

```

C*****
****
C THIS PROGRAM PERFORMS FLASH CALCULATION USING PENG-ROBINSON EOS
C BY SS METHOD AND CALCULATES EIGENVALUES OF THE ITERATION MATRIX.
C
*****
***
      IMPLICIT REAL*8 (A-H,O-Z), INTEGER (I-N)
      COMMON/CR/TC,VC,PC,W,AMOLW,AM
      COMMON/TAP/T,P,RT,AOA,BOB,T1,P1
      COMMON/CONS/C1,C2,C3,IL,IV,CPS,R,EPS1,EPS2,KMAX,ISTATE
      COMMON/CID/COMP,DIJ,NUMB,APJ
      COMMON/NOS/K,IERR
      DIMENSION ZZ(40),X(40),Y(40),AK(40),ZC(40),AKO(40)
      DIMENSION TC(40),VC(40),PC(40),W(40),AMOLW(40),AM(40)
      DIMENSION FEED(40),NUMB(40),DIJ(40,40),COMP(40,2)
      1  ,APJ(40,40)

C*****
C INPUT AND OUTPUT FILES
      open(unit=20,file='input.dat',status='old',ACCESS='SEQUENTIAL')
      OPEN(UNIT=21,FILE='OUTPUTN.DAT',STATUS='NEW')
      OPEN(UNIT=22,FILE='OUTPUTN1.DAT',STATUS='NEW')
C*****
      IL=1
      IV=0
          C1=1.0+DSQRT(2.0D0)
      C2=2.0-C1
      C3=C2-C1
      CPS=5.372697
      R=8.20597D-5
      EPS1=1.D-4
      EPS2=1.D-5
      KMAX=300
C*****
C INPUT BLOCK
C*****
C NC=NUMBER OF COMPONENTS,
C NUMB=IDENTIFICATION OF COMPONENTS
C FEED=FEED MOLE FRACTION
C*****
      READ(20,*) NC
          READ(20,*) (NUMB(I),I=1,NC)
          READ(20,*) (FEED(I),I=1,NC)
          CALL EOS(NC)

      TOTAL=0.0
      DO 30 I=1,NC
30  TOTAL=TOTAL+FEED(I)
      DO 40 I=1,NC
40  ZZ(I)=FEED(I)/TOTAL
      IERR=0
C*****

```

```

95     TYPE*, 'P1 IN KPA', 'T1 IN K'
      TYPE*, 'TO TERMINATE PUT P1=0'
      ACCEPT*, P1, T1
      IF (P1.EQ.0) GO TO 120
C*****
      P11=P1/1000.
      WRITE (22, 9999) T1, P11
9999  FORMAT (//, 2X, 'TEMP.= ', F6.2, 'K', 3X, 'PRESS= ', F6.3, 'MPa')

          P=P1*9.8692327D-03
      T=T1
      CALL APARA (NC)
      RT=R*T
      BOB=P/RT
      AOA=BOB/RT
      DO 90 I=1, NC
90     AK(I)=(DEXP(CPS*(1.+W(I))*(1.-TC(I)/T))*PC(I)/P)
      CALL PHASE1 (NC, X, Y, ZZ, AK, VW)
      IF ((VW.EQ.0.0).OR.(VW.EQ.1.0)) THEN
      CALL HANDS (NC, X, Y, ZZ, AK, VW)
      ENDIF
C*****
C     OUTPUT BLOCK
C*****
      CALL RITE (NC, X, Y, ZZ, AK, VW)
      GO TO 95
120   STOP
      END
      SUBROUTINE HANDS (NC, X, Y, ZZ, AK, VW)
      IMPLICIT REAL*8 (A-H, O-Z), INTEGER (I-N)
      COMMON/CONS/C1, C2, C3, IL, IV, CPS, R, EPS1, EPS2, KMAX, ISTATE
      COMMON/CR/TC, VC, PC, W, AMOLW, AM
      COMMON/TAP/T, P, RT, AOA, BOB, T1, P1
      COMMON/ENT/VL, VV, ZL, ZV
      DIMENSION TC (20), VC (20), PC (20), W (20), AMOLW (20), AM (20)
      DIMENSION ZZ (20), X (20), Y (20), AK (20), SAX (20)
      K=0.0
      IF (VW.EQ.0.) THEN
      ISTATE=1.0
      ELSE
          ISTATE=0.
      ENDIF
      CALL ABMIX (NC, ZZ, AMX, BMX, SAX)
      CALL VOLUME (AMX, BMX, V, Z, ISTATE)
      IMI=IL-ISTATE
      DO 1 I=1, NC
      AK(I)=1.0
      X(I)=ZZ(I)*ISTATE
1     Y(I)=ZZ(I)*IMI
      VW=IMI
      ZL=Z*ISTATE
      VL=V*ISTATE
      ZV=V*IMI
      VV=V*IMI

