NO*)
      DO 100 I=2,NJM
      IF(TN(I+1).EQ.TN(I-1)) GOTO 100
      Y=VN(I-1)+(VN(I+1)-VN(I-1))*(TN(I)-TN(I-1))/(TN(I+1)-TN(I-1))
      W=ABS(Y-VN(I))
      IF(W.LT.C) GOTO 100
      NER=NER+1
      NCT=I-NSEP
      IF(NCT.NE.1)NDER=NDER+1
      NSEP=I
      WRITE(6,110) NDER,NER,I,TN(I),VN(I),Y,W
  110 FORMAT(1X,2I10,I5,4F10.2)
      WRITE(7,500) IDS(NCY),TN(I),VN(I)
  500 FORMAT(1X,*H=*,I10,*=H=*,F8.3,*=G=*,F8.2,*=L*)
  100 CONTINUE
      NCK=NJ/10
      IF(NER.LE.NCK)GOTO 140
      WRITE(6,*) #TOTAL NO OF ERROR TRIGGERS IS TOO LARGE-TRY NEW C#
      C=C*2.
      GOTO 130
  140 CONTINUE
      WRITE(6,*) #EST NO OF ERRORS IS #,NDER
C
C   USE  NDER AS A GUIDE TO VALUE FOR MAXIT IN DATA SMOOTHER
C   ICSMOU IS IMSL DATA SMOOTHER USING ERROR DETECTION (CUBIC)
      DIS=1.0
      ISN=IDS(NCY)
      K=0
      IF(NGRAPH.GT.0) CALL GRF(NCY,NJ,ISN,K,SX,SY,ZX,ZY,TN,VN)
      WRITE(6,250) K
  250 FORMAT(* K=*,I3,*  GRAPH PLOTTED*)
C   PLOT LINE OF DATA POINTS
   60 K=K+1
      IF(K.EQ.1) MAXIT=NDER
      IF(K.EQ.2) MAXIT=NDER*3
      WRITE(6,*) MAXIT,DIS
      DO 30 J=1,NJ
   30 A(J)=VN(J)
      CALL ICSMOU(TN,A,NJ,DIS,0.,MAXIT,WK,IER)
      NCH=0
      WRITE(6,300)
  300 FORMAT(4X,*BORE      I    TN(I)      VN(I)      SPLINE      DIFF*,
     1/,37X,* ESTIMATE*)
      DO 40 J=1,NJ
      Z=A(J)-VN(J)
```

```
      IF(Z.EQ.0.) GOTO 24
      WRITE(6,90) IDS(NCY),J,TN(J),VN(J),A(J),Z
60    IF(IFPF.GT.0 .AND .K.EQ.1) WRITE(7,400) IDS(NCY),J,TN(J),
     1VN(J),A(J)
  400 FORMAT(I10,I5,3F10.2)
      NCH=NCH+1
      A(NCH)=A(J)
65    T(NCH)=TN(J)
   24 CONTINUE
   90 FORMAT(1X,I10,I5,4F10.2)
   40 CONTINUE
      WRITE(6,*) K,# SMOOTHED - MAXIT,DIS = #,MAXIT,DIS
70    IF(NGRAPH.GT.0) CALL GRF(NCY,NCH,ISN,<,SX,SY,ZX,ZY,T,A)
      WRITE(6,250) K
      IF(K.LE.1) GOTO 60
      RETURN
      END
```

DAMSIM

RESERVOIR OPERATION AND GROUNDWATER RECHARGE SIMULATOR

```
      PROGRAM DAMSIM(INPUT,OUTPUT,TAPE6=OUTPUT,TAPE5)
C
C     DAMSIM SIMULATES OPERATION OF A RESERVOIR WHICH ACTS AS A
C     PUMPED STORAGE AND ALSO RECEIVES CATCHMENT INFLOW.
C     ALTERNATIVE RELEASE POLICIES ARE TESTED TO PROVIDE RECHARGE
C     TO AN AQUIFER WITH MINIMAL PUMPING COST.
C     PROGRAM WRITTEN BY PETER DILLON, CIVIL ENG DEPT, UNIV OF
C     ADELAIDE, JANUARY 1983.
C     INITIALLY APPLIED TO LITTLE PARA RESERVOIR AND NORTHERN
C     ADELAIDE PLAINS AQUIFERS FOR ENGINEERING AND WATER SUPPLY
C     DEPARTMENT OF SOUTH AUSTRALIA.
C     ----------------------------------------------------------------
      COMMON /FLOW/ EN(12),T(12),W(12),X(12),QN(600)
      COMMON /STATE/ V(600),G(600),B(600)
      COMMON /METD/ EV(12),TMP(12),TMEAN
      COMMON /RPAR/ R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,
     1R13,R14,R15,R16,D(6)
      COMMON /MAX/ RMAX,VMAX,GMAX,PMAX,BMAX,EMAX
      COMMON /VARS/ N,J,JM,Q1,QCAP,RCAP,R,ER,C,CCAP,NCONT,NDRY,
     1ETL,RCAPN,CEN
      COMMON /RELS/ IREL,RL2,RL3,RL4,QL3,PCRIT,Q2,E,NT,RL9,
     1RL10,RL11,RL12,RL13,RL14,C3,B0,MDEL,P
      DIMENSION TITLE(20)
      DIMENSION YQN(12),YER(12),YP(12),YQ1(12),YE(12),YRECH(12),
     1YDOLP(12),YDOLE(12),YDOLSH(12),YX(12),YC(12),YL(12),
     2YSPILL(12),YTBAL(12),YDOLF(12)
      DIMENSION RREAL(600),SHEP(12),IFP(12),NBFAIL(12),NRFAIL(12)
C
C   INPUT TEXT GLOSSARY
C   UNITS ARE MONTHS,MEGALITRES,DOLLARS
C
C   IREL1 =1 IF HISTORIC RELEASE DATA TO BE USED (IREL=11)
C   NMAX  = NO OF MONTHS OPERATION TO BE SIMULATED
C   IF1   =1 IF INFLOW RECORD TO BE PRINTED
C   MN1   = MONTH NO OF FIRST MONTH (JAN=1,DEC=12)
C
C   RMAX  = RECHARGE CAPACITY OF ARTIFICIAL RECHARGE WORKS
C   VMAX  = RESERVOIR STORAGE CAPACITY AT F.S.L.
C   GMAX  = GROUNDWATER STORAGE CAPACITY (ARBITRARY LARGE NO)
C   PMAX  = PIPELINE CAPACITY FOR PUMPED INFLOW (ML/MONTH)
C   BMAX  = ARBITRARY DEFICIT LIMIT FOR EWS GROUNDWATER CREDIT ACCOUNT
C   EMAX = GW EXTRACTION CAPACITY FOR MAINS SUPPLY
C
C   R1    = MAX STREAMBED RECHARGE CAPACITY (84 TEMP ADJUSTMENT)
C   R2    = MIN     "       "       "     "   "       "
C   R3    = CATCHMENT AREA ABOVE DAM/FORMER CATCHMENT AREA ABOVE STN 3
C   R4    = EVAPOTRANSPIRATION CONSTANT= AREA VEGETATION ADJACENT
C           STREAM * E.T. COEFF. (14KM*20M*0.5*0.001ML/CU.M. =0.14)
C   R5    = RECHARGE PROPORTION OF FLOW EXCEEDING QCAP
C   R6      ACCOUNTS FOR NON UNIFORM STREAMBED WETNESS E.T.  (=1.0)
C   R7    = INITIAL STREAMBED RECHARGE CAPACITY
C   R8      ACCOUNTS FOR PROPORTION OF STREAMBED SUBJECT TO RUNOFF
C           BELOW DAM  (=1.0)
C
C   R9    = MAX INC IN RCAP (WHEN Q3.LT.R11)
C   R10   = MAX DEC IN RCAP (WHEN Q3.GT.R12)
C   R11   = MAX DISCHARGE AT STN 3 WITH INC IN RCAP=R9
```

