```
C
C ***************************************************************************
C *                                                                         *
C *                          R E F J E T                                    *
C *                                                                         *
C *                       A.H.BRIDLE 25-Mar-81                              *
C *                            Version 1.2                                  *
C *                                                                         *
C * COMPUTES JET REFRACTION FOR JET IN SPHEROIDAL OR CHAN-HENRIKSEN         *
C * ATMOSPHERES AT OBLIQUE ANGLES TO SYMMETRY AXES OF THE ATMOSPHERIC       *
C * PRESSURE DISTRIBUTION. PLOTS INTRINSIC SOURCE SHAPE AND THAT SEEN       *
C * BY OBSERVER AT SPECIFIED ANGLES TO THE PRESSURE SYMMETRY AXES.          *
C *                                                                         *
C * BASED ON ALGORITHMS GIVEN BY HENRIKSEN, VALLEE AND BRIDLE (1981)        *
C *                              ASTROPHYSICAL JOURNAL, IN PRESS            *
C ***************************************************************************
C
      DIMENSION Z(250),Y(250),ZD(250),PZ(250),PY(250),DPZDPY(250)
      REAL M1,M2
C
CODE FOR DEBUGGING ON TERMINAL.  DISABLE BY PUTTING DEBUG=.FALSE.
C                                ENABLE  BY PUTTING DEBUG=.TRUE.
C
      LOGICAL DEBUG
      COMMON/DEBUG/DEBUG
      DEBUG=.FALSE.
C
COMPUTER DATE ACQUISITION TO DOCUMENT EXECUTION
C
      DIMENSION DAT(2)
      CALL DATE(DAT)
      WRITE(5,9) DAT
    9 FORMAT(1H1,'REFRACTED JET COMPUTATION   AHB Version 1.2 coded
     * 25-Mar-81',/,26('='),/,' EXECUTION ON ',2A5)
C
CONSTANTS FOR KPC-->CM and DEG-->RAD CONVERSIONS
C
      OKPC=3.0857E+21
      PI=3.14159
      ODEG=PI/180.
C
C     INPUT SOURCE PARAMETERS ********************************************
C
  999 WRITE(5,10)
   10 FORMAT(' RATIO OF SPECIFIC HEATS IN JET (GAMMA) ?')
   11 FORMAT(F10.3)
      READ(5,11) GAMA
      WRITE(5,12)
   12 FORMAT(' INPUT MODEL CODE: 1=SPHEROID, 2=CH ')
      READ(5,13) MODEL
   13 FORMAT(I1)
      IF (MODEL.EQ.2) GO TO 100
C
C     AS (MODEL.EQ.1) WE PICK UP SPHEROIDAL INPUTS THIS TIME
C
      WRITE(5,14)
   14 FORMAT(' INPUT SPHEROID PARMS: ALPHA, PS/PIG, A2(KPC), A3(KPC)')
      READ(5,11) ALPHA
      READ(5,11) PSOPIG
      READ(5,11) A2
      READ(5,11) A3
      WRITE(5,15)
   15 FORMAT(' INPUT START PARMS: ZS(KPC), YS(KPC), PHIS(DEG)')
      READ(5,11) ZS
```

```
06500         READ(5,11) YS
06600         READ(5,11) PHIS
06700         GO TO 110
06800   C
06900   C     AS (MODEL.EQ.2) PICK UP THE CH INPUTS THIS TIME
07000   C