```

RETURN  
END

```
SUBROUTINE EOS (NC)
  IMPLICIT REAL*8 (A-H,O-Z), INTEGER (I-N)
  COMMON/CR/TC, VC, PC, W, AMOLW, AM
  COMMON/PAR/B
  COMMON/INT/ACIJ
  COMMON/CONS/C1, C2, C3, IL, IV, CPS, R, EPS1, EPS2, KMAX, ISTATE
  COMMON/CID/COMP, DIJ, NUMB
  COMMON/NOS/K, IERR, K1
  DIMENSION COMP (40, 2), TC (40), VC (40), PC (40), W (40), AMOLW (40),
1  NUMB (40), AM (40), AIN (350), DIJ (40, 40), ACIJ (40, 40), AC (40), B (40)
  DIMENSION CN1 (40), CN2 (40), CP (40), CV (40), CT (40), CW (40), CM (40)
  DATA CN1 (1), CN2 (1), CP (1), CT (1), CV (1), CW (1), CM (1) / 'METH',
1  'ANE', 45.80, 190.7, 0.0990, 0.0130, 16.043/
  DATA CN1 (2), CN2 (2), CP (2), CT (2), CV (2), CW (2), CM (2) / 'ETHA',
1  'NE', 48.20, 305.43, 0.148, 0.0986, 30.07/
  DATA CN1 (3), CN2 (3), CP (3), CT (3), CV (3), CW (3), CM (3) / 'PROP',
1  'ANE', 42.01, 369.9, 0.200, 0.1524, 44.097/
  DATA CN1 (4), CN2 (4), CP (4), CT (4), CV (4), CW (4), CM (4) / 'I-BU',
1  'TANE', 36.00, 408.1, 0.263, 0.1848, 58.124/
  DATA CN1 (5), CN2 (5), CP (5), CT (5), CV (5), CW (5), CM (5) / 'N-BU',
1  'TANE', 37.47, 425.2, 0.255, 0.2010, 58.124/
  DATA CN1 (6), CN2 (6), CP (6), CT (6), CV (6), CW (6), CM (6) / 'I-PE',
1  'NTAN', 32.90, 460.4, 0.308, 0.2223, 72.151/

  DATA CN1 (7), CN2 (7), CP (7), CT (7), CV (7), CW (7), CM (7) / 'N-PE',
1  'NTAN', 33.31, 469.8, 0.311, 0.2539, 72.151/
  DATA CN1 (8), CN2 (8), CP (8), CT (8), CV (8), CW (8), CM (8) / 'N-HE',
1  'XANE', 29.92, 507.9, 0.368, 0.3007, 86.178/
  DATA CN1 (9), CN2 (9), CP (9), CT (9), CV (9), CW (9), CM (9) / 'N-HE',
1  'PTAN', 27.01, 540.16, 0.426, 0.3498, 100.205/
  DATA CN1 (10), CN2 (10), CP (10), CT (10), CV (10), CW (10), CM (10) /
1  'N-OC', 'TANE', 24.64, 568.6, 0.486, 0.4018, 114.232/
  DATA CN1 (11), CN2 (11), CP (11), CT (11), CV (11), CW (11), CM (11) /
1  'N-NO', 'NANE', 22.50, 594.6, 0.543, 0.4455, 128.259/
  DATA CN1 (12), CN2 (12), CP (12), CT (12), CV (12), CW (12), CM (12) /
1  'N-DE', 'CANE', 20.80, 617.6, 0.602, 0.4885, 142.286/
  DATA CN1 (13), CN2 (13), CP (13), CT (13), CV (13), CW (13), CM (13) /
1  'N2', ' ', 33.50, 126.2, 0.090, 0.040, 28.013/
  DATA CN1 (14), CN2 (14), CP (14), CT (14), CV (14), CW (14), CM (14) /
1  'CO2', ' ', 72.90, 304.2, 0.094, 0.225, 44.01/
  DATA CN1 (15), CN2 (15), CP (15), CT (15), CV (15), CW (15), CM (15) /
1  'H2S', ' ', 88.90, 373.6, 0.098, 0.100, 34.076/
  DATA CN1 (16), CN2 (16), CP (16), CT (16), CV (16), CW (16), CM (16) /
1  'TOLU', 'ENE ', 41.60, 592.0, 0.316, 0.2596, 92.13/
  DATA CN1 (17), CN2 (17), CP (17), CT (17), CV (17), CW (17), CM (17) /