```
      C   R12   = MIN DISCHARGE AT STN 3 WITH DEC IN RCAP=R10
      C   R13,R14 DEFINE DAM SEEPAGE  R13=-0.031  R14=31.
      C   R15,R16 DEFINE RECHARGE FROM RUNOFF ENTERING D/S STN 3
      C
      C   D(1) TO D(6) DEFINE THE EQUIVALENT UNIFORM FLOW WHICH
      C   PRODUCES THE SAME RECHARGE AS THE NATURAL FLOW PATTERN
      C   - SEE FUNCTION EQVQN-
      C
      C   C1 = B0 , WHERE B0 = INITIAL BALANCE IN GW RECHARGE ACCOUNT
      C   C2 = PCRIT/PMAX ,WHERE PCRIT= PUMPED DISCHARGE BELOW
      C        WHICH THE UNIT PUMPING COST IS UNIFORM
      C   C3 = MEASURE OF PREDICTIVE ABILITY OF OPERATOR AND
      C        FLEXIBILITY OF OPERATION
      C      = 0 IF PUMPING RATE SET AT START OF MONTH AND IS NOT
      C          RESET DURING MONTH
      C      = 1 IF PUMPING RATE CAN BE ADJUSTED DURING MONTH IN
      C          SYMPATHY WITH CATCHMENT INTAKE TO REACH TARGET
      C          LEVEL EXACTLY
      C        GUESS C3 LIES BETWEEN 0.3 AND 0.9
      C   C4 = CONSTANT RELEASE RATE FOR POLICY NO 0.
      C   TMEAN = MEAN TEMPERATURE  (=16.5 DEGREES C)
      C
      C   C7 = UNIT PUMPING COST UP TO PCRIT
      C   C8 = UNIT COST OF PUMPING AT PMAX
      C   C9   NOT USED
      C   C10= UNIT GROUNDWATER EXTRACTION COST UP TO EMAX
      C   C11,C12 NOT USED
      C   C13= UNIT SHADOW COST OF WASTE TO SEA FROM RELEASES OR SPILL
      C
      C   RL2,RL3,RL4,QL3 USED IN RELEASE RULES 3 AND 4 (SEE RELPOL)
      C   GOUT = DEEP PERCOLATION PLUS LATERAL OUTFLOW OF WATER IN
      C          SHALLOW AQUIFER
      C   MDEL NOT USED
      C
      C   NT   NOT USED
      C   RL9  =VCRIT/VMAX  ( IREL = 8)
      C   RL10 MINIMUM RESERVOIR VOLUME PERMITTED
      C   RL11 MIN RELEASE TO SATISFY LOCAL SHALLOW GW DEMAND
      C   RL12 =Q1/(QCAP-CEN) WHEN TARGET EXCEEDED AND SPILL UNLIKELY (IREL=5)
      C   RL13 SPILL FREEBOARD THRESHOLD (IREL=5)
      C   RL14 NOT USED
      C
      C   X  - MONTHLY WATER SUPPLY DEMAND
      C   W  - MONTHLY IRRIGATION GROUNDWATER DEMAND
      C   T  - TARGET RESERVOIR VOLUME AT END OF MONTH
      C   EV - MONTHLY NET EVAPORATION MINUS RAINFALL
      C   TMP- MONTHLY MEAN TEMPERATURE
      C   SHEP - MONTHLY RELEASES RECOMMENDED BY KINGSTON AND SHEPHERD
      C   IFP  - =1 IF FULL PIPELINE CAPACITY AVAILABLE IN THIS MONTH
      C   EN - EXPECTED NATURAL STN 3 DISCHARGE DURING MONTH
      C   QN - MONTHLY STN 3 DISCHARGE (NO DAM) SIMULATED DATA SET
      C   Q1 - RESERVOIR RELEASE DURING MONTH ACCORDING TO POLICY
      C   RREAL - HISTORICAL RELEASE RECORD
      C
      C   MODEL VARIABLES AND OUTPUT TEXT GLOSSARY
      C
      C   V  - RESERVOIR VOLUME AT END OF MONTH
```