07100     100 WRITE(5,16)
07200      16 FORMAT(' INPUT CH PARMS: ZS(KPC),M1,ZE(KPC),ZH(KPC),M2')
07300         READ(5,11) ZS
07400         READ(5,11) M1
07500         READ(5,11) ZE
07600         READ(5,11) ZH
07700         READ(5,11) M2
07800         WRITE(5,17)
07900      17 FORMAT(' INPUT START PARMS: YS(KPC), PHIS(DEG)')
08000         READ(5,11) YS
08100         READ(5,11) PHIS
08200   C
08300   CONTINUE GETTING THE SETUP PARAMETERS THAT DON'T DEPEND ON MODEL
08400   C
08500     110 WRITE(5,18)
08600      18 FORMAT(' INPUT INITIAL RATIO OF PRESSURE TO RHO*V*V')
08700         READ(5,11) PSORVS
08800         WRITE(5,19)
08900      19 FORMAT(' INPUT ITERATION PARMS: STEP(KPC), # OF STEPS (<250)')
09000         READ(5,11) STEP
09100         READ(5,20) NPTS
09200      20 FORMAT(I3)
09300         WRITE(5,21)
09400      21 FORMAT(' INPUT ANGLES TO LINE OF SIGHT: THETA(DEG), PHI(DEG)')
09500         READ(5,11) THETA
09600         READ(5,11) PHI
09700   C
09800   C     END OF PARAMETER INPUTS ****************************************
09900   C
10000   C
10100   CHECK THE INPUTS ON THE TERMINAL
10200   C
10300         IF(MODEL.EQ.2) GO TO 120
10400         WRITE(5,22) ALPHA,PSOPIG,A2,A3,ZS,YS,PHIS
10500      22 FORMAT(' SPHEROIDAL ATMOSPHERE SPECIFIED AS FOLLOWS:',/,
10600        *       ' PRESSURE INDEX ALPHA                          ',F10.3,/,
10700        *       ' PRESSURE RATIO PS/PIG                         ',F10.3,/,
10800        *       ' PRINCIPAL AXIS A2 (KPC)                       ',F10.3,/,
10900        *       ' MINOR AXIS A3 (KPC)                           ',F10.3,//,
11000        *       ' INITIAL PARAMETERS OF JET:',/,
11100        *       ' OFFSETS FROM PRESSURE CENTER, ZS,YS (KPC)  ',F10.3,F9.3,/,
11200        *       ' ANGLE TO SYMMETRY PLANE OF PRESSURE (DEG)  ',F10.3)
11300         GO TO 125
11400   C
11500   CH MODEL PARAMETERS IF (MODEL.EQ.2)
11600   C
11700     120 WRITE(5,23) ZS,M1,ZE,ZH,M2,PHIS
11800      23 FORMAT(' CH ATMOSPHERE SPECIFIED AS FOLLOWS:          ',/,
11900        *       ' SONIC HEIGHT ZS (KPC)                         ',F10.3,/,
12000        *       ' FIRST PRESSURE INDEX                          ',F10.3,/,
12100        *       ' EQUAL-PRESSURE HEIGHT ZE (KPC)                ',F10.3,/,
12200        *       ' SCALE HEIGHT OF SECOND PRESSURE ZH (KPC)      ',F10.3,/,
12300        *       ' SECOND PRESSURE INDEX                         ',F10.3,//,
12400        *       ' INITIAL PARAMETERS OF JET:',/,
12500        *       ' ANGLE TO SYMMETRY PLANE OF PRESSURE (DEG)  ',F10.3)
12600   C
12700   CONTINUE WITH PARAMETERS IN COMMON TO BOTH MODELS
12800   C
```

```
  125 WRITE(5,24) PSORVS,GAMA,NPTS,STEP,THETA,PHI