1  'BENZ', 'ENE ', 48.60, 562.1, 0.260, 0.215, 78.11/
  DATA CN1 (18), CN2 (18), CP (18), CT (18), CV (18), CW (18), CM (18) /
1  'CYCL', 'O-C6', 40.0, 553.2, 0.308, 0.2133, 84.16/
  DATA CN1 (19), CN2 (19), CP (19), CT (19), CV (19), CW (19), CM (19) /
1  'H2O', ' ', 218.3, 647.0, 0.056, 0.348, 18.02/
  DATA CN1 (20), CN2 (20), CP (20), CT (20), CV (20), CW (20), CM (20) /
```

```

1  'HYDR', 'GANE', 20.20, 43.6, 0.0, 0.0, 2.016/
   DATA CN1(21), CN2(21), CP(21), CT(21), CV(21), CW(21), CM(21) /
1  'CO', ' ', 34.53, 132.93, 0.0, .0520, 28.01/
   DATA CN1(22), CN2(22), CP(22), CT(22), CV(22), CW(22), CM(22) /
1  'NH3', ' ', 111.4, 405.11, 0.0, 0.26, 17.03/
   DATA CN1(23), CN2(23), CP(23), CT(23), CV(23), CW(23), CM(23) /
1  'METH', 'ANOL', 79.9, 512.2, 0.0, 0.559, 32.042/
   DATA CN1(24), CN2(24), CP(24), CT(24), CV(24), CW(24), CM(24) /
1  'HEL', 'IUM', 6.67, 10.47, 0.0, 0.0, 4.0026/
   DATA CN1(25), CN2(25), CP(25), CT(25), CV(25), CW(25), CM(25) /
1  'BITU', 'MEN', 6.834, 911.48, 0.0, 1.6591, 544.0/

   DATA CN1(26), CN2(26), CP(26), CT(26), CV(26), CW(26), CM(26) /
1  'TETR', 'DCAN', 16.02, 694.72, 1.0934, 0.6505, 198.327/
   DATA (AIN(I), I=1, 325) / 66*0.0
*      , 0.036, 0.050, 0.080, 0.095, 0.090, 0.095
*      , 7*0.1, 0.130, 0.135, 2*0.13, 3*0.125, 0.1, 0.115, 2*0.11, -0.02, 0.085
*      , 0.084, 0.075, 0.05, 2*0.06, 0.065, 2*0.06, 0.055, 0.05, 0.045, 0.18,
*      0.1, 0.04, 2*0.02, 8*0.0, 0.01, 0.18, 0.09, 0.0, 0.04, 2*0.02, 8*0.0,
*      0.01, 0.16, 0.075, 2*0.0, 0.035, 0.02, 0.02, 8*0.0, 0.01, 2*0.1,
*      3*0.0, 2*0.05, 0.55, 2*0.56, 7*0.48, 0.508, 0.21, 0.164, 3*0.48, -0.15
*      , 11*-0.5, -0.036, -0.3, 0.1, 3*-0.5, 0., 0.02, .026, 0.03, 9*0.04,
*      .012, -0.03, 0.1, 0.105, 0.11, 2*0.1, -0.036, 0.18, .2, 10*0.2, .24,
*      2*0, 3*0.08, -.27, -.183, 0, 0.2, 0.02, .03, 2*.05, 2*.1, 5*.15, .5, -.2,
*      .0, 3*.2, -.08, 15*0., -.4, .4, 5*0., .55, 3*0., .115, .025, 11*0, .095,
*      4*0, .135, 5*0.0, 12*0.0, 2*.1, 0.05, 3*0.0, 0.45, -0.5, 0.04, 2*0.2,
*      -0.5, 0./
   IF(NC.EQ.0.0) GO TO 11
   DO 1 I=1, NC
   J=NUMB(I)
   COMP(I, 1)=CN1(J)
   COMP(I, 2)=CN2(J)
   PC(I)=CP(J)
   TC(I)=CT(J)
   VC(I)=CV(J)
   W(I)=CW(J)
   AMOLW(I)=CM(J)
1  CONTINUE
   NM=NC-1
   IF(NM.EQ.0) GO TO 11
   DO 3 II=1, NM
   I=NUMB(II)
   IP1=II+1.
   DO 3 JJ=IP1, NC
   J=NUMB(JJ)
   IF(J.LT.I) THEN
       KIJ=J+(I-1)*(I-2)/2
       DIJ(JJ, II)=AIN(KIJ)
   ELSE
       KIJ=I+(J-1)*(J-2)/2
       DIJ(JJ, II)=AIN(KIJ)
   ENDIF
3  CONTINUE
11 CONTINUE