```
115          C    G  - GROUNDWATER VOLUME AT END OF MONTH
             C    B  - G MINUS GW VOLUME WITHOUT RESERVOIR = BALANCE IN EWS
             C         RECHARGE CREDIT ACCOUNT
             C    P  - PUMPED INFLOW TO RESERVOIR
             C    R  - RECHARGE DUE TO RELEASES
120          C    C  - RECHARGE DUE TO DAM SEEPAGE AND RUNOFF FROM CATCHMENT
             C         D/S OF DAM
             C    ER - NATURAL RECHARGE (NO DAM)
             C    EVAP  - RESERVOIR EVAPORATION
             C    QCAP  - MAX UNIFORM DISCHARGE AT STN 3 WHICH WILL NOT REACH SEA
125          C    E     - GROUNDWATER EXTRACTION FOR MAINS SUPPLY
             C    N     - MONTH NUMBER SINCE START
             C    J     - MONTH OF YEAR
             C    DOLP  - PUMPING COST $
             C    DOLE  - GW EXTRACTION COST
130          C    DOLSH - SHADOW COST OF SPILL AND RESERVOIR RELEASES REACHING SEA
             C    DOLF  - SHADOW COST OF RESERVOIR STORAGE AND GROUNDWATER RECHARGE
             C            CREDIT AT END OF FINAL MONTH
             C    DOL   - DOLP+DOLE
             C    TBAL  - MEAN ANNUAL CHANGE IN RECHARGE CREDIT ACCOUNT
135          C    RF    - NO OF MONTHS RESERVOIR RAN DRY
             C    BF    - NO OF MONTHS THAT GROUNDWATER RECHARGE ACCOUNT FELL BELOW
             C            DEFICIT LIMIT (B<BMAX)
             C
             C    RELEASE POLICIES
140          C    0  - Q1 = CONSTANT RELEASE RATE = C4
             C    1  - Q1 = QCAP = RIVERBED+ARTIFICIAL FACILITY RECHARGE CAPACITY
             C    2  - Q1 = ESTIMATED FLOW LOSSES   (KINGSTON AND SHEPHERD 1973)
             C    3  - Q1 = F(PREVIOUS MONTH'S ACTUAL INFLOW)  (DILLON 1977A)
             C    4  - Q1 = F(PREVIOUS MONTH'S INFLOW,SEASON)  (DILLON 1977B)
145          C    5  - Q1 = ANTICIPATES SPILL AND RIVERBED RECHARGE CAPACITY
             C    6  - Q1 = QCAP BUT REDUCED IF VOLUME FALLS BELOW TARGET
             C    7  - Q1 = AS FOR =6 AND PART OF DEMAND X, IS MET FROM GROUNDWATER
             C    8  - Q1 = GROUNDWATER DEFICIT TRACKING,COMPARES RES VOL W/- TARGET
             C    9  - Q1 = TARGET RESERVOIR BALANCING BY GW EXTRACTION
150          C   10  - Q1 = EQUIVALENT UNIFORM NATURAL INFLOW (B=0+RECHARGE
             C              FROM DAM SEEPAGE)
             C   11  - Q1 = HISTORICAL RELEASE RECORD
             C
             C
155          C    READ CONTROL PARAMETERS,MODEL PARAMETERS
                    LR=5
                    LP=6
                    REWIND 5
             C
160                 READ(LR,1000) (TITLE(I),I=1,20)
              1000  FORMAT(20A4)
                    WRITE(LP,1100) (TITLE(I),I=1,20)
              1100  FORMAT(///,1X,20A4)
                    READ(LR,*) IREL1,NMAX,IF1,MN1
165                 WRITE(LP,2201) IREL1,NMAX,IF1,MN1
                    READ(LR,*) RMAX,VMAX,GMAX,PMAX,BMAX,EMAX
                    WRITE(LP,2202) RMAX,VMAX,GMAX,PMAX,BMAX,EMAX
                    READ(LR,*)  R1,R2,R3,R4,R5,R6,R7,R8
                    WRITE(LP,2203) R1,R2,R3,R4,R5,R6,R7,R8
170                 READ(LR,*)  R9,R10,R11,R12,R13,R14,R15,R16
                    WRITE(LP,2218) R9,R10,R11,R12,R13,R14,R15,R16
```