   24 FORMAT(' RATIO OF INITIAL PRESSURE TO RHO*VSQ IN JET',F10.3,/,
     *       ' RATIO OF SPECIFIC HEATS IN JET            ',F10.3,//,
     * I4,' ITERATIONS OF STEP',F10.3,' KPC WILL BE CALCULATED',//,
     * ' THE OBSERVER IS AT POLAR ANGLE THETA (DEG)',F10.3,/,
     * '                    AND AZIMUTH PHI (DEG)  ',F10.3,
     * ' IN THE PRESSURE COORDINATE SYSTEM')
C
CHECK WHETHER WE GOT THE INPUTS CORRECT
C
      WRITE(5,25)
   25 FORMAT(' INPUT 1 TO CONTINUE, 0 TO RE-ENTER PARMS')
      READ(5,13) ICHEK
      IF(ICHEK.EQ.0) GO TO 999
C
CHECKED INPUTS OK ON TERMINAL, NOW PRINT THEM OUT
C
      WRITE(3,9) DAT
      IF(MODEL.EQ.2) GO TO 130
      WRITE(3,22) ALPHA,PSOPIG,A2,A3,ZS,YS,PHIS
      GO TO 135
  130 WRITE(3,23) ZS,M1,ZE,ZH,M2,PHIS
  135 WRITE(3,24) PSORVS,GAMA,NPTS,STEP,THETA,PHI
C
COMPUTATIONS BEGIN HERE (INPUTS COMPLETE AND CHECKED) ******************
C
C
CGS UNITS ARE USED THROUGHOUT - TRANSFORM HERE
C
      A2=A2*OKPC
      A3=A3*OKPC
      ZS=ZS*OKPC
      YS=YS*OKPC
      H=STEP*OKPC
      ZE=ZE*OKPC
      ZH=ZH*OKPC
C
CONVERT ANGLES TO RADIANS FROM DEGREES
C
      PHIS=PHIS*ODEG
      THETA=THETA*ODEG
      PHI=PHI*ODEG
      WRITE(3,33)
   33 FORMAT(1X/1X,79(1H-)/ )
C
COMMENCE INTEGRATION STEPS
C
      IF(DEBUG) WRITE(5,9002)
 9002 FORMAT(' STARTING INTEGRATION')
      CALL INTEG(GAMA,ALPHA,PSOPIG,PSORVS,A2,PHIS,ZS,YS,A3,
     * Z,Y,ZD,H,NPTS,N,M2,M1,ZE,ZH,MODEL)
      N2=N
      IF(MODEL.EQ.2) N2=NPTS
C
CONVERT BACK TO KPC FOR OUTPUT
C
      DO 40 I=1,N2
      Z(I)=Z(I)/OKPC
      Y(I)=Y(I)/OKPC
      PZ(I)=PZ(I)/OKPC
      PY(I)=PY(I)/OKPC
   40 CONTINUE
C
COMPUTE PROJECTION ONTO OBSERVER'S SKY PLANE
```

```
C
      IF(DEBUG) WRITE(5,9001)
 9001 FORMAT(' CALCULATING PROJECTION')
      CALL PROJ(Z,Y,PZ,PY,DPZDPY,N2,THETA,PHI)
      DO 2 I=1,N2
      ZD(I)=ATAN(ZD(I))/ODEG
    2 DPZDPY(I)=ATAN(DPZDPY(I))/ODEG
C
C     PRINT OUT INTRINSIC AND PROJECTED PARAMETERS AT THIS STEP
C
    6 WRITE(3,35) (I,Z(I),Y(I),ZD(I),PZ(I),PY(I),DPZDPY(I), I=1,N2)
   35 FORMAT(' I=',I3,' Z=',E9.2,' Y=',E9.2,' ANG=',F5.1,
     * ' // PZ=',E9.2,' PY=',E9.2,' P.ANG=',F5.1)
C
CALCULATE OUTPUT PLOTS
C
      IF(DEBUG) WRITE(5,9000)
 9000 FORMAT(' STARTING TO COMPUTE PLOTTER OUTPUT')
C
C     FIRST PLOT THE PROJECTED SOURCE (RA,DEC ORIENTATION)
C
      WRITE (3,30)
   30 FORMAT(1H1, 6('+----+----'), '> PROJECTED SOURCE Y')
      CALL PLOTZY(PZ,PY,N2,SCALE)
      TIC=SCALE/58.
      WRITE(3,31) TIC
   31 FORMAT(1H , 6('+----+----'),'>TIC=',F8.2,' KPC')
C
C     THEN PLOT THE INTRINSIC SOURCE
C
      WRITE(3,32)
   32 FORMAT(1H1, 6('+----+----'), '> INTRINSIC SOURCE Y')
      CALL PLOTZY(Z,Y,N2,SCALE)
      TIC=SCALE/58.