```

```

DO 6 I=1,NC
RTC=R*TC(I)
RTP=RTC/PC(I)
AC(I)=0.457235*RTP*RTC
B(I)=0.077796*RTP
AM(I)=0.37464+(1.54226-0.26992*W(I))*W(I)
ACIJ(I,I)=AC(I)
6 DIJ(I,I)=0.0
IF(NC.EQ.1)RETURN
DO 10 I=2,NC
IM1=I-1
DO 10 J=1,IM1
ACIJ(I,J)=(1.-DIJ(I,J))*DSQRT(AC(I)*AC(J))
DIJ(J,I)=DIJ(I,J)
10 ACIJ(J,I)=ACIJ(I,J)

RETURN
END

```

```

SUBROUTINE APARA(NC)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
COMMON/TAP/T,P,RT,AOA,BOB,T1,P1
COMMON/CR/TC,VC,PC,W,AMOLW,AM
COMMON/INT/ACIJ
COMMON/AS/AIJ,ALFSQ
COMMON/NOS/K,IERR,K1
DIMENSION TC(40),VC(40),PC(40),W(40),AMOLW(40),AM(40)
DIMENSION ALFSQ(40),ACIJ(40,40),AIJ(40,40)
DO 1 I=1,NC
ALFSQ(I)=1.+AM(I)*(1.-DSQRT(T/TC(I)))
1 AIJ(I,I)=ACIJ(I,I)*ALFSQ(I)*ALFSQ(I)
IF(NC.EQ.1.)RETURN
DO 2 I=2,NC
IM1=I-1
DO 2 J=1,IM1
AIJ(I,J)=ACIJ(I,J)*ALFSQ(I)*ALFSQ(J)
2 AIJ(J,I)=AIJ(I,J)
RETURN
END

```

```

SUBROUTINE ABMIX(NC,X,AM,BM,SAM)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
COMMON/AS/AIJ,ALFSQ
COMMON/NOS/K,IERR
COMMON/PAR/B
DIMENSION AIJ(40,40),X(40),SAM(40),ALFSQ(40),B(40)
AM=0.0
BM=0.0
DO 2 I=1,NC

```

```

BM=BM+X(I)*B(I)
SAM(I)=0.0
DO 1 J=1,NC
1 SAM(I)=SAM(I)+X(J)*AIJ(I,J)
AM=AM+X(I)*SAM(I)
2 CONTINUE
RETURN
END

```

```

SUBROUTINE VOLUME (AMX, BMX, V, Z, IFASE)
IMPLICIT REAL*8 (A-H, O-Z), INTEGER (I-N)
COMMON/TAP/T, P, RT, AOA, BOB, T1, P1
COMMON/AB/AA, BB
DIMENSION ZR(3), CU(3)
AA=AMX*AOA
BB=BMX*BOB
CU(1)=BB-1.
CU(2)=AA-BB*(2.+3.*BB)
CU(3)=BB*(BB*BB+BB-AA)
CALL CUBEQ (IRT, ZR, CU)
IF (IRT) 1, 1, 6
1 IF (IFASE) 2, 2, 3
2 Z=DMAX1 (ZR(1), ZR(2), ZR(3))
GO TO 99
3 ZDUM=100.
DO 5 I=1, 3
IF (ZR(I)) 4, 4, 5
4 ZR(I)=ZDUM
5 CONTINUE
Z=DMIN1 (ZR(1), ZR(2), ZR(3))
GO TO 99
6 Z=ZR(1)
99 V=Z/BOB
RETURN
END

```

```

SUBROUTINE CUBEQ (IROOT, Z, B)
IMPLICIT REAL*8 (A-H, O-Z), INTEGER (I-N)
DIMENSION B(3), Z(3)
TRD=1./3.
B13=B(1)*TRD
BET=B(3)+B13*(2.*B13*B13-B(2))
BO2=0.5*BET
AO3=TRD*(B(2)-B(1)*B13)
CUA=AO3*AO3*AO3
SQB=BO2*BO2
DEL=SQB+CUA
IF (DEL) 30, 10, 20
10 IROOT=0
GAM=DSQRT (-AO3)
IF (BET) 12, 12, 11
11 Z(1)=-2.*GAM-B13
Z(2)=GAM-B13
Z(3)=Z(2)

