```
C*******************************************************************
C****                                                           ****
C**** SIMULATION OF TWO-PHASE OIL-WATER FLOW IN A HORIZONTAL    ****
C**** ONE-DIMENSIONAL SYSTEM USING THE IMPES FORMULATION AND    ****
C**** GAUSSIAN ELIMINATION                                      ****
C****                                                           ****
C**** MAXIMUM NUMBER OF GRID BLOCKS IS CURRENTLY SET AT 100     ****
C*******************************************************************
      PROGRAM OW
C
C-----DECLARATIONS
C
      INTEGER EN,I,J,K,N,NULL
      REAL*8 LAMOM,LAMOP,LAMWM,LAMWP
      REAL*8 PERM(100),PHI(100),DX(100),X(100)
      REAL*8 SWT(100),KROT(100),KRWT(100),PCT(100)
      REAL*8 KRO(100),KRW(100),PCOW(100),DPCOW(100)
      REAL*8 PT(100),BOT(100),BWT(100),MUOT(100),MUWT(100)
      REAL*8 BO(100),BW(100),DBO(100),DBW(100),MUO(100),MUW(100)
      REAL*8 SW(100),PO(100),PONEW(100),PW(100),LAMO(100),LAMW(100)
      REAL*8 TXOP(100),TXOM(100),TXWP(100),TXWM(100)
      REAL*8 CPOO(100),CPOW(100),CSWO(100),CSWW(100)
      REAL*8 A(100),B(100),C(100),D(100)
      REAL*8 AREA,SWI,CR,T,DT,TMAX,PINIT,PR,PL,QWI,ALFA
C
      CHARACTER*8 PHEAD(103)
      DATA PHEAD/" TIME   ","PL      ",
     *"P( 1)   ","P( 2)   ","P( 3)   ",
     *"P( 4)   ","P( 5)   ","P( 6)   ","P( 7)   ","P( 8)   ",
     *"P( 9)   ","P(10)   ","P(11)   ","P(12)   ","P(13)   ",
     *"P(14)   ","P(15)   ","P(16)   ","P(17)   ","P(18)   ",
     *"P(19)   ","P(20)   ","P(21)   ","P(22)   ","P(23)   ",
     *"P(24)   ","P(25)   ","P(26)   ","P(27)   ","P(28)   ",
     *"P(29)   ","P(30)   ","P(31)   ","P(32)   ","P(33)   ",
     *"P(34)   ","P(35)   ","P(36)   ","P(37)   ","P(38)   ",
     *"P(39)   ","P(40)   ","P(41)   ","P(42)   ","P(44)   ",
     *"P(44)   ","P(45)   ","P(46)   ","P(47)   ","P(48)   ",
     *"P(49)   ","P(50)   ","P(51)   ","P(52)   ","P(55)   ",
     *"P(54)   ","P(55)   ","P(56)   ","P(57)   ","P(58)   ",
     *"P(59)   ","P(60)   ","P(61)   ","P(62)   ","P(66)   ",
     *"P(64)   ","P(65)   ","P(66)   ","P(67)   ","P(68)   ",
     *"P(69)   ","P(70)   ","P(71)   ","P(72)   ","P(77)   ",
     *"P(74)   ","P(75)   ","P(76)   ","P(77)   ","P(78)   ",
     *"P(79)   ","P(80)   ","P(81)   ","P(82)   ","P(88)   ",
     *"P(84)   ","P(85)   ","P(86)   ","P(87)   ","P(88)   ",
     *"P(89)   ","P(90)   ","P(91)   ","P(92)   ","P(99)   ",
     *"P(94)   ","P(95)   ","P(96)   ","P(97)   ","P(98)   ",
     *"P(99)   ","P(100)  ","PR      "/
C
      CHARACTER*8 SHEAD(101)
      DATA SHEAD/" TIME   ","SW( 1)  ","SW( 2)  ","SW( 3)  ",
     *"SW( 4)  ","SW( 5)  ","SW( 6)  ","SW( 7)  ","SW( 8)  ",
     *"SW( 9)  ","SW(10)  ","SW(11)  ","SW(12)  ","SW(13)  ",
     *"SW(14)  ","SW(15)  ","SW(16)  ","SW(17)  ","SW(18)  ",
     *"SW(19)  ","SW(20)  ","SW(21)  ","SW(22)  ","SW(23)  ",
     *"SW(24)  ","SW(25)  ","SW(26)  ","SW(27)  ","SW(28)  ",
     *"SW(29)  ","SW(30)  ","SW(31)  ","SW(32)  ","SW(33)  ",
     *"SW(34)  ","SW(35)  ","SW(36)  ","SW(37)  ","SW(38)  ",
     *"SW(39)  ","SW(40)  ","SW(41)  ","SW(42)  ","SW(44)  ",