```
      READ(LR,*)  (D(J),J=1,6)
      WRITE(LP,2204) (D(J),J=1,6)
      READ(LR,*)  C1,C2,C3,C4,TMEAN
      WRITE(LP,2205) C1,C2,C3,C4,TMEAN
      READ(LR,*)  C7,C8,C9,C10,C11,C12,C13
      WRITE(LP,2206) C7,C8,C9,C10,C11,C12,C13
      READ(LR,*)  RL2,RL3,RL4,QL3,GOUT,MDEL
      WRITE(LP,2207) RL2,RL3,RL4,QL3,GOUT,MDEL
      READ(LR,*)  NT,RL9,RL10,RL11,RL12,RL13,RL14
      WRITE(LP,2208) NT,RL9,RL10,RL11,RL12,RL13,RL14
C     READ DATA
      READ(LR,*)  (X(J),J=1,12)
      WRITE(LP,2209) (X(J),J=1,12)
      READ(LR,*)  (W(J),J=1,12)
      WRITE(LP,2210) (W(J),J=1,12)
      READ(LR,*)  (T(J),J=1,12)
      WRITE(LP,2211) (T(J),J=1,12)
      READ(LR,*) (EV(J),J=1,12)
      WRITE(LP,2212) (EV(J),J=1,12)
      READ(LR,*) (TMP(J),J=1,12)
      WRITE(LP,2213) (TMP(J),J=1,12)
      READ(LR,*)  (SHEP(J),J=1,12)
      WRITE(LP,2216) (SHEP(J),J=1,12)
      READ(LR,*)  (IFP(J),J=1,12)
      WRITE(LP,2217) (IFP(J),J=1,12)
      READ(LR,*)  (EN(J),J=1,12)
      WRITE(LP,2214) (EN(J),J=1,12)
      NY=NMAX/(FLOAT(12))
      READ(LR,*) (QN(J),J=1,NMAX)
      WRITE(LP,2215) NY
      IF(IF1.EQ.0) GOTO 2300
      DO 2100 I=1,NY
      K=12*(I-1)
 2100 WRITE(LP,2200) I,(QN(K+L),L=1,12)
 2300 CONTINUE
      IF(IREL1.EQ.0) GOTO 2400
      DO 2500 L=1,NMAX
 2500 RREAL(L)=0.
      READ(LR,*) MSTART,MEND,MZERO
      WRITE(LP,2401) MSTART,MEND,MZERO
 2401 FORMAT(* ACTUAL RELEASES (Q1=RREAL(J)) MSTART,MEND =*,2I10,
     110X,*MZERO =*,I10)
      READ(LR,*) (RREAL(J),J=MSTART,MEND)
      DO 2600 I=1,NY
      K=12*(I-1)
 2600 WRITE(LP,2200) I,(RREAL(K+L),L=1,12)
 2400 CONTINUE
 2200 FORMAT(1X,I6,12F10.1)
 2201 FORMAT(* IREL1,NMAX,IF1,MN1                =*,4I10)
 2202 FORMAT(* RMAX,VMAX,GMAX,PMAX,BMAX,EMAX     =*,6F10.3)
 2203 FORMAT(* R1,R2,R3,R4,R5,R6,R7,R8            =*,8F10.3)
 2218 FORMAT(* R9,R10,R11,R12,R13,R14,R15,R16     =*,8F10.3)
 2204 FORMAT(* D(J)                              =*,6F10.3)
 2205 FORMAT(* C1,C2,C3,C4,TMEAN                 =*,5F10.3)
 2206 FORMAT(* C7,C8,C9,C10,C11,C12,C13          =*,7F10.6)
 2207 FORMAT(* RL2,RL3,RL4,QL3,GOUT,MDEL         =*,5F10.3,I10)
 2208 FORMAT(* NT,RL9,RL10,RL11,RL12,RL13,RL14   =*,I10,6F10.3)
```

```
 2209 FORMAT(* X(J) =*,12F10.1)
 2210 FORMAT(* W(J) =*,12F10.1)
 2211 FORMAT(* T(J) =*,12F10.1)
 2212 FORMAT(* EV(J)=*,12F10.1)
 2213 FORMAT(* TMP()=*,12F10.1)
 22.4 FORMAT(* EN(J)=*,12F10.1)
 2215 FORMAT(* NATURAL INFLOW DATA SET (QN(J),J=1,12)  *,I3,* YEARS*)
 2216 FORMAT(* SHEP =*,12F10.1)
 2217 FORMAT(* IFP()=*,12I10)
C
C    LOOP THROUGH EACH RELEASE POLICY
      KKMAX=11+IREL1
      DO 9000 KK=1,KKMAX
      IREL=KK-1
      WRITE(LP,2700) IREL
 2700 FORMAT(///,10X,*-----------------------------------*,
     1*   RELEASE POLICY NO *,I5,*   --------------------*,
     2*-----------------*,/)
C    INITIALIZE
      N=1
      J=MN1
      JM=MN1-1
      IF(JM.EQ.0) JM=12
      VO=T(JM)
      BO=C1
      GO=0.5*GMAX
      E=0.
      PCRIT=C2*PMAX
      PFACT=(C8-C7)/(PMAX-PCRIT)
      NGFAIL=0
      NBFAIL(KK)=NPFAIL(KK)=0
      NDRY=0
      NCONT=0
      TQN=TER=TP=TQ1=TE=0.
      TX=TC=TL=TSPILL=TRECH=0.
      TDOLP=TDOLE=TDOLSH=0.
      GOTO 3001
C    START OF TIME LOOP
 3000 N=N+1
      JM=J
      J=J+1
      IF(J.EQ.13) J=1
      VO=V(N-1)
 3001 CONTINUE
C    CALC RECHARGE POTENTIAL OF STREAMBED (+ARTIFICIAL FACILITY)
      CALL RECH
C    PREDICT INFLOW FOR MONTH (AT START C3=0, AT END C3=1 )
C    R3=RATIO OF CATCHMENT ABOVE DAM TO CATCHMENT AT GS504503
      QIN=(C3*QN(N)+(1.-C3)*EN(J))*R3
C    ASSUME CONSERVATIVE PUMPING POLICY S.T. TARGET LEVEL IS
C    ACHIEVED FOR ALL MONTHS WHEN PUMPING IS ALLOWED
C    (PUMPING STARTS EARLIER IN DRY MONTHS WHEN V(N-1).LT.TARGET)
      IF(QN(N).LT.EN(J)) QIN=QN(N)
C    SET THE RELEASE RATE ACCORDING TO OPERATING POLICY
      Q1=C4
      IF(IREL.EQ.0) GOTO 3050
C    SET RELEASE (Q1)= RECH CAPACITY+ET (QCAP) - EXPECTED RECH FROM
```