      WRITE(3,31) TIC
C
CHECK WHETHER WE WANT TO STOP OR RUN ANOTHER MODEL
C
      WRITE(5,8000)
 8000 FORMAT(' INPUT 0 TO EXIT GRACEFULLY, 1 TO CONTINUE')
      READ(5,13) ISTOP
      IF (ISTOP.EQ.0) GO TO 8999
C
CYCLE BACK TO INPUT STEP IF (ISTOP.EQ.1)
C
      GO TO 999
 8999 STOP
      END
C
C            ****************************************
C            *                                      *
C            *            I N T E G                 *
C            *                                      *
C            ****************************************
C
      SUBROUTINE INTEG(GAMA,ALPHA,PSOPIG,PSORVS,A2,PHIS,ZS,YS,A3,
     * Z,Y,ZD,H,NPTS,N,M2,M1,ZE,ZH,MODEL)
      DIMENSION Z(250),Y(250),ZD(250)
      REAL M1,M2
      COMMON/DEBUG/DEBUG
      IF (MODEL.EQ.2) GO TO 10
C
COMPUTE SIGMAS IF DOING THE SPHEROID (MODEL.EQ.1)
C
```

```
25700          SIGMAS=(ZS/A3)*(ZS/A3) + (YS/A2)*(YS/A2)
25800       10 HHK=1
25900          ZOD=SIN(PHIS)/COS(PHIS)
26000          ZO=ZS
26100          YO=YS
26200          ZODD=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,ZO,YO,ZOD,
26300         * MODEL,ZS,M1,ZE,ZH,M2)
26400          ZODDD=0.0
26500    C
26600    COMMENCE PRE-CYCLING
26700    C
26800          IF(DEBUG) WRITE(5,9000)
26900     9000 FORMAT(' PRECYCLING STEP')
27000          ZM1DD=ZODD-H*ZODDD
27100          ZM2DD=ZODD-2.0*H*ZODDD
27200          ZD(1)=ZOD+(5.0*ZM2DD-16.0*ZM1DD+23.0*ZODD)*H/12.0
27300          Z(1)=ZO+H*ZOD+(4.0*ZODD-ZM1DD)*H*H/6.0
27400          Y(1)=YO+H
27500          Q1=Y(1)
27600          Q2=Z(1)
27700          Q3=ZD(1)
27800          Q4=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,Q2,Q1,Q3,
27900         * MODEL,ZS,M1,ZE,ZH,M2)
28000          ZD(2)=ZD(1)+(5.0*ZM1DD-16.0*ZODD+23.0*Q4)*H/12.0
28100          Z(2)=Z(1)+H*ZD(1)+(4.0*Q4-ZODD)*H*H/6.0
28200          Y(2)=Y(1)+H
28300          Q5=Y(2)
28400          Q6=Z(2)
28500          Q7=ZD(2)
28600          Q8=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,Q6,Q5,Q7,
28700         * MODEL,ZS,M1,ZE,ZH,M2)
28800          ZD(3)=ZD(2)+(5.0*ZODD-16.0*Q4+23.0*Q8)*H/12.0
28900          Z(3)=Z(2)+H*ZD(2)+(4.0*Q8-Q4)*H*H/6.0
29000          Y(3)=Y(2)+H
29100    C
29200    CYCLE THE SOLUTION NOW
29300    C
29400          IF(DEBUG) WRITE(5,9001)
29500     9001 FORMAT(' START CYCLING SOLUTION')
29600          DO 1 J=4,NPTS
29700          I=J-1
29800          IM2=I-2
29900          IM1=I-1
30000          IF(DEBUG) WRITE(5,9002) J
30100     9002 FORMAT(' SOLUTION NOW AT STEP J=',I5)
30200          Q9=Y(IM2)
30300          Q10=Z(IM2)
30400          Q11=ZD(IM2)
30500          Q12=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,Q10,Q9,Q11,
30600         * MODEL,ZS,M1,ZE,ZH,M2)
30700          Q13=Y(IM1)
30800          Q14=Z(IM1)
30900          Q15=ZD(IM1)
31000          Q16=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,Q14,Q13,Q15,
31100         * MODEL,ZS,M1,ZE,ZH,M2)
31200          Q17=Y(I)
31300          Q18=Z(I)
31400          Q19=ZD(I)
31500          Q20=d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,Q18,Q17,Q19,
31600         * MODEL,ZS,M1,ZE,ZH,M2)
31700          Y(J)=Y(I)+H