```

```

RETURN
12  Z (1) =2.*GAM-B13
    Z (2) =-GAM-B13
    Z (3) =Z (2)
RETURN
20  IROOT=1
    EPS=DSQRT (DEL)
    TAU=-BO2
    RCU=TAU+EPS
    SCU=TAU-EPS
    SIR=1.0
    SIS=1.0
    IF (RCU) 21, 22, 22
21  SIR=-1.0
22  IF (SCU) 23, 24, 24
23  SIS=-1.0
24  R=SIR*(SIR*RCU)**TRD
    S=SIS*(SIS*SCU)**TRD
    Z (1) =R+S-B13
    Z (2) =-0.5*(R+S)-B13
    Z (3) =0.8660254*(R-S)
RETURN
30  IROOT=-1
    ROOT=DSQRT (-SQB/CUA)
    IF (BET) 32, 31, 31
31  PEI=(1.5707963+DATAN (ROOT/DSQRT (1.-ROOT*ROOT) )) *TRD
    GO TO 33
32  PEI=DATAN (DSQRT (1.-ROOT*ROOT) /ROOT) *TRD
33  FACT=2.*DSQRT (-AO3)
    CZ=2.0943951
    DO 34 I=1, 3
34  Z (I) =FACT*DCOS (PEI+CZ*(I-1) ) -B13
RETURN
END
C*****
C FLASH BY SS METHOD AND CALCULATE ITERATION MATRIX AND ITS
C EIGENVALUES BY DEM AND RIGOROUS METHODS
C*****
SUBROUTINE PHASE1 (NC, X, Y, ZZ, AK, VW)
IMPLICIT REAL*8 (A-H, O-Z), INTEGER (I-N)
COMMON/CONS/C1, C2, C3, IL, IV, CPS, R, EPS1, EPS2, KMAX, ISTATE
COMMON/TAP/T, P, RT, AOA, BOB, T1, P1
COMMON/RF/RATIO
COMMON/CR/TC, VC, PC, W, AMOLW, AM
COMMON/ENT/VL, VV, ZL, ZV
COMMON/NOS/K, IERR
COMMON/SWIT/KL, KV, NR
COMMON/AB/AA, BB
COMMON/BZ/BLN, ZMBB
COMMON/PTXY/ROXY
COMMON/NT/NCP, NCM
DIMENSION X (40), Y (40), ZZ (40), AK (40), TC (40), PC (40), VC (40)
DIMENSION RATIO (40), FL (40), FV (40), SAL (40), SAV (40), W (40),
1  AMOLW (40), AM (40), AKO (40), SRO (0:200), AKL (40, 0:200),

```

```

1    EI (200) , FX (40) , FY (40) , ROXY (40, 40) , ROXX (40, 39) , ROYY (40, 39)
1    , XXZ (40) , DXDK (39, 40) , DYDK (39, 40) , DVDK (40) , WR (40) , WI (40)
    K=0.0
    NCM=NC-1
10   DO 60 I=1,NC
    AKL (I, K)=DLOG (AK (I) )
60   CONTINUE
    CALL FLASH (NC, X, Y, ZZ, AK, VW, DGDV)
    CALL ABMIX (NC, X, AML, BML, SAL)
    CALL VOLUME (AML, BML, VL, ZL, IL)
    IF (VL.LE.BML) GO TO 25
    CALL FUGA (NC, X, AML, BML, ZL, SAL, FL, FX)
    CALL DRVS (NC, X, AML, BML, ZL, SAL, IL)
    DO 26 I=1,NC
    DO 26 J=1,NCM
26   ROXX (I, J)=ROXY (I, J+1)
    CALL ABMIX (NC, Y, AMV, BMV, SAV)
    CALL VOLUME (AMV, BMV, VV, ZV, IV)
    IF (VV.LE.BMV) GO TO 25
    CALL FUGA (NC, Y, AMV, BMV, ZV, SAV, FV, FY)
    CALL DRVS (NC, Y, AMV, BMV, ZV, SAV, IV)
    DO 28 I=1,NC
    DO 28 J=1,NCM
28   ROYY (I, J)=ROXY (I, J+1)
    SRO (K)=0.
    DO 5 I=1,NC
    RATIO (I)=FL (I) /FV (I)
    AK (I)=AK (I) *RATIO (I)
5    SRO (K)=SRO (K) + ( (1-RATIO (I) ) **2.)
    SRO (K)=SRO (K) /NC
    DO 37 I=1,NC
    XXZ (I)=X (I) *X (I) /ZZ (I)
    DVDK (I)=XXZ (I) /DGDV
37   CONTINUE
    DO 38 L=1,NCM
    DXK=-X (L) * (Y (L) -X (L) ) /ZZ (L)
    AKK=AK (L)
    DO 381 J=1,NC
    DXDK (L, J)=DXK *DVDK (J)
381  DYDK (L, J)=AKK *DXDK (L, J)
    DXDK (L, L)=DXDK (L, L) -XXZ (L) *VW
38   DYDK (L, L)=DYDK (L, L) +XXZ (L) * (1.-VW)
    DO 39 I=1,NC
    DO 391 J=1,NC
    RXK=0
    RYK=0.
    DO 392 L=1,NCM
    RXK=RXK+ROXX (I, L) *DXDK (L, J)
392  RYK=RYK+ROYY (I, L) *DYDK (L, J)
391  ROXY (I, J) = (RXK-RYK) *AK (J)
39   CONTINUE
    CALL HESS (ROXY, NC, WR, WI)
    WRITE (22, 1000) K, SRO (K)
1000 FORMAT (//, 2X, 'K=' , I3, 3X, 'ERROR=' , E15.5)