```
      C   D/S RUNOFF (CEN)
            CALL ECN
            Q1=QCAP-CEN
            IF(Q1.LT.0.) Q1=0.
290         IF(IREL.EQ.1) GOTO 1
            IF(IREL.EQ.6 .OR. IREL.EQ.7) GOTO 1
            IF(IREL.EQ.5) CALL REL5
            IF(IREL.EQ.8) CALL REL8
            IF(IREL.EQ.9) CALL REL9
295         IF(Q1.GT.QCAP) Q1=QCAP
            IF(IREL.EQ.2)  Q1=SHEP(J)
            IF(IREL.EQ.3 .OR. IREL.EQ.4) CALL RELPOL
            IF(IREL.EQ.10) Q1=EQVQN(O,N)
            IF(IREL.NE.11) GOTO 1
300         Q1=RREAL(N)
            IF(N.GE.MZERO) GOTO 1
            Q1=EQVQN(O,N)
          1 CONTINUE
      C   CALCULATE PUMPING TO ACHIEVE TARGET RESERVOIR LEVEL
305    3050 P=T(J)-V0+Q1+X(J)+EVAP(V0,J)-QIN
      C   PIPELINE CAPACITY AVAILABLE ONLY IN  SPECIFIED MONTHS (IFP(J)=1)
            IF(IREL.NE.9) E=0.
            IF(IREL.EQ.6 .OR. IREL.EQ.7) CALL REL57
            IF(IFP(J).EQ.0) P=0.
310         IF(P.LT.0.) P=0.
            IF(P.GT.PMAX) P=PMAX
            V1=V0+P+QN(N)*R3-Q1-X(J)-EVAP(V0,J)+E
      C   ALLOW PUMPING AT PREMIUM COST TO AVERT RESERVOIR FAILURE
            IF(V1.GT.RL10) GOTO 3400
315         P=RL10-V1
            V1=RL10
      C   CALCULATE RESERVOIR VOLUME AT END OF MONTH
     3400 VM=0.5*(V0+V1)
            EL=EVAP(VM,J)
320         V(N)=V0+P+QN(N)*R3-Q1-X(J)-EL+E
            IF(V(N).GT.0.) GOTO 3600
            NRFAIL(KK)=NRFAIL(KK)+1
            WRITE(LP,3500) NRFAIL(KK),V(N),V0,V1
     3500 FORMAT(* RESERVOIR IS DRY - POLICY FAILS*,I5,3F9.0)
325    3600 SPILL=0.
            IF(Q1.GT.0.) GOTO 3800
            NDRY=NDRY+1
            NCONT=0
            GOTO 3900
330    3800 NCONT=NCONT+1
            NDRY=0
     3900 IF(V(N).LT.VMAX) GOTO 4000
            SPILL=V(N)-VMAX
            V(N)=VMAX
335   C   CALC RECHARGE DUE TO RESERVOIR RELEASE (R)
     4000 CALL Q1REL
            IF(SPILL.LT.0.0001) GOTO 4100
            IF(IREL.EQ.10) GOTO 4100
            R=R+SPILL*R5
340    4100 CONTINUE
      C   CALC RECHARGE WHICH RES INFLOW WOULD HAVE GIVEN (ER)
            CALL ERN
```

```
      C   CALC RECHARGE DUE TO RUNOFF FROM D/S CATCHMENT (C)
          CALL CRN
      C   CALCULATE ARTIFICIAL RECHARGE CREDIT (B)
          IF(N.EQ.1) GOTO 4001
          BO=B(N-1)
 4001 B(N)=BO+R-ER-E+C
          IF(B(N).GT.BMAX) GOTO 5000
          NBFAIL(KK)=NBFAIL(KK)+1
          IF(NBFAIL(KK).LE.10) WRITE(LP,4010) N,NBFAIL(KK),B(N),R,ER,E
 4010 FORMAT(* MONTH *,I4,*  RECHARGE DEFICIT EXCESSIVE *,I4,4F10.1)
 5000 CONTINUE
      C   CALCULATE GROUNDWATER STORAGE
          IF(N.EQ.1) GOTO 5001
          GO=G(N-1)
      C   GOUT=DEEP PECOLATION+OUTFLOW. APPROX=MEAN MONTHLY INTAKE MINUS PUMPING
 5001 G(N)=GO+R+C-W(J)-E-GOUT
          GMIN=GMAX*0.25
          IF(G(N).GT.GMIN) GOTO 5100
          NGFAIL=NGFAIL+1
          WRITE(LP,5010) N,NGFAIL,G(N),GMAX,GMIN,GO,R,C,W(J),E,GOUT
 5010 FORMAT(* MONTH *,I4,*  AQUIFER LEVEL LOW *,I4,9F10.1)
 5100 IF(G(N).LT.GMAX) GOTO 6000
          WRITE(LP,5020) N,G(N),GMAX,GO,R,C,W(J),E,GOUT
 5020 FORMAT(* MONTH *,I4,*  AQUIFER FULL *,8F10.1)
 6000 CONTINUE
      C   CALCULATE PUMPING AND EXTRACTION COSTS
          DOLP=0.
          IF(P.EQ.0.) GOTO 6200
          IF(IFP(J).EQ.1) GOTO 6100
          D2=C7+(PMAX-P)*PFACT
          DOLP=P*0.5*(D2+C8)
          GOTO 6200
 6100 DOLP=C7*P
          IF(P.LE.PCRIT) GOTO 6200
          D1=C7+(P-PCRIT)*PFACT
          DOLP=C7*PCRIT+0.5*(C7+D1)*(P-PCRIT)
 6200 DOLE=C10*E
          DOL=DOLP+DOLE
      C   CALC SHADDOW COST OF SPILL AND RELEASES REACHING THE SEA
          DOLSH=C13*SPILL
          SEA=Q1-R-ETL
          IF(SEA.GT.0.) DOLSH=DOLSH+C13*SEA
          IF(N.EQ.1) WRITE(LP,7001)
 7001 FORMAT(1X,*  N  J    QN      P      Q1      E      X    QCAP*,
         1*      R       ER       C   B(N)    G(N)   V(N)    EVAP  SPILL*,
         2*    DOL    DOLSH*/)
          WRITE(LP,7000) N,J,QN(N),P,Q1,E,X(J),QCAP,R,ER,C,B(N),G(N),
         1V(N),EL,SPILL,DOL,DOLSH
 7000 FORMAT(1X,2I3,16F7.0)
          TQN=TQN+QN(N)
          TER=TER+ER
          TP=TP+P
          TQ1=TQ1+Q1
          TE=TE+E
          TRECH=TRECH+R
          TDOLP=TDOLP+DOLP
          TDOLE=TDOLE+DOLE
```