31800          ZD(J)=ZD(I)+(5.0*Q12-16.0*Q16+23*Q20)*H/12.0
31900          Z(J)=Z(I)+H*ZD(I)+(4.0*Q20-Q16)*H*H/6.0
32000        1 CONTINUE
```

```
32100         N=NPTS
32200         RETURN
32300       2 N=J-1
32400         RETURN
32500         END
32600   C
32700   C               ****************************************
32800   C               *                                      *
32900   C               *           d 2 Z / d r 2              *
33000   C               *                                      *
33100   C               ****************************************
33200   C
33300   COMPUTES SECOND DERIVATIVE OF JET PATH, d2Zdr2, = Z" OF HVB PAPER
33400   C
33500   C      PNORM IS EXTERNAL PRESSURE IN UNITS OF SONIC PRESSURE
33600   C
33700         FUNCTION d2Zdr2(HH,GAMA,ALPHA,PSOPIG,PSORVS,A2,A3,SIGMAS,
33800        * Z1,Y1,ZD1,MODEL,ZS,M1,ZE,ZH,M2)
33900         REAL M1,M2
34000         IF(MODEL.EQ.2) GO TO 2
34100   C
34200   COMPUTE SPHEROIDAL MODEL IF (MODEL.EQ.1)
34300   C
34400         SIGMA=(Z1/A3)*(Z1/A3)+(Y1/A2)*(Y1/A2)
34500         HH=SIGMA
34600   C
34700   COMPUTE NORMALISED PRESSURE AT (Z1,Y1)
34800   C
34900         PNORM=(1.0-SIGMAS**ALPHA)/(1.0-SIGMA**ALPHA+PSOPIG*
35000        * (SIGMA**ALPHA-SIGMAS**ALPHA))
35100         AM1=ALPHA-1.0
35200         GMFCT=2.0-1.0/GAMA
35300   C
35400   COMPUTE SECOND DERIVATIVE OF JET PATH
35500   C
35600         d2Zdr2=(-2.0*ALPHA/(1.0-SIGMAS**ALPHA))*PSORVS*(PSOPIG-1.0)*
35700        * (SIGMA**AM1)*(PNORM**GMFCT)*(1.0+ZD1*ZD1)*((ZD1/A2)*(Y1/A2)-
35800        * Z1/A3/A3)
35900         RETURN
36000   C
36100   C ***************************************************************************
36200   C
36300   COMPUTE CH SLAB MODEL IF (MODEL.EQ.2)
36400   C
36500       2 FF=GAMA/(GAMA-1.0)
36600         F = (0.5*GAMA+0.5)**FF
36700         FM1=F-1.0
36800         DEN2=1.0+(Z1/ZH)**M2
36900         PEQF=(ZS/ZE)**M1
37000         ZNORM=Z1/ZS
37100         DEN1=1.0+FM1*(ZNORM**M1)
37200   C
37300   COMPUTE NORMALISED PRESSURE AT HEIGHT Z1
37400   C
37500         PNORM=F/DEN1+(1.0-1.0/ZNORM)*(F/FM1)*PEQF/DEN2
37600         RECGAM=1.0/GAMA
37700         DPDZ=(-F/FM1)*((M1*FM1*FM1*(ZNORM**(M1-1.0))/
37800        * ZS/DEN1/DEN1)+((1.0-1.0/ZNORM)*PEQF*(M2/ZH)*((Z1/ZH)**(M2-
37900        * 1.0))/DEN2/DEN2)+(-ZS/Z1/Z1*PEQF/DEN2))
38000   C
38100   COMPUTE SECOND DERIVATIVE OF JET PATH
38200   C
38300         d2Zdr2=-(1.0+ZD1*ZD1)*PSORVS/(PNORM**RECGAM)*DPDZ
38400         RETURN
```

```
      END
C
C              ****************************************
C              *                                      *
C              *            P L O T Z Y               *
C              *                                      *
C              ****************************************
C
C
CONVERTS (Z,Y) COORDINATE DATA TO LINE PRINTER PLOT IMAGES
C
      SUBROUTINE PLOTZY(Z,Y,NPTS,SCALE)
      DIMENSION Z(250),Y(250),PLOT(58,58)
      COMMON/DEBUG/DEBUG
C
C     DEFINE PLOT SYMBOLS TO BE USED
C
      DATA BLANK,STAR/' ','*'/
C
COMPUTE MAXIMUM IN Z,Y
C
      ZM=ABS(Z(1))
      YM=ABS(Y(1))
      DO 1 I=1,NPTS
      IF (ABS(Z(I)).GT.ZM) ZM=ABS(Z(I))