     *"SW(44)  ","SW(45)  ","SW(46)  ","SW(47)  ","SW(48)  ",
     *"SW(49)  ","SW(50)  ","SW(51)  ","SW(52)  ","SW(55)  ",
     *"SW(54)  ","SW(55)  ","SW(56)  ","SW(57)  ","SW(58)  ",
     *"SW(59)  ","SW(60)  ","SW(61)  ","SW(62)  ","SW(66)  ",
     *"SW(64)  ","SW(65)  ","SW(66)  ","SW(67)  ","SW(68)  ",
     *"SW(69)  ","SW(70)  ","SW(71)  ","SW(72)  ","SW(77)  ",
     *"SW(74)  ","SW(75)  ","SW(76)  ","SW(77)  ","SW(78)  ",
     *"SW(79)  ","SW(80)  ","SW(81)  ","SW(82)  ","SW(88)  ",
     *"SW(84)  ","SW(85)  ","SW(86)  ","SW(87)  ","SW(88)  ",
     *"SW(89)  ","SW(90)  ","SW(91)  ","SW(92)  ","SW(99)  ",
     *"SW(94)  ","SW(95)  ","SW(96)  ","SW(97)  ","SW(98)  ",
     *"SW(99)  ","SW(100) "/
C
      EN=1
      NULL=0
C
C----READ SYSTEM DATA FROM FILE SYST.DAT
C
      OPEN(12,FILE='SYST.DAT')
      READ(12,*)
C----USO AND USW ARE UPSTREAM WEIGHTING FACTORS FOR OIL AND WATER
C-------1.0 IS UPSTREAM
C-------0.0 OS DOWNSTREAM
C-------SHOULD BE A VALUE BETWEEN 0.0 AND 1.0 (STANDARD IS 1.0)
      READ(12,*)USO,USW
      READ(12,*)AREA
      READ(12,*)N
      READ(12,*)(DX(I),I=1,N)
      READ(12,*)(PHI(I),I=1,N)
      READ(12,*)(PERM(I),I=1,N)
      READ(12,*)SWI
      READ(12,*)CR
      READ(12,*)DT
      READ(12,*)TMAX
      READ(12,*)PINIT
      READ(12,*)PR
      READ(12,*)PL
      READ(12,*)QWI
      CLOSE(12)
C
C----GENERATE X-POSITIONS
C
      X(1)=DX(1)/2.
      DO 9 I=2,N
    9 X(I)=X(I-1)+(DX(I-1)+DX(I))/2.
C
C----DETERMINE WHICH LEFT SIDE BOUNDARY CONDIION TO BE USED
C----IF CONSTANT INJECTION RATE, IBC=2
C----IF CONSTANT LEFT SIDE PRESSURE, IBC=1
C
      IF(PL.NE.0.)THEN
      QWI=0.
      IBC=1
      ELSE
      IBC=2
      ENDIF
C
      QOP=0.
      WC=0.
C
      PHEAD(N+3)="PR     "
C
C----READ TABLES FOR RELATIVE PERMEABILITIES AND CAPILLARY PRESSURES
C----FROM INPUT DATA FILE DATA FILE SAT.DAT
C
```

```
      OPEN(10,FILE='SAT.DAT')
      READ(10,*)NSAT,PCMULT
      READ(10,*)
      DO 10 J=1,NSAT
      READ(10,*)SWT(J),KROT(J),KRWT(J),PCT(J)
   10 CONTINUE
      CLOSE(10)
C
      DO 11 J=1,NSAT
   11 PCT(J)=PCT(J)*PCMULT
C----READ PVT DATA FROM INPUT DATA FILE PVT.DAT
C
      OPEN(11,FILE='PVT.DAT')
      READ(11,*)NPVT
      READ(11,*)
      DO 20 J=1,NPVT
      READ(11,*)PT(J),BOT(J),BWT(J),MUOT(J),MUWT(J)
C
C----CONVERT BO AND BW TO 1/BO OG 1/BW
C
      BOT(J)=1./BOT(J)
      BWT(J)=1./BWT(J)
   20 CONTINUE
      CLOSE(11)
C
C-----OPEN FILES FOR PRINTOUT AND WRITE HEADINGS
C-----(PO.OUT FOR OIL PRESSURES, SW.OUT FOR WATER SATURATIONS
C-----AND WELLS.OUT FOR PRODUCTION/INJECTION RESULTS)
C
      OPEN(13,FILE='PO.OUT')
      WRITE(13,2001)(PHEAD(I),I=1,N+3)
      WRITE(13,2007)(X(I),I=1,N)
 2001 FORMAT(' TIME, RATES AND OIL PRESSURES',/,2X,100A8)
      OPEN(14,FILE='SW.OUT')
      WRITE(14,2002)(SHEAD(I),I=1,N+1)
      WRITE(14,2007)(X(I),I=1,N)
 2002 FORMAT(' TIME AND WATER SATURATIONS',/,2X,100A8)
      OPEN(15,FILE='WELLS.OUT')
      WRITE(15,2003)
 2003 FORMAT(' PRODUCTION/INJECTION RESULTS',/,
     * '   TIME   PL      QWI     PR       QOP      QWP     WC')
 2007 FORMAT(5X," X=",100F8.2)
C
C----INITIALIZATION
C
      T=0.