```

```

        write(22,6000)
6000  format(//,2x,'A(I,J)= Iteration Matrix')
        do 199 i=1,nc
        write(22,5000) (Roxy(i,j),j=1,nc)
199   continue
5000  format(/,2x,7E11.3)
        WRITE(22,3000)
3000  FORMAT(/,2X,'REAL ROOT',3X,'IMAGINARY ROOT')
        DO 99 I=1,NC
        WRITE(22,2000)WR(I),WI(I)
99    CONTINUE
2000  FORMAT(/,2X,E10.4,3X,E10.4)

        IF(K.LT.1) THEN
        K=K+1
        GO TO 10
        ENDIF
        DLTN=0.
        DLTD=0.
        DO 6 I=1,NC
        DLTN=DLTN+(DLOG(RATIO(I)))**2
        DLTD=DLTD+(AKL(I,K)-AKL(I,K-1))**2
6     CONTINUE
        EI(K)=(DLTN/DLTD)**0.5
        WRITE(22,4000)K,ei(k)
4000  format(/,2x,'K=',I3,3X,'LAMBDA=',F7.4)
        IF(SRO(K).Lt.1E-12)RETURN
        IF(K.GT.KMAX)GO TO 21
        k=k+1
        GO TO 10

21    IERR=1
25    IERR=2

        RETURN
        END

SUBROUTINE DRVS(NC,X,AMX,BMX,Z,SA,IFASE)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
COMMON/CR/TC,VC,PC,W,AMOLW,AM
COMMON/AS/AIJ,ALFSQ
COMMON/TAP/T,P,RT,AOA,BOB,T1,P1
COMMON/CONS/C1,C2,C3,IL,IV,CPS,R,EPS1,EPS2,KMAX,ISTATE
COMMON/NOS/K,IERR,K1
COMMON/AB/AA,BB
COMMON/SST/SAB,TEMP
COMMON/PAR/B
COMMON/PTXY/ROXY
COMMON/BZ/BLN,ZMBB
COMMON/NT/NCP,NCM
DIMENSION AM(40),TC(40),VC(40),PC(40),W(40),AMOLW(40)
DIMENSION ALFSQ(40),ACIJ(40,40),AIJ(40,40),B(40),ROXY(40,41)
DIMENSION X(40),SA(40),SAB(40),TEMP(40)

```

```

U0=AA-(3*BB+2.)*BB
U3=AA/BB/(Z+C1*BB)/(Z+C2*BB)
U5=6.*BB+2.
Z50=Z*(Z-U5)-U0
DENO=Z*(3.*Z-2.*(1.-BB))+U0
R3=2.*BLN/AMX
DO 70 J=1,NCM
AXY=2.*(SA(J)-SA(NC))*AOA
BXY=(B(J)-B(NC))*BOB
ZXY=AXY*ZMBB+BXY*Z50
ZXY=-ZXY/DENO
R1=BXY/BB
R2=(ZXY-BXY)/ZMBB
R4=U3*(Z*BXY-BB*ZXY)
R5=((SAB(J)-SAB(NC))*BLN+ZXY)/BMX
L=J+1
DO 70 I=1,NC
ROXY(I,L)=R1*TEMP(I)+R2+R3*(AIJ(J,I)-AIJ(NC,I))
1 +R4*SAB(I)-R5*B(I)
70 ROXY(I,L)=-ROXY(I,L)
RETURN
END
C*****
C EIGENVALUE CALCULATION BY HESSENBERG METHOD
C*****
SUBROUTINE HESS(A,N,WR,WI)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
DIMENSION A(40,40),wr(40),wi(40)
IF(N.GT.2)THEN
DO 2 M=2,N-1
X=0.
I=M
DO 3 J=M,N
IF(ABS(A(J,M-1)).GT.ABS(X))THEN
X=A(J,M-1)
I=J
ENDIF
3 CONTINUE
IF(I.NE.M)THEN
DO 4 J=M-1,N
Y=A(I,J)
A(I,J)=A(M,J)
A(M,J)=Y
4 CONTINUE
DO 5 J=1,N
Y=A(J,I)
A(J,I)=A(J,M)
A(J,M)=Y
5 CONTINUE
ENDIF
IF(X.NE.0.)THEN
DO 6 I=M+1,N
Y=A(I,M-1)
IF(Y.NE.0.)THEN