      IF (ABS(Y(I)).GT.YM) YM=ABS(Y(I))
    1 CONTINUE
C
C     SQUARE OFF THE PLOT SCALE
C
      SCALE=AMAX1(YM,ZM)
C
CLEAR THE PLOT ARRAY
C
      DO 2 I=1,58
      DO 3 J=1,58
      PLOT(I,J)=BLANK
    3 CONTINUE
    2 CONTINUE
C
CONVERT TO 58x58 FORMAT
C
      DO 4 K=1,NPTS
      I=58-INT(Z(K)*58./SCALE+0.5)
      J=INT(Y(K)*58./SCALE+0.5)
C
CHECK THAT INDICES OF PLOT ARRAY ARE IN ALLOWED RANGE (1 TO 58)
C
      IF(I.LT.1) I=1
      IF(I.GT.58) I=58
      IF(J.LT.1) J=1
      IF(J.GT.58) J=58
C
CHARACTER 'STAR' AT EACH OCCUPIED POINT IN PLOT
C
      PLOT(I,J)=STAR
    4 CONTINUE
C
C     ROUTE THE PLOT IMAGE TO LINE PRINTER OUTPUT FILE
C
      DO 15 I=1,58
      WRITE(3,10) (PLOT(I,J), J=1,58)
   10 FORMAT(1H , 58A1)
   15 CONTINUE
```

```
      RETURN
      END
C
C           ****************************************
C           *                                      *
C           *              P R O J                 *
C           *                                      *
C           ****************************************
C
C
CALCULATES PROJECTED COORDINATES PZ,PY AND GRADIENTS DPZDPY FROM
C          INTRINSIC COORDINATES Z,Y
C
      SUBROUTINE PROJ(Z,Y,PZ,PY,DPZDPY,N,THETA,PHI)
      DIMENSION Z(250),Y(250),PZ(250),PY(250),DPZDPY(250)
      COMMON/DEBUG/DEBUG
      PI=3.14159
C
C     FIND Z=0 AXIS IN PROJECTED COORDINATES
C
      PZ0=-Y(1)*COS(THETA)*SIN(PHI)
      PY0=-Y(1)*COS(PHI)
      WRITE(3,200) PY0,PZ0
 200  FORMAT(' PY0=',E9.2,' PZ0=',E9.2)
C
COMPUTE ROTATION ANGLE OF Z AXIS TO HORIZONTAL
C
      COSV=PY0/SQRT(PY0*PY0+PZ0*PZ0)
      SINV=PZ0/SQRT(PY0*PY0+PZ0*PZ0)
      IF(SINV.GE.0.0.AND.COSV.GE.0.0) V=ASIN(SINV)
      IF(SINV.GE.0.0.AND.COSV.LT.0.0) V=ACOS(COSV)
      IF(SINV.LT.0.0.AND.COSV.LT.0.0) V=-ACOS(COSV)+2*PI
      IF(SINV.LT.0.0.AND.COSV.GE.0.0) V=ASIN(SINV)+2*PI
      IF(V.GT.PI) V = -2*PI+V
      DO 1 I=1,N
C
C     FIRST PROJECT THROUGH THE POLAR ANGLE
C
      PZ(I)=Z(I)*SIN(THETA)-Y(I)*COS(THETA)*SIN(PHI)
      PY(I)=-Y(I)*COS(PHI)
C
C     THEN ROTATE IN THE XY PLANE
C
      YP=PY(I)*COS(V)+PZ(I)*SIN(V)
      ZP=PZ(I)*COS(V)-PY(I)*SIN(V)
      PY(I)=YP
      PZ(I)=ZP
      IF(I.EQ.1) GO TO 2
C
COMPUTE SLOPE OF JET IN PROJECTED COORDINATES
C
      DPZDPY(I)=(PZ(I)-XZ)/(PY(I)-XY)
      XZ=PZ(I)
      XY=PY(I)
      GO TO 1
    2 DPZDPY(1)=PZ(1)/PY(1)
      XZ=PZ(1)
      XY=PY(1)
    1 CONTINUE
      RETURN
      END
```

```
C *                                                                    *
C
C **********************************************************************
C *                                                                    *
C *                          P L T S P H                               *
C *                                                                    *
C *                                                                    *
C * PLOTS AND PRINTS PROFILES THROUGH HENRIKSEN/VALLEE SPHEROIDS       *
C *                                                                    *
C **********************************************************************
C
      DIMENSION PROF(101,101)
      LOGICAL DEBUG
      DEBUG=.TRUE.