      DO 30 I=1,N
      PO(I)=PINIT
      SW(I)=SWI
   30 CONTINUE
      IF(IBC.EQ.2)PL=PO(1)
C
C----TIME LOOP
C
      DO 1000 J=1,1000
C
C----PRINTOUT OF TIME, PRESSURES, SATURATIONS AND PROD/INJ
C
      WRITE (13,2004)T,PL,(PO(I),I=1,N),PR
 2004 FORMAT(103F8.2)
      WRITE (14,2005)T,(SW(I),I=1,N)
 2005 FORMAT(F8.2,101F8.4)
```

```
      WRITE (15,2006)T,PL,QWI,PR,QOP,QWP,WC
 2006 FORMAT(7F8.2)
C
      T=T+DT
C
C----END OF RUN ?
C
      IF(T.GT.TMAX)GO TO 1001
C
C----INTERPOLATE IN INPUT DATA TABLES
C----FOR SATURATION DEPENDENT PROPERTIES (KRO, KRW, PCOW)
C----AND PRESSURE DEPENDENT PROPERTIES (BO, BW, MUO, MUW)
C
      DO 100 I=1,N
C
C----FIND REL. PERM. FOR OIL
C
      CALL INTERP(SW(I),KRO(I),DUMMY,NULL,NSAT,SWT,KROT)
C
C----FIND REL. PERM. FOR WATER
C
      CALL INTERP(SW(I),KRW(I),DUMMY,NULL,NSAT,SWT,KRWT)
C
C----FIND PC AND ITS DERIVATIVE
C
      CALL INTERP(SW(I),PCOW(I),DPCOW(I),EN,NSAT,SWT,PCT)
C
C----FIND (1/BO) AND ITS DERIVATIVE
C
      CALL INTERP(PO(I),BO(I),DBO(I),EN,NPVT,PT,BOT)
C
C----FIND (1/BW) AND ITS DERIVATIVE
C
      PW(I)=PO(I)-PCOW(I)
      CALL INTERP(PW(I),BW(I),DBW(I),EN,NPVT,PT,BWT)
C
C----FIND OIL VISCOSITY
C
      CALL INTERP(PO(I),MUO(I),DUMMY,NULL,NPVT,PT,MUOT)
C
C----FIND WATER VISCOSITY
C
      CALL INTERP(PW(I),MUW(I),DUMMY,NULL,NPVT,PT,MUWT)
C
C----COMPUTE MOBILITY TERMS
C
      LAMO(I)=KRO(I)*BO(I)/MUO(I)
      LAMW(I)=KRW(I)*BW(I)/MUW(I)
  100 CONTINUE
C
C----LOOP FOR GRID BLOCKS
C
      DO 200 I=1,N
C
      IF(I.NE.1)THEN
        LAMOM=LAMO(I-1)*USO+LAMO(I)*(1.-USO)
        LAMWM=LAMW(I-1)*USW+LAMW(I)*(1.-USW)
        IF(PO(I-1).LT.PO(I))LAMOM=LAMO(I)*USO+LAMO(I-1)*(1.-USO)
        IF(PW(I-1).LT.PW(I))LAMWM=LAMW(I)*USW+LAMW(I-1)*(1.-USW)
      ELSE
        LAMOM=LAMO(I)
        LAMWM=LAMW(I)
      ENDIF
      IF(I.NE.N)THEN
        LAMOP=LAMO(I)*USO+LAMO(I+1)*(1.-USO)
        LAMWP=LAMW(I)*USW+LAMW(I+1)*(1.-USW)
        IF(PO(I+1).GT.PO(I))LAMOP=LAMO(I+1)*USO+LAMO(I)*(1.-USO)
        IF(PW(I+1).GT.PW(I))LAMWP=LAMW(I+1)*USW+LAMW(I)*(1.-USW)
      ELSE
        LAMOP=LAMO(I)
        LAMWP=LAMW(I)
      ENDIF
C-----------------------------------TRANSMISSIBILITIES
      IF(I.NE.1)THEN
        TXOM(I)=2.*LAMOM/(DX(I)/PERM(I)+DX(I-1)/PERM(I-1))/DX(I)
        TXWM(I)=2.*LAMWM/(DX(I)/PERM(I)+DX(I-1)/PERM(I-1))/DX(I)
      ELSE
        TXOM(I)=2.*LAMOM/DX(I)*PERM(I)/DX(I)
        TXWM(I)=2.*LAMWM/DX(I)*PERM(I)/DX(I)