```

```

Y=Y/X
A(I,M-1)=Y
DO 7 J=M,N
A(I,J)=A(I,J)-Y*A(M,J)
7 CONTINUE
DO 8 J=1,N
A(J,M)=A(J,M)+Y*A(J,I)
8 CONTINUE
ENDIF
6 CONTINUE
ENDIF
2 CONTINUE
ENDIF
CALL EIGEN(A,N,WR,WI)
RETURN
END
SUBROUTINE EIGEN(A,N,WR,WI)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
DIMENSION A(40,40),WR(40),WI(40)
ANORM= DABS(A(1,1))
DO 12 I=2,N
DO 11 J=I-1,N
ANORM=ANORM+DABS(A(I,J))
11 CONTINUE
12 CONTINUE
NN=N
T=0.
1 IF(NN.GE.1) THEN
ITS=0
2 DO 13 L=NN,2,-1
S=DABS(A(L-1,L-1))+DABS(A(L,L))
IF(S.EQ.0.)S=ANORM
IF(DABS(A(L,L-1))+S.EQ.S)GO TO 3
13 CONTINUE
L=1
3 X=A(NN,NN)
IF(L.EQ.NN) THEN
WR(NN)=X+T
WI(NN)=0.
NN=NN-1
ELSE
Y=A(NN-1,NN-1)
W=A(NN,NN-1)*A(NN-1,NN)
IF(L.EQ.NN-1) THEN
P=0.5*(Y-X)
Q=P**2+W
Z=DSQRT(DABS(Q))
X=X+T
IF(Q.GE.0.) THEN
Z=P+SIGN(Z,P)
WR(NN)=X+Z
WR(NN-1)=WR(NN)
IF(Z.NE.0.)WR(NN)=X-W/Z
WI(NN)=0.

```

```

WI (NN-1)=0.
ELSE
WR (NN)=X+P
WR (NN-1)=WR (NN)
WI (NN)=Z
WI (NN-1)=-Z
ENDIF
NN=NN-2
ELSE
IF (ITS.EQ.30) return
IF (ITS.EQ.10.OR.ITS.EQ.20) then
t=t+x
do 14 i=1,nn
a (i,i)=a (i,i)-x
14 CONTINUE
S=DABS (A (NN,NN-1)) +DABS (A (NN-1,NN-2))
X=0.75*S
Y=X
W=-0.4375*S**2
ENDIF
ITS=ITS+1
DO 15 M=NN-2,L,-1
Z=A (M,M)
R=X-Z
S=Y-Z
P=(R*S-W)/A (M+1,M) +A (M,M+1)
Q=A (M+1,M+1) -Z-R-S
R=A (M+2,M+1)
S=DABS (P) +DABS (Q) +DABS (R)
P=P/S
Q=Q/S
R=R/S
IF (M.EQ.L) GO TO 4
U=DABS (A (M,M-1)) * (DABS (Q) +DABS (R))
V=DABS (P) * (DABS (A (M-1,M-1)) +DABS (Z) +DABS (A (M+1,M+1)))
IF (U+V.EQ.V) GO TO 4
15 CONTINUE
4 DO 16 I=M+2,NN
A (I,I-2)=0.
IF (I.NE.M+2) A (I,I-3)=0.
16 CONTINUE
DO 19 K=M,NN-1
IF (K.NE.m) then
P=A (K,K-1)
Q=A (K+1,K-1)
R=0.
IF (K.NE.NN-1) R=A (K+2,K-1)
X=DABS (P) +DABS (Q) +DABS (R)
IF (X.NE.0.) THEN
P=P/X
Q=Q/X
R=R/X
ENDIF
ENDIF
ENDIF

```

```

S=SIGN(DSQRT(P**2+Q**2+R**2),P)
IF(S.NE.0.) THEN
IF(K.EQ.M) THEN
IF(L.NE.M) A(K,K-1)=-A(K,K-1)
ELSE
A(K,K-1)=-S*X
ENDIF
P=P+S
X=P/S
Y=Q/S
Z=R/S
Q=Q/P
R=R/P
DO 17 J=K,NN
P=A(K,J)+Q*A(K+1,J)
IF(K.NE.NN-1) THEN
P=P+R*A(K+2,J)
A(K+2,J)=A(K+2,J)-P*Z
ENDIF
A(K+1,J)=A(K+1,J)-P*Y
A(K,J)=A(K,J)-P*X
17 CONTINUE
DO 18 I=L,MIN(NN,K+3)
P=X*A(I,K)+Y*A(I,K+1)
IF(K.NE.NN-1) THEN
P=P+Z*A(I,K+2)
A(I,K+2)=A(I,K+2)-P*R
ENDIF
A(I,K+1)=A(I,K+1)-P*Q
A(I,K)=A(I,K)-P
18 CONTINUE
ENDIF
19 CONTINUE
GO TO 2
ENDIF
ENDIF
GO TO 1
ENDIF
RETURN
END

```

```

SUBROUTINE FLASH(NC,X,Y,ZZ,AK,VW,DF)
IMPLICIT REAL*8(A-H,O-Z), INTEGER(I-N)
COMMON/CONS/C1,C2,C3,IL,IV,CPS,R,EPS1,EPS2,ISTATE
COMMON/NOS/K,IERR
DIMENSION X(40),Y(40),ZZ(40),AK(40)
SUMX=0.0
DO 1 I=1,NC
X(I)=ZZ(I)/AK(I)
1 SUMX=SUMX+X(I)
UMS=1.-SUMX
IF(UMS.GE.0.0) THEN
VW=1.0