```
400               TDOLSH=TDOLSH+DOLSH
                  TX=TX+X(J)
                  TC=TC+C
                  TL=TL+EL
                  TSPILL=TSPILL+SPILL
405               IF(N.LT.NMAX) GOTO 3000
          C    ACCOUNT FOR RECHARGE CREDIT AND SURFACE STORAGE AT END OF RUN
                  DEBV=C13*(T(J)-V(NMAX))
                  DEBB=-C13*B(NMAX)
                  TEND=DEBV+DEBB
410               WRITE(LP,7900) DEBV,DEBB,TEND
           7900 FORMAT(/,* SHADOW COSTS OF FINAL STORAGES*/,
                 1*  RESERVOIR =*,F9.0,
                 2*  AQUIFER   =*,F9.0,
                 3*  TOTAL     =*,F9.0)
415        C    END OF TIME LOOP  CALC MEAN ANNUAL STATISTICS
                  YR=NMAX/12.
                  YQN(KK)=TQN/YR
                  YER(KK)=TER/YR
                  YP(KK)=TP/YR
420               YQ1(KK)=TQ1/YR
                  YE(KK)=TE/YR
                  YDOLP(KK)=TDOLP/YR
                  YDOLE(KK)=TDOLE/YR
                  YDOLSH(KK)=TDOLSH/YR
425               YDOLF(KK)=TEND/YR
                  YX(KK)=TX/YR
                  YC(KK)=TC/YR
                  YL(KK)=TL/YR
                  YSPILL(KK)=TSPILL/YR
430               YRECH(KK)=TRECH/YR
                  WRITE(LP,8000)
           8000 FORMAT(1X,///,* MEAN ANNUAL FLOWS AND COSTS*,/,
                 1*      QN          P         Q1          E          X          R*,
                 2*        ER          C       EVAP     SPILL      DOLP      DOLE   *,
435              3* DOLSH      DOLF    RF BF*)
                  WRITE(LP,8100) YQN(KK),YP(KK),YQ1(KK),YE(KK),YX(KK),YRECH(KK),
                 1YER(KK),YC(KK),YL(KK),YSPILL(KK),YDOLP(KK),YDOLE(KK),
                 1YDOLSH(KK),YDOLF(KK),NRFAIL(KK),NBFAIL(KK)
           8100 FORMAT(1X,14F9.0,2I3)
440               YTBAL(KK)=YRECH(KK)+YC(KK)
                  YTDOL=YDOLP(KK)+YDOLE(KK)+YDOLF(KK)
                  WRITE(LP,8200) IREL,YTDOL
           8200 FORMAT(1X,/,* RELEASE POLICY NO *,I5,*     TOTAL ANNUAL COST $*,
                 1F9.0)
445        9000 CONTINUE
                  WRITE(LP,8300)
           8300 FORMAT(1H1,*   SUMMARY OF RELEASE POLICY OUTCOMES*,/,*  IREL*,
                 1*     QN          P         Q1          E          X          R*,
                 2*       ER          C       EVAP     SPILL      DOLP      DOLE   *,
450              3*DOLSH      DOLF    TRECH RF BF*)
                  DO 9100 KK=1,KKMAX
                  IREL=KK-1
                  WRITE(LP,8400) IREL,YQN(KK),YP(KK),YQ1(KK),YE(KK),YX(KK),
                 1YRECH(KK),YER(KK),YC(KK),YL(KK),YSPILL(KK),YDOLP(KK),YDOLE(KK),
455              2YDOLSH(KK),YDOLF(KK),YTBAL(KK),NRFAIL(KK),NBFAIL(KK)
           8400 FORMAT(1X,I5,15F8.0,2I3)
```

```
 9100 CONTINUE
      STOP
      END
```

```
      SUBROUTINE RECH
C   THIS CALCULATES THE RECHARGE POTENTIAL OF THE STREAMBED
      COMMON /FLOW/ EN(12),T(12),W(12),X(12),QN(600)
      COMMON /STATE/ V(600),G(600),B(600)
      COMMON /METD/ EV(12),TMP(12),TMEAN
      COMMON /RPAR/ R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,
     1R13,R14,R15,R16,D(6)
      COMMON /MAX/ RMAX,VMAX,GMAX,PMAX,BMAX,EMAX
      COMMON /VARS/ N,J,JM,Q1,QCAP,RCAP,R,ER,C,CCAP,NCONT,NDRY,
     1ETL,RCAPN,CEN
      IF(N.NE.1) GOTO 2
      RCAP=R7
      DC=0.
      GOTO 3
    2 ACQ3=(1.-R3)*QN(N-1)+R13*QN(N-1)+R14
C   ACQ3=CATCHMENT RUNOFF BETWEEN DAM AND GS504503 IN LAST MONTH
      Q3=ACQ3+Q1
      DC=R9
      IF(Q3.LT.R11) GOTO 1
      IF(Q3.LT.R12) GOTO 4
      DC=R10
      GOTO 1
    4 SLOPE=(R10-R9)/(R12-R11)
      DC=-SLOPE*R11+R9+SLOPE*Q3
    1 RCAP=RCAP+DC
      IF(RCAP.GT.R1) RCAP=R1
      IF(RCAP.LT.R2) RCAP=R2
C   TEMPERATURE COMPENSATION FOR PERMIABILITY OF STREAMBED
    3 RCAPN=RCAP*(1.+0.1*(TMP(J)-TMEAN)/TMEAN)
C   ADD CAPACITY OF ARTIFICIAL RECHARGE FACILITIES (RMAX)
      TCAP=RCAPN+RMAX
C   ALLOW FOR EVAPOTRANSPIRATION LOSSES ALONG WETTED STREAMBED (ETL)
      ARTE=RMAX*2./(R1+R2)
      ETL=EV(J)*R4*(1.+ARTE)
      QCAP=TCAP+ETL
C   QCAP=MAX RELEASE RATE GIVING NO DISCHARGE INTO THE SEA
      RETURN
C
      ENTRY Q1REL
C   CALCULATES RECHARGE DUE TO RELEASE Q1
      ARTE=RMAX*2./(R1+R2)
      ET=EV(J)*R4
      IF(Q1.GT.QCAP) GOTO 100
      QCAPN=RCAPN+ET
      IF(Q1.GT.QCAPN) GOTO 150
      ETL=ET*Q1/QCAPN
      R=Q1-ETL
      RETURN
  150 ETL=ET*(1.+ARTE)
      R=Q1-ETL
      IF(R.LT.RCAPN) R=RCAPN
      RETURN
  100 R=RCAPN+RMAX+(Q1-ET*(1.+ARTE)-RCAPN-RMAX)*R5
      RETURN
C
      ENTRY ERN
C   CALCULATES NATURAL RECHARGE WHICH INFLOW TO RESERVOIR
```

