ADAS Subroutine rd2fs

```
       FUNCTION RD2FS(N,L,L2,E2)
C-----------------------------------------------------------------------
C  PURPOSE: GENERATION OF HYDROGENIC BOUND-FREE RADIAL INTEGRALS USING
C           RECURRENCE RELATIONS.
C-----------------------------------------------------------------------
C UNIX-IDL PORT:
C
C AUTHOR:  WILLIAM OSBORN (TESSELLA SUPPORT SERVICES PLC)
C
C DATE:    4TH JULY 1996
C
C VERSION: 1.1                          DATE: 04-07-96
C MODIFIED: WILLIAM OSBORN
C               - FIRST VERSION.
C
C VERSION: 1.2                          DATE: 19-12-01
C MODIFIED: Martin O'Mullane
C               - Removed junk from > column 72.
C
C VERSION: 1.3                          DATE: 16-05-07
C MODIFIED: Allan Whiteford
C               - Modified comments as part of subroutine documentation
C                 procedure.
C
C-----------------------------------------------------------------------
       IMPLICIT REAL*8(A-H,O-Z)
       SC=64.0
       SCL=0.015625
       EN=N
       EN2=EN*EN
       EK2=-E2
       V=1.0+EN2*EK2
       EK=DSQRT(EK2)
       V=V*V
       U=8.0*EN2/V
       P=1.0
       JS=0
       SC2=SC*SC
       SCL2=SCL*SCL
       DO 5 I=1,N
       EI=I
       P=P*U*(1.0+EI*EI*EK2)/(EI*(2.0*EI-1.0))
       AP=DABS(P)
       IF(SCL2.LE.AP)GO TO 5
       JS=JS-1
       P=SC2*P
    5  CONTINUE
       IF(EK.GT.0.04D0)GO TO 6
       P=EN*P
       GO TO 7
    6  P=EN*P/(1.0-DEXP(-6.283185/EK))
    7  IF(EK.GT.1.0D-5)GO TO 8
```

```
      U=-2.0*EN
      GO TO 9
    8 U=-2.0*DATAN(EN*EK)/EK
    9 T2=7.089815*DSQRT(P)*DEXP(U)/V
      V=1.0+EN2*EK2
      IF(L2.EQ.L+1)GO TO 11
      IF(L2.EQ.L-1)GO TO 20
      RD2FS=0.0
      GO TO 50
   11 U=(2.0*EN-1.0)*V
      U=DSQRT(U)
      T3=0.5*U*T2
      NU=N-2
      IF(L-NU)14,13,12
   12 T3=T2
   13 GO TO 40
   14 DO 16 I=L2,NU
      LI=NU-I+L
      EL1=LI+1
      EL2=LI+2
      ES=EL2*EL2
      T1=T2
      T2=T3
      T3=(4.0*(EN2-ES)+EL2*(2.0*EL2-1.0)*V)*T2-2.0*EN*U*T1
      U=(EN2-EL1*EL1)*(1.0+ES*EK2)
      U=DSQRT(U)
      T3=T3/(2.0*EN*U)
      AT3=DABS(T3)
      IF(AT3.LE.SC)GO TO 16
      JS=JS+1
      T3=SCL*T3
      T2=SCL*T2
   16 CONTINUE
      GO TO 40
   20 EN1=N-1
      U=V/(1.0+EN1*EN1*EK2)
      T2=DSQRT(U)*T2/(2.0*EN)
      U=(2.0*EN-1.0)*(1.0+(EN-2.0)*(EN-2.0)*EK2)
      U=DSQRT(U)
      T3=(4.0+EN1*V)*(2.0*EN-1.0)*T2/(2.0*EN*U)
      NU=N-3
      IF(L-NU-1)24,23,22
   22 T3=T2
   23 GO TO 40
   24 DO 26 I=L,NU
      LI=NU-I+L
      EL=LI
      EL1=LI+1
      ES=EL1*EL1
      T1=T2
      T2=T3
      T3=(4.0*(EN2-ES)+EL1*(2.0*EL1+1.0)*V)*T2-2.0*EN*U*T1
      U=(EN2-ES)*(1.0+EL*EL*EK2)
```

```
      U=DSQRT(U)
      T3=T3/(2.0*EN*U)
      AT3=DABS(T3)
      IF(AT3.LE.SC)GO TO 26
      JS=JS+1
      T3=SCL*T3
      T2=SCL*T2
26    CONTINUE
40    RJS=JS
      RD2FS=EN2*EN2*T3*T3*4096.0**RJS
50    RETURN
     END
     INTEGER             L,           L2,          N
     REAL*8              E2
```