ADAS Subroutine e3spln

```
C
      SUBROUTINE E3SPLN( NTDIM  , NDDIM  ,
     &                   ITA    , IDA    , ITVAL   ,
     &                   TETA   , TEDA   , TEVA    , DIN    ,
     &                   PEC    ,          PECA    ,
     &                                     LTRNG   , LDRNG
     &                 )
C
C-----------------------------------------------------------------------
C
C  ****************** FORTRAN77 SUBROUTINE: E3SPLN *********************
C
C  PURPOSE:
C          PERFORMS CUBIC SPLINE ON LOG(TEMPERATURE AND DENSITY)
C          VERSUS LOG(IONIZATIONS PER PHOTON)
C          INPUT DATA FOR A GIVEN WAVELENGTH DATA-BLOCK.
C
C          USING  TWO-WAY SPLINES IT CALCULATES  THE  PHOTON EMISSIVITY
C          FOR  'ITVAL'  PAIRS OF  ELECTRON TEMPERATURES  AND  DENSITIES
C          FROM THE TWO-DIMENSIONAL TABLE OF TEMPERATURES/DENSITIES READ
C          IN FROM THE INPUT FILE. IF A  VALUE  CANNOT  BE  INTERPOLATED
C          USING SPLINES IT IS EXTRAPOLATED VIA 'XXSPLE'.
C
C  CALLING PROGRAM: ADAS503/SPEC
C
C
C  SUBROUTINE:
C
C  INPUT : (I*4)  NTDIM   = MAX NUMBER OF ELECTRON TEMPERATURES ALLOWED
C  INPUT : (I*4)  NDDIM   = MAX NUMBER OF ELECTRON DENSITIES    ALLOWED
C
C  INPUT : (I*4)  ITA     = INPUT DATA FILE: NUMBER OF ELECTRON TEMPERA-
C                           TURES READ FOR THE DATA-BLOCK BEING ASSESSED
C  INPUT : (I*4)  IDA     = INPUT DATA FILE: NUMBER OF ELECTRON DENSIT-
C                           IES   READ FOR THE DATA-BLOCK BEING ASSESSED
C  INPUT : (I*4)  ITVAL   = NUMBER OF ISPF ENTERED TEMPERATURE/DENSITY
C                           PAIRS  FOR  WHICH  IOINIZATIONS PER PHOTON
C                           ARE REQUIRED FOR TABULAR/GRAPHICAL OUTPUT.
C
C  INPUT : (R*8)  TETA()  = INPUT DATA FILE: ELECTRON TEMPERATURES (EV)
C                           FOR THE DATA-BLOCK BEING ASSESSED.
C                           DIMENSION: ELECTRON TEMPERATURE INDEX
C  INPUT : (R*8)  TEDA()  = INPUT DATA FILE: ELECTRON DENSITIES (CM-3)
C                           FOR THE DATA-BLOCK BEING ASSESSED.
C                           DIMENSION: ELECTRON DENSITY INDEX
C  INPUT : (R*8)  TEVA()  = USER ENTERED: ELECTRON TEMPERATURES (EV)
C                           DIMENSION: TEMPERATURE/DENSITY PAIR INDEX
C  INPUT : (R*8)  DIN()   = USER ENTERED: ELECTRON DENSITIES (CM-3)
C                           DIMENSION: TEMPERATURE/DENSITY PAIR INDEX
C
C
C  INPUT : (R*8)  PEC(,)   =INPUT DATA FILE: FULL SET OF IONIZATIONS PER
```

```
C                              PHOTON VALUES FOR THE DATA-BLOCK BEING
C                              ANALYSED.
C                              1ST DIMENSION: ELECTRON TEMPERATURE INDEX
C                              2ND DIMENSION: ELECTRON DENSITY     INDEX
C  OUTPUT: (R*8)  PECA()   = SPLINE INTERPOLATED OR  EXTRAPOLATED  IONIZ-
C                              ATIONS/PHOTON FOR THE USER ENTERED ELECTRON
C                              TEMPERATURE/DENSITY PAIRS.
C                              DIMENSION: TEMPERATURE/DENSITY PAIR INDEX
C
C  OUTPUT: (L*4)  LTRNG()=  .TRUE.  => OUTPUT 'PECA()' VALUE WAS INTER-
C                                      POLATED  FOR  THE  USER  ENTERED
C                                      ELECTRON TEMPERATURE 'TEVA()'.
C                           .FALSE. => OUTPUT 'PECA()' VALUE WAS EXTRA-
C                                      POLATED  FOR  THE  USER  ENTERED
C                                      ELECTRON TEMPERATURE 'TEVA()'.
C                           DIMENSION: TEMPERATURE/DENSITY PAIR INDEX
C
C  OUTPUT: (L*4)  LDRNG()=  .TRUE.  => OUTPUT 'PECA()' VALUE WAS INTER-
C                                      POLATED  FOR  THE  USER  ENTERED
C                                      ELECTRON DENSITY 'DIN()'.
C                           .FALSE. => OUTPUT 'PECA()' VALUE WAS EXTRA-
C                                      POLATED  FOR  THE  USER  ENTERED
C                                      ELECTRON DENSITY 'DIN()'.