      WRITE(5,10)
   10 FORMAT(' SPHEROID PRINT/PLOT',/,
     *       ' ==================',//,
     *       ' INPUT ALPHA, A2, A3, RMAX, NS/NIG')
      READ(5,11) ALPHA
   11 FORMAT(F8.3)
      READ(5,11) A2
      READ(5,11) A3
      READ(5,11) RMAX
      READ(5,11) PSOPIG
      STEP=RMAX/100.
      DO 1 I=1,101
      Z=STEP*(I-1)
      DO 2 J=1,101
      XPI=STEP*(J-1)
      SIGMA=(Z/A3)**2 + (XPI/A2)**2
      PROF(I,J) = 1./(1.+(PSOPIG-1.)*(SIGMA**ALPHA))
    2 CONTINUE
    1 CONTINUE
C
C
C WRITE OUT THE PRESSURE PARAMETERS
C
      WRITE(3,12) ALPHA,A2,A3,PSOPIG,RMAX
   12 FORMAT(1H1,'SPHEROID PARAMETERS AS FOLLOWS:',/,
     *       ' ==============================',//,
     *       ' PRESSURE INDEX ALPHA           ',F8.3,/,
     *       ' PRINCIPAL AXIS A2              ',F8.3,/,
     *       ' MINOR AXIS A3                  ',F8.3,/,
     *       ' PRESSURE RATIO PS/PIG          ',F8.3,/,
     *       ' MAXIMUM RADIUS COMPUTED (RMAX) ',F8.3)
C
C WRITE OUT A SHORT MATRIX OF PRESSURES
C
      DO 3 I=1,101,5
      WRITE(3,13) (PROF(I,J), J=1,101,5)
   13 FORMAT(21F6.3)
    3 CONTINUE
      STOP
      END
```