C----------------------------INJECTION OF WATER AT LEFT SIDE
C----------------------------REQUIRES SUM OF TRANSMISSIBILITIES
        TXWM(I)=TXWM(I)+TXOM(I)
      ENDIF
C
      IF(I.NE.N)THEN
        TXWP(I)=2.*LAMWP/(DX(I+1)/PERM(I+1)+DX(I)/PERM(I))/DX(I)
        TXOP(I)=2.*LAMOP/(DX(I+1)/PERM(I+1)+DX(I)/PERM(I))/DX(I)
      ELSE
        TXOP(I)=2.*LAMOP/DX(I)*PERM(I)/DX(I)
        TXWP(I)=2.*LAMWP/DX(I)*PERM(I)/DX(I)
      ENDIF
C-----------------------------------STORAGE COEFFICIENTS
      CPOO(I)=(1.-SW(I))*PHI(I)*(CR*BO(I)+DBO(I))/DT
      CSWO(I)=-PHI(I)*BO(I)/DT
      CPOW(I)=SW(I)*PHI(I)*(CR*BW(I)+DBW(I))/DT
      CSWW(I)=PHI(I)*BW(I)/DT-CPOW(I)*DPCOW(I)
C-----------------------------------MATRIX COEFFICIENTS
      ALFA=-CSWO(I)/CSWW(I)
      A(I)=TXOM(I)+ALFA*TXWM(I)
      C(I)=TXOP(I)+ALFA*TXWP(I)
      IF(I.NE.1)THEN

        B(I)=-(TXOP(I)+TXOM(I)+CPOO(I))
     *       -(TXWP(I)+TXWM(I)+CPOW(I))*ALFA
      ENDIF
      IF(I.NE.N.AND.I.NE.1)THEN
        D(I)=-(CPOO(I)+ALFA*CPOW(I))*PO(I)
     *       +ALFA*(TXWP(I)*(PCOW(I+1)-PCOW(I))
     *       +TXWM(I)*(PCOW(I-1)-PCOW(I)))
      ENDIF
      IF(I.EQ.1.AND.IBC.EQ.2)THEN
        D(I)=-(CPOO(I)+ALFA*CPOW(I))*PO(I)
     *     +ALFA*(TXWP(I)*(PCOW(I+1)-PCOW(I))+QWI/DX(I)/AREA)
        B(I)=-(TXOP(I)+CPOO(I))
     *       -(TXWP(I)+CPOW(I))*ALFA
      ENDIF
      IF(I.EQ.1.AND.IBC.EQ.1)THEN
        D(I)=-(CPOO(I)+ALFA*CPOW(I))*PO(I)
     *     +ALFA*(TXWP(I)*(PCOW(I+1)-PCOW(I))+TXWM(I)*(PL+PCOW(1)))
        B(I)=-(TXOP(I)+CPOO(I))
     *               -(TXWP(I)+TXWM(I)+CPOW(I))*ALFA
      ENDIF
      IF(I.EQ.N)THEN
        D(I)=-(CPOO(I)+ALFA*CPOW(I))*PO(I)
     *       -(TXOP(I)+ALFA*TXWP(I))*PR
     *       +ALFA*TXWM(I)*(PCOW(I-1)-PCOW(I))
      ENDIF
  200 CONTINUE
C
C----PRESSURE SOLUTION
C
      CALL TRIDIA(N,A,B,C,D,PONEW)
C
C----SATURATION SOLUTION
C
      DO 300 I=1,N
      IF(I.NE.N.AND.I.NE.1)SW(I)=SW(I)+(TXOP(I)*(PONEW(I+1)-PONEW(I))
     *     +TXOM(I)*(PONEW(I-1)-PONEW(I))
     *     -CPOO(I)*(PONEW(I)-PO(I)))/CSWO(I)
      IF(I.EQ.N)SW(I)=SW(I)+(TXOP(I)*(PR-PONEW(I))
     *     +TXOM(I)*(PONEW(I-1)-PONEW(I))
     *     -CPOO(I)*(PONEW(I)-PO(I)))/CSWO(I)
      IF(I.EQ.1)SW(I)=SW(I)+(TXOP(I)*(PONEW(I+1)-PONEW(I))
     *     -CPOO(I)*(PONEW(I)-PO(I)))/CSWO(I)
  300 CONTINUE
C
C----UPDATE PRESSURES
C
      DO 400 I=1,N
      PO(I)=PONEW(I)
  400 CONTINUE
C
C----COMPUTE PL IF IBC=2
C
      IF(IBC.EQ.2)PL=PO(1)-PCOW(1)-QWI/DX(1)/AREA/TXWM(1)