```
      C   WOULD HAVE GIVEN IF THE DAM DID NOT EXIST
      C   R6 ACCOUNTS FOR REDUCED EVAP DUE TO NON-UNIFORM STREAM WETNESS
   60       DQN=EQVQN(Q,N)
            CQN=R15*QN(N)+R16
            IF(CQN.GT.QN(N)) CQN=QN(N)
            ET=R4*EV(J)
            QCAPN=RCAPN+ET
   65       IF(DQN.GE.QCAPN) GOTO 200
            EDL=R6*R4*EV(J)*DQN/QCAPN
            ERU=DQN-EDL
            IF(ERU.LT.0.) ERU=0.
            ALL=DQN+CQN
   70       IF(ALL.GE.QCAPN) GOTO 250
      C   U/S + D/S RUNOFF .LT. RECHARGE CAPACITY
            CTL=R6*R4*R8*EV(J)*CQN/QCAPN
            IF((EDL+CTL).GT.ET) CTL=ET-EDL
            ERD=CQN-CTL
   75       IF(ERU.LT.0.) ERU=0.
            IF(ERD.LT.0.) ERD=0.
            ER=ERU+ERD
            RETURN
      C   UPSTREAM DISCHARGE ALONE EXCEEDS RECHARGE CAPACITY
   80     200 EDL=R4*EV(J)
            EX=(DQN+CQN*R8-EDL-RCAPN)*R5
            IF(EX.LT.0.) EX=0.
            ER=RCAPN+EX
            RETURN
   85   C   U/S COMBINED W/- D/S RUNOFF EXCEEDS RECHARGE CAPACITY
        250 CTL=R6*R4*R8*EV(J)*(QCAPN-DQN)/QCAPN
            CF=(QCAPN-DQN)*R8-CTL
            CS=(ALL-QCAPN)*R8*R5
            IF(CF.LT.0.) CF=0.
   90       IF(CS.LT.0.) CS=0.
            IF(ERU.LT.0.) ERU=0.
            ER=ERU+CF+CS
            RETURN
      C
   95       ENTRY CRN
      C   THIS CALCULATES THE RECHARGE OCCURRING DUE TO INFLOW
      C   FROM THE CATCHMENT DOWNSTREAM OF THE DAM
      C   CQ3=DAM SEEPAGE (EXTRA FLOW GAIN BETWEEN DAM AND GS504503
      C       SINCE DAM CONSTRUCTION
  100       CQ3=R13*QN(N)+R14
            IF(CQ3.LT.0.) CQ3=0.
      C   CQN=RECHARGE CONTRIBUTION OF RUNOFF ENTERING BETWEEN STNS 3 AND 15
            CQN=R15*QN(N)+R16
            IF(CQN.GT.QN(N)) CQN=QN(N)
  105       CQN=CQN+CQ3
            ET=R4*EV(J)
            QCAPN=RCAPN+ET
            IF(Q1.GE.QCAPN) GOTO 300
            ALL=Q1+CQN
  110       IF(ALL.GT.QCAPN) GOTO 350
      C   U/S + D/S RUNOFF .LT. RECHARGE CAPACITY
            CTL=R6*R4*R8*EV(J)*CQN/QCAPN
            C=CQN-CTL
            IF(C.LT.0.) C=0.
```

```
115                RETURN
              C    U/S DISCHARGE ALONE EXCEEDS RECHARGE CAPACITY
                300 C=R5*CQN*R8
                    RETURN
              C    U/S COMBINED W/- D/S DISCHARGE EXCEED RECHARGE CAPACITY
120             350 CTL=R6*R4*R8*EV(J)*(QCAPN-Q1)/QCAPN
                    CF=(QCAPN-Q1)*R8-CTL
                    CS=(ALL-QCAPN)*R8*R5
                    IF(CS.LT.0.) CS=0.
                    IF(CF.LT.0.) CF=0.
125                 C=CS+CF
                    RETURN
              C
                    ENTRY ECN
              C    THIS CALCULATES THE EXPECTED RECHARGE DUE TO INFLOW
130           C    FROM THE CATCHMENT DOWNSTREAM OF THE DAM (W/O RELEASE)
                    CEN=R15*EN(J)+R16
                    IF(CEN.GT.EN(J)) CEN=EN(J)
                    CQ3=R13*EN(J)+R14
                    IF(CQ3.LT.0.) CQ3=0.
135                 CEN=CEN+CQ3
                    CCAP=RCAPN*R8
                    CTL=R6*R4*R8*EV(J)
                    QCCAP=CCAP+CTL
                    IF(CEN.GT.QCCAP) GOTO 400
140                 CTL=CTL*CEN/QCCAP
                    C=CEN-CTL
                    IF(C.LT.0.)C=0.
                    RETURN
                  400 C=CCAP+(CEN-CCAP-CTL)*R5
145                 IF(C.LT.CCAP) C=CCAP
                    RETURN
                    END
```