C                           DIMENSION: TEMPERATURE/DENSITY PAIR INDEX
C
C          (I*4)  NIN      = PARAMETER = MAX. NO. OF INPUT  TEMP/DENSITY
C                                        VALUES. MUST BE >= 'ITA'&'IDA'
C          (I*4)  NOUT     = PARAMETER = MAX. NO. OF OUTPUT TEMP/DENSITY
C                                        PAIRS.  MUST BE >= 'ITVAL'
C          (I*4)  L1       = PARAMETER = 1
C
C          (I*4)  IED      = ARRAY SUBSCRIPT USED INPUT  FILE  ELECTRON
C                            DENSITIES.
C          (I*4)  IET      = ARRAY SUBSCRIPT USED INPUT  FILE  ELECTRON
C                            TEMPERATURES.
C          (I*4)  IT       = ARRAY  SUBSCRIPT  USED  FOR  USER  ENTERED
C                            TEMPERATURE/DENSITY PAIRS .
C          (I*4)  IOPT     = DEFINES THE BOUNDARY DERIVATIVES FOR THE
C                            SPLINE ROUTINE 'XXSPLE', SEE 'XXSPLE'.
C                            (VALID VALUES = <0, 0, 1, 2, 3, 4)
C
C          (L*4)  LSETX    = .TRUE.  => SET UP SPLINE PARAMETERS RELATING
C                                       TO 'XIN' AXIS.
C                            .FALSE. => DO NOT SET UP SPLINE PARAMETERS
C                                       RELATING TO 'XIN' AXIS.
C                                       (I.E. THEY WERE SET IN A PREVIOUS
C                                             CALL )
C                            (VALUE SET TO .FALSE. BY 'XXSPLE')
C
C          (R*8)  R8FUN1   = FUNCTION - (SEE ROUTINES SECTION BELOW)
C
C          (R*8)  XIN()    = 1) LOG( DATA FILE ELECTRON DENSITIES    )
C                            2) LOG( DATA FILE ELECTRON TEMPERATURES )
```

```
C          (R*8)  YIN()    = LOG( DATA FILE IONIZATIONS/PHOTON )
C          (R*8)  XOUT()   = 1) LOG( SCALED USER ENTERED ELECTRON DENS. )
C                            2) LOG( SCALED USER ENTERED ELECTRON TEMPS.)
C          (R*8)  YOUT()   = LOG( OUTPUT GENERATED IONIZATIONS/PHOTON )
C          (R*8)  YPASS(,)= LOG( IONIZATIONS/PHOTON) INTERMEDIATE ARRAY
C                            WHICH   STORES   INTERPOLATED/EXTRAPOLATED
C                            VALUES  BETWEEN  THE  TWO  SPLINE SECTIONS.
C                            SECTIONS.
C          (R*8)  DF()     = SPLINE INTERPOLATED DERIVATIVES
C
C
C NOTE:
C
C ROUTINES:
C          ROUTINE    SOURCE    BRIEF DESCRIPTION
C          ------------------------------------------------------------
C          XXSPLE     ADAS      SPLINE SUBROUTINE (EXTENDED DIAGNOSTICS)
C          R8FUN1     ADAS      REAL*8 FUNCTION: ( X -> X )
C
C AUTHOR:  H. P. SUMMERS
C          K1/1/57
C          JET EXT. 4941
C
C DATE:    11/10/91
C UNIX-IDL PORT:
C
C VERSION: 1.1                          DATE: 28-02-95
C MODIFIED: LALIT JALOTA (TESSELLA SUPPORT SERVICES PLC)
C               - FIRST VERSION
C
C VERSION: 1.2                          DATE: 15-04-96
C MODIFIED: TIM HAMMOND (TESSELLA SUPPORT SERVICES PLC)
C               - INCREASED PARAMETER NIN 24 -> 26
C                 (CURRENT JET VALUE 35, BUT KEPT TO 26 FOR PRESENT TO
C                  PREVENT CHANGES TO IDL ROUTINES).
C
C VERSION: 1.3                          DATE: 22-04-96
C MODIFIED: TIM HAMMOND
C - INCREASED NIN TO 35 IN LINE WITH JET.
C
C VERSION: 1.4                          DATE: 13-10-97
C MODIFIED: Martin O'Mullane
C               - INCREASED PARAMETER NOUT 20 -> 35.
C
C VERSION: 1.5                          DATE: 20-09-99
C MODIFIED: RICHARD MARTIN
C               - INCREASED NOUT 35 -> 100 ; NIN 35 -> 100
C
C-----------------------------------------------------------------------
C
C-----------------------------------------------------------------------
      INTEGER            IDA,         ITA,         ITVAL,       NDDIM
      INTEGER            NTDIM
```

```
LOGICAL            LDRNG(ITVAL),             LTRNG(ITVAL)
REAL*8             DIN(ITVAL),  PEC(NTDIM,NDDIM)
REAL*8             PECA(ITVAL), TEDA(IDA),   TETA(ITA)
REAL*8             TEVA(ITVAL)
```