```

```

                DO 22 I=1,NC
                Y(I)=ZZ(I)
                X(I)=X(I)/SUMX
22             CONTINUE
                RETURN

            ENDIF
                SUMY=0.0
                DO 2 I=1,NC
                Y(I)=ZZ(I)*AK(I)
2             SUMY=SUMY+Y(I)
                SMU=SUMY-1.
                IF (SMU.LE.0.) THEN
                    VW=0.0
                    DO 32 I=1,NC
                    X(I)=ZZ(I)
                    Y(I)=Y(I)/SUMY
32             CONTINUE
                    RETURN
                ENDIF
                VW=SMU/(SMU-UMS)
                AVO=VW
10             F=0.0
                DF=0.
                C     SUMX=0.
                C     SUMY=0.
                DO 11 I=1,NC
                X(I)=ZZ(I)/(1.+(AK(I)-1.)*VW)
                C     SUMX=SUMX+X(I)
                Y(I)=AK(I)*X(I)
                C     SUMY=SUMY+Y(I)
                YMX=Y(I)-X(I)
                F=F+YMX
                DF=DF+YMX*YMX/ZZ(I)
11             CONTINUE
                DLV=F/DF
                IF (DABS(DLV).LT.eps2) RETURN
                VW=VW+DLV
                IF (VW.GE.1.0) VW=0.5*(1.0+AVO)
                AVO=VW
                GO TO 10
            END

SUBROUTINE FUGA(NC,X,AMX,BMX,Z,SA,FUG,FG)
IMPLICIT REAL*8(A-H,O-Z),INTEGER(I-N)
COMMON/TAP/T,P,RT,AOA,BOB,T1,P1
COMMON/NOS/K,IERR
COMMON/CONS/C1,C2,C3,IL,IV,CPS,R,EPS1,EPS2,KMAX,ISTATE
COMMON/PAR/B
COMMON/BZ/BLN,ZMBB
COMMON/AB/AA,BB
common/sst/sab,temp
DIMENSION SA(40),SAB(40),TEMP(40),FUG(40),B(40),X(40),FG(40)

```

```

ZMBB=Z-BB
ALN=-DLOG (ZMBB)
BLN=AA*DLOG ( (Z+C2*BB) / (Z+C1*BB) ) /BB/C3
DO 3 I=1,NC
BIOB=B (I) /BMX
SAB (I)=SA (I) *2. /AMX-BIOB
TEMP (I) = (Z-1.) *BIOB-BLN*SAB (I)
FOXP=TEMP (I) +ALN
FG (I) =DEXP (FOXP)
3 FUG (I) =X (I) *P*DEXP (FOXP)
RETURN
END

```

```

SUBROUTINE RITE (NC, X, Y, ZZ, AK, VW, METHOD)
IMPLICIT REAL*8 (A-H, O-Z), INTEGER (I-N)
COMMON/TAP/T, P, RT, AOA, BOB, T1, P1
COMMON/CR/TC, VC, PC, W, AMOLW, AM
COMMON/CID/COMP, DIJ, NUMB
COMMON/RF/RATIO
COMMON/ENT/VL, VV, ZL, ZV
COMMON/NOS/K, IERR
COMMON/SWIT/KL, KV, NR
DIMENSION ZZ (40), X (40), Y (40), AK (40), COMP (40, 2), RATIO (40)
DIMENSION TC (40), VC (40), PC (40), W (40), AMOLW (40), AM (40)
      PSI=P*0.101325
TF=T
VL=VL*1000.
VV=VV*1000.

WRITE (21, 61) PSI, TF
IF (IERR.LE.0) THEN
      WRITE (21, 1000)
1000 FORMAT (2X, 'COMP.', 9X, 'Z', 13X, 'X', 14X, 'Y', 13X, 'Ki', 12x, 'Ri', /)
      WRITE (21, 114) ((COMP (I, J), J=1, 2), ZZ (I), X (I), Y (I), AK (I),
1 RATIO (I), I=1, NC)

WRITE (21, 115) ZL, ZV, VL, VV
WRITE (21, 116) K
      WRITE (21, 117) VW
ELSEIF (IERR.EQ.1.) THEN
WRITE (21, 71)
ELSEIF (IERR.EQ.2.) THEN
WRITE (21, 72)
ELSEIF (IERR.EQ.3) THEN
WRITE (21, 73)
ENDIF
61 FORMAT (1X, 'FLASH CALCULATION AT', F10.2, 'MPa', F10.2, 'DEG. K', /)
71 FORMAT (1X, 'NO. OF ITERATIONS REACHED MAXIMUM')
72 FORMAT (1X, 'UNREALISTIC VOLUME')
73 FORMAT (1X, 'POSSIBLE SINGLE PHASE')
114 FORMAT (1X, 2A4, 5E14.5)

```