```
      SUBROUTINE RELPOL
C  THIS SELECTS THE RELEASE RATE AND THE GROUNDWATER EXRACTION
C  RATE ACCORDING TO THE ASSIGNED RELEASE POLICY NO  (IREL)
      COMMON /FLOW/ EN(12),T(12),W(12),X(12),QN(600)
      COMMON /STATE/ V(600),G(600),B(600)
      COMMON /METD/ EV(12),TMP(12),TMEAN
      COMMON /RPAR/ R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,
     1R13,R14,R15,R16,D(6)
      COMMON /MAX/ RMAX,VMAX,GMAX,PMAX,BMAX,EMAX
      COMMON /VARS/ N,J,JM,Q1,QCAP,RCAP,R,ER,C,CCAP,NCONT,NDRY,
     1ETL,RCAPN,CEN
      COMMON /RELS/ IREL,RL2,RL3,RL4,QL3,PCRIT,Q2,E,NT,RL9,
     1RL10,RL11,RL12,RL13,RL14,C3,BO,MDEL,P
      Q1=QN(N-1)
      IF(Q1.LE.RL2) GOTO 20
      IF(Q1.LT.QL3) GOTO 10
      Q1=RL3
      GOTO 20
   10 Q1=(Q1-RL2)*(RL3-RL2)/(QL3-RL2)
   20 IF(IREL.EQ.3) RETURN
      IF(J.GT.3 .AND. J.LT.10) GOTO 40
      IF(Q1.LT.RL2) Q1=RL2
   40 IF(Q1.GT.RL4) Q1=RL4
      RETURN
C
      ENTRY REL5
      IF(N.EQ.1) RETURN
      IF(V(N-1).GT.T(JM)) GOTO 101
      Q1=0.
      IF(NDRY.GE.4) Q1=RL11
      RETURN
  101 VSPILL=VMAX-RL13
      IF(V(N-1).GT.VSPILL) RETURN
      Q1=RL12*Q1
      IF(NCONT.GE.4) Q1=0.
      RETURN
C
      ENTRY REL67
      F=(PMAX-P)/(PMAX-PCRIT)
      IF(F.LT.0.) F=0.
      IF(F.GT.1.) F=1.
      Q2=F*Q1
      IF(IREL.EQ.7) GOTO 70
      Q1=Q2
      RETURN
   70 IF(P.LT.PCRIT) RETURN
      EXP=P-PCRIT
C   E= GROUNDWATER PARTIALLY REPLACES RES. WATER TO MEET DEMAND X(J)
C   WATER QUALITY CONSTRAINT ON MAINS WATER (X/E .LT. 0.5)
      E=(1.-F)*X(J)*0.5
      IF(E.GT.EXP) E=EXP
      IF(E.GT.EMAX) E=EMAX
      IF(N.EQ.1) GOTO 71
      IF(B(N-1).LT.BMAX) E=0.
   71 Q1=0.
      IF(NDRY.GE.4) Q1=RL11
      RETURN
```

```
      C
            ENTRY REL8
60    C  IREL=8
            IF(N.EQ.1) RETURN
            VCRIT=VMAX*RL9
            VE=V(N-1)-T(JM)
            IF(VE.LT.0.) GOTO 80
65          IF(VE.GE.VCRIT) RETURN
            IF(B(N-1).LT.0.) RETURN
            Q1=Q1*VE/VCRIT
            RETURN
         80 Q1=0.
70          IF(NDRY.GE.4) Q1=RL11
            RETURN
      C
            ENTRY REL9
      C   IREL=9
75          IF(N.EQ.1) RETURN
            IF(V(N-1).GE.T(JM)) RETURN
            Q1=0.
            IF(NDRY.GE.4) Q1=RL11
            E=T(JM)-V(N-1)
80          EMX=X(J)*0.5
            IF(E.GT.EMAX) E=EMAX
            IF(E.GT.EMX) E=EMX
            BTST=B(N-1)-E
            IF(BTST.GT.BMAX) RETURN
85          E=B(N-1)-BMAX
            IF(E.LT.0.) E=0.
            RETURN
            END
```

```
      FUNCTION EVAP(S,J)
C     THIS CALCULATES THE EVAPORATION LOSS FROM THE RESERVOIR
      COMMON /METD/ EV(12),TMP(12),TMEAN
      DIMENSION A(15),VA(15)
      DATA (A(I),I=1,15)/0.,1.98,5.12,11.9,28.7,48.8,64.3,76.6,88.5,
     199.8,111.0,121.7,132.4,142.7,146.9/
      DATA (VA(I),I=1,15)/0.,50.,250.,600.,2000.,4000.,6000.,8000.,
     110000.,12000.,14000.,16000.,18000.,20000.,20800./
      DO 10 I=2,15
      IF(S.LE.VA(I)) GOTO 20
   10 CONTINUE
      AREA=A(15)
      GOTO 30
   20 I=I-1
      AREA=A(I)+(S-VA(I))*(A(I+1)-A(I))/(VA(I+1)-VA(I))
   30 EVAP=AREA*EV(J)*0.7*0.01
C     0.7=LAKE EVAP/ A CLASS PAN EVAP
      RETURN
      END
```

```
      FUNCTION EQVQN(IFE,N)
C     THIS CALCULATES THE EQUIVALENT UNIFORM FLOW WHICH PRODUCES
C     THE SAME FLOW LOSS AS THE NATURAL FLOW PATTERN (OR EXPECTED
C     FLOW PATTERN IF IFE.EQ.1 )
      COMMON /FLOW/ EN(12),T(12),W(12),X(12),QN(600)
      COMMON /RPAR/ R1,R2,R3,R4,R5,R6,R7,R8,R9,R10,R11,R12,
     1R13,R14,R15,R16,D(6)
      EQVQN=QN(N)
      IF(IFE.EQ.1) EQVQN=EN(N)
      IF(EQVQN.LE.D(1)) RETURN
      IF(EQVQN.GT.D(2)) GOTO 1
      EQVQN=EQVQN*D(3)+D(4)
      RETURN
    1 EQVQN=EQVQN*D(5)+D(6)
      RETURN
      END
```