```
00100   C
00200   C ***************************************************************************
00300   C *                                                                         *
00400   C *                              S P H I N T                                *
00500   C *                                                                         *
00600   C * COMPUTES INTEGRAL OF N AND N**2 OVER SPHEROIDAL DENSITY                 *
00700   C * DISTRIBUTIONS TO PERFORM MASS AND X-RAY LUMINOSITY CALCULATIONS         *
00800   C * SPHEROIDAL ATMOSPHERES AS IN HENRIKSEN, VALLEE, BRIDLE REFRACTED-       *
00900   C * JET MODELS (AP.J., IN PRESS)                                           *
01000   C *                                                                         *
01100   C * INTEGRATION IS BY DOUBLE SIMPSON'S ONE-THIRD RULE IN 100 STRIPS         *
01200   C * OVER EACH OF RADIAL AND HEIGHT COORDINATES (CYLINDER ENCLOSING          *
01300   C * SPHEROID)                                                              *
01400   C *                                                                         *
01500   C * RESTRICTIONS: ALPHA MUST BE .GT. ZERO                                   *
01600   C *                                                                         *
01700   C ***************************************************************************
01800   C
01900         DIMENSION PROJ1(101),DEN1(101),PROJ2(101),DEN2(101)
02000         REAL NSONIG,NO,MASS
02100         LOGICAL DEBUG
02200         DEBUG=.FALSE.
02300       999 WRITE(5,10)
02400        10 FORMAT(' SPHEROIDAL INTEGRATION:',/,
02500           *       ' ======================',/,
02600           *       ' INPUT ALPHA, A2 (KPC), A3 (KPC), RMAX (KPC), NS/NIG')
02700         READ(5,11) ALPHA
02800        11 FORMAT(F8.3)
02900         IF(ALPHA.GT.0.) GO TO 998
03000         GO TO 999
03100       998 READ(5,11) A2
03200         READ(5,11) A3
03300         READ(5,11) RMAX
03400         READ(5,11) NSONIG
03500         STEP=RMAX/100.
03600         DO 1 I=1,101
03700         Z=STEP*(I-1)
03800         IF(DEBUG) WRITE(5,11)Z
03900         DO 2 J=1,101
04000         XPI=STEP*(J-1)
04100         SIGMA=(Z/A3)**2 + (XPI/A2)**2
04200         DEN1(J)=1./(1.+(NSONIG-1.)*(SIGMA**ALPHA))
04300         DEN2(J)=(DEN1(J))**2
04400         IF(DEBUG) GO TO 990
04500         GO TO 995
04600       990 IF(J.EQ.1) WRITE(5,15) DEN1(J),DEN2(J)
04700        15 FORMAT(' DENSITY=',F8.4,' DENSSQ=',F8.4)
04800       995 DEN1(J)=XPI*DEN1(J)
04900         DEN2(J)=XPI*DEN2(J)
05000         2 CONTINUE
05100         SUMEV1=0.
05200         SUMOD1=0.
05300         SUMEV2=0.
05400         SUMOD2=0.
05500         DO 4 L=2,100,2
05600         SUMEV1=SUMEV1+DEN1(L)
05700         SUMEV2=SUMEV2+DEN2(L)
05800         4 CONTINUE
05900         DO 5 L=3,99,2
06000         SUMOD1=SUMOD1+DEN1(L)
06100         SUMOD2=SUMOD2+DEN2(L)
06200         5 CONTINUE
06300         PROJ1(I)=STEP/3.*(DEN1(1)+4.*SUMEV1+2.*SUMOD1+DEN1(101))
06400         PROJ2(I)=STEP/3.*(DEN2(1)+4.*SUMEV2+2.*SUMOD2+DEN2(101))
```

```
      IF(DEBUG) WRITE(5,19) I,PROJ1(I),PROJ2(I)
   19 FORMAT(' I=',I5,' PROJ1=',F8.3,' PROJ2=',F8.3)
    1 CONTINUE
      SUMEV1=0.
      SUMOD1=0.
      SUMEV2=0.
      SUMOD2=0.
      DO 6 L=2,100,2
      SUMEV1=SUMEV1+PROJ1(L)
      SUMEV2=SUMEV2+PROJ2(L)
    6 CONTINUE
      DO 7 L=3,99,2
      SUMOD1=SUMOD1+PROJ1(L)
      SUMOD2=SUMOD2+PROJ2(L)
    7 CONTINUE
      FACT1=STEP/3.*(PROJ1(1)+4.*SUMEV1+2.*SUMOD1+PROJ1(101))
      FACT2=STEP/3.*(PROJ2(1)+4.*SUMEV2+2.*SUMOD2+PROJ2(101))
  980 WRITE(5,12)
   12 FORMAT(' INPUT CENTRAL DENSITY (CM**-3), TEMP (K) - E10.2 FMT')
      READ(5,13) NO
      READ(5,13) TO
   13 FORMAT(E10.2)
      MASS=FACT1*4.*3.14159*NO*2.4707E7
      TKEV=TO/1.1605E7
      EXPFCT=EXP(-0.5/TKEV) - EXP(-4.0/TKEV)
      XLUM=FACT2*4.*3.14159*7.034E-3*(NO**2)*SQRT(TO)*EXPFCT
      WRITE(5,20) MASS,XLUM
   20 FORMAT(' TOTAL MASS      =',E9.3,' SOLAR MASSES',/,
     *       ' XRAY LUMINOSITY =',E9.3,'x 1E40 ERG/S FROM 0.5 TO 4KEV')
      WRITE(5,25)
   25 FORMAT(' TYPE 1 TO RE-ENTER NO,TO, 2 FOR NEW SPHEROID, 3 TO STOP')
      READ(5,26) IGO
   26 FORMAT(I1)
      GO TO (980,999,970) IGO
  970 STOP
      END
```