C
C----COMPUTE QWI IF IBC=1
C
      IF(IBC.EQ.1)QWI=(PO(1)-PCOW(1)-PL)*DX(1)*AREA*TXWM(1)
C
C----COMPUTE QOP, QWP AND WC
C
      QOP=-(PR-PO(N))*DX(N)*AREA*TXOP(N)
      QWP=-(PR-PO(N)+PCOW(N))*DX(N)*AREA*TXWP(N)
      WC=QWP/(QWP+QOP)
C
 1000 CONTINUE
 1001 CONTINUE
      END
C
C----SUBROUTINE FOR INTERPOLATION OF DATA FROM INPUT TABLE
C
      SUBROUTINE INTERP(X,Y,DY,ISW,N,XT,YT)
C
C------------------------------------------------------------------
C     THE ROUTINE SEARCES IN THE TABLE AND CONDUCTS LINEAR INTERPO-
C     LATION IN ORDER TO DETERMINE Y IN POINT X, AND ITS DERIVATIVE
C     IF ISW.NE.0
C     IF THE ARGUMENT (X) IS OUTSIDE THE TABLE, ENPOINTS ARE USED
C
C     X=ARGUMENT
C     Y=INTERPOLATED VALUE
C     DY=COMPUDED DERIVATIVE OF Y
C     ISW=0 IF DERIVATIVE IS NOT TO BE COMPUTED
C     N=NUMBER OF ENTRIES IN TABLE
C     XT=INDEPENDENT TABLE VARIABLE
```

```
C     YT=DEPENDENT TABLE VARIABLE
C------------------------------------------------------------------
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 X,Y,DY,XT(100),YT(100)
C
C----IF X IS GREATER THAN LARGEST TABLE VALUE
C
      IF(X.LT.XT(N)) GOTO 10
      Y=YT(N)
      IF(ISW.NE.0)DY=(YT(N)-YT(N-1))/(XT(N)-XT(N-1))
      RETURN
C
C----IF X IS LESS THAN THE SMALLEST TABLE VALUE
C
   10 IF(X.GT.XT(1)) GOTO 11
      Y=YT(1)
      IF(ISW.NE.0)DY=(YT(2)-YT(1))/(XT(2)-XT(1))
      RETURN
C
C----IN GENERAL
C
  11  DO 20 I=2,N
      IF(X.GE.XT(I)) GO TO 20
      Y=YT(I-1) +(X-XT(I-1))*(YT(I)-YT(I-1))/(XT(I)-XT(I-1))
      IF(ISW.NE.0)DY=(YT(I)-YT(I-1))/(XT(I)-XT(I-1))
      RETURN
   20 CONTINUE
      RETURN
      END
C
C----SUBROUTINE FOR GAUSSIAN ELIMINATION
C
      SUBROUTINE TRIDIA(N,A,B,C,D,P)
C
C------------------------------------------------------------------
C     THE SUBROUTINE USES GAUSSIAN ELIMINATION FOR SOLUTION OF THE
C     SET OF EQUATIONS
C
C     A(I)*P(I-1) + B(I)*P(I) + C(I)*P(I+1) = D(I)
C
C     A(I),B(I),C(I),D(I)=MATRIX COEFFICIENTS
C     P=PRESSURE
C     N=NUMBER OF EQUATIONS
C------------------------------------------------------------------
C
      REAL*8 A(100),B(100),C(100),D(100),P(100),BB(100),DD(100),X
      BB(1)=B(1)
      DD(1)=D(1)
      DO 60 I=2,N
      X=A(I)/BB(I-1)
      BB(I)=B(I)-X*C(I-1)
   60 DD(I)=D(I)-X*DD(I-1)
      P(N)=DD(N)/BB(N)
      DO 70 K=2,N
      I=N-K+1
   70 P(I)=(DD(I)-C(I)*P(I+1))/BB(I)
      RETURN
      END
```