ADAS Subroutine xxdata_02

```
       SUBROUTINE xxdata_02( IUNIT  , DSNAME ,
     &                       NSTORE , NEDIM  ,
     &                       NBSEL  , ISELA  ,
     &                       CPRIMY , CSECDY , CTYPE,
     &                       AMPA   , AMSA   , ALPHA  , ETHRA  ,
     &                       IEA    ,
     &                       TEEA   , SIA
     &                     )
C-----------------------------------------------------------------------
C
C  ***************** FORTRAN77 SUBROUTINE: xxdata_02 ******************
C
C  PURPOSE:  TO  FETCH  DATA  FROM  INPUT ION/ATOM CROSS-SECTION
C            FILES OF TYPE ADF02.
C
C  CALLING PROGRAM: ADAS302/SSIA
C
C  DATA:
C
C           UP TO 'NSTORE' SETS (DATA-BLOCKS) OF DATA MAY BE  READ FROM
C           THE FILE - EACH BLOCK FORMING A COMPLETE SET OF  CROSS-
C           SECTION VALUES FOR GIVEN COLLISION ENERGIES.
C           EACH DATA-BLOCK  IS  ANALYSED INDEPENDENTLY OF ANY  OTHER
C           DATA-BLOCK.
C
C           THE UNITS USED IN THE DATA FILE ARE TAKEN AS FOLLOWS:
C
C           COLLISION ENERGIES  : EV/AMU
C           CROSS-SECTION       : CM**2
C
C  SUBROUTINE:
C
C  INPUT : (I*4)  IUNIT    = UNIT TO WHICH INPUT FILE IS ALLOCATED.
C  INPUT : (C*80) DSNAME   = MVS DATA SET NAME OF DATA SET BEING READ
C
C  INPUT : (I*4)  NSTORE   = MAXIMUM NUMBER  OF  INPUT DATA-BLOCKS  THAT
C                            CAN BE STORED.
C  INPUT : (I*4)  NEDIM    = MAX NUMBER OF COLLISION ENERGIES ALLOWED
C
C  OUTPUT: (I*4)  NBSEL    = NUMBER OF DATA-BLOCKS ACCEPTED & READ IN.
C  OUTPUT: (I*4)  ISELA()  = READ - DATA-SET DATA-BLOCK ENTRY INDICES
C                            DIMENSION: DATA-BLOCK INDEX
C
C  OUTPUT: (C*5)  CPRIMY() = READ - PRIMARY SPECIES IDENTIFICATION
C                            DIMENSION: DATA-BLOCK INDEX
C  OUTPUT: (C*5)  CSECDY() = READ - SECONDARY SPECIES IDENTIFICATION
C                            DIMENSION: DATA-BLOCK INDEX
C  OUTPUT: (C*3)  CTYPE()  = READ - CROSS-SECTION TYPE
C                            DIMENSION: DATA-BLOCK INDEX
C
C  OUTPUT: (R*8)  AMPA()   = READ - PRIMARY SPECIES ATOMIC MASS NUMBER
C                            DIMENSION: DATA-BLOCK INDEX
```

```
C  OUTPUT: (R*8)  AMSA()   = READ - SECONDARY SPECIES ATOMIC MASS NUMBER
C                           DIMENSION: DATA-BLOCK INDEX
C  OUTPUT: (R*8)  ALPHA()  = READ - HIGH ENERGY EXTRAPOLATION PARM.
C                           DIMENSION: DATA-BLOCK INDEX
C  OUTPUT: (R*8)  ETHRA()  = READ - ENERGY THRESHOLD (EV)
C                           DIMENSION: DATA-BLOCK INDEX
C
C  OUTPUT: (I*4)  IEA()    = READ - NUMBER OF COLLISION ENERGIES
C                           DIMENSION: DATA-BLOCK INDEX
C
C  OUTPUT: (R*8)  TEEA(,)  = READ - COLLISION ENERGIES (UNITS: eV/AMU)
C                           1st DIMENSION: COLLISION ENERGY INDEX
C                           2nd DIMENSION: DATA-BLOCK INDEX
C
C  OUTPUT: (R*8)  SIA(,)   =READ - FULL SET OF COLLISION CROSS-
C                                  SECTION VALUES (cm**2)
C                           1st DIMENSION: COLLISION ENERGY INDEX
C                           2nd DIMENSION: DATA-BLOCK INDEX
C
C          (I*4)  I4EIZ0   = FUNCTION - (SEE ROUTINES SECTION BELOW)
C          (I*4)  I4FCTN   = FUNCTION - (SEE ROUTINES SECTION BELOW)
C          (I*4)  I4UNIT   = FUNCTION - (SEE ROUTINE SECTION BELOW)
C          (I*4)  IBLK     = ARRAY INDEX: DATA-BLOCK INDEX
C          (I*4)  ITT      = ARRAY INDEX: COLLISION ENERGY INDEX
C          (I*4)  NENUM    = NUMBER OF COLLISION ENERGIES FOR CURRENT
C                           DATA-BLOCK
C          (I*4)  IABT     = RETURN CODE FROM 'I4FCTN'
C          (I*4)  IPOS2    = GENERAL USE STRING INDEX VARIABLE
C
C          (R*8)  R8FCTN   = FUNCTION - (SEE ROUTINES SECTION BELOW)
C
C          (L*4)  LBEND    = IDENTIFIES WHETHER THE LAST OF THE  INPUT
C                           DATA SUB-BLOCKS HAS BEEN LOCATED.
C                           (.TRUE. => END OF SUB-BLOCKS REACHED)
C
C          (C*1)  CSLASH   = '/' - DELIMITER FOR 'XXHKEY'
C          (C*2)  C2       = GENERAL USE TWO BYTE CHARACTER STRING
C          (C*1)  CKEY1    = 'P'      - INPUT BLOCK HEADER KEY
C          (C*1)  CKEY2    = 'S'      - INPUT BLOCK HEADER KEY
C          (C*1)  CKEY3    = 'A'      - INPUT BLOCK HEADER KEY
C          (C*1)  CKEY4    = 'E'      - INPUT BLOCK HEADER KEY
C          (C*4)  CKEY5    = 'T'      - INPUT BLOCK HEADER KEY
C          (C*3)  C3       = GENERAL USE THREE BYTE CHARACTER STRING
C          (C*9)  C10      = GENERAL USE NINE BYTE CHARACTER STRING
C          (C*80) C80      = GENERAL USE 80 BYTE  CHARACTER  STRING  FOR
C                           THE INPUT OF DATA-SET RECORDS.
C
C ROUTINES:
C          ROUTINE    SOURCE    BRIEF DESCRIPTION
C          ------------------------------------------------------------
C          XXHKEY     ADAS      OBTAIN KEY/RESPONSE STRINGS FROM TEXT
C          I4FCTN     ADAS      INTEGER*4 FUNCTION    -
C                               CONVERT CHARACTER STRING TO INTEGER
```

```
C           I4UNIT     ADAS      INTEGER*4 FUNCTION    -
C                                FETCH UNIT NUMBER FOR OUTPUT OF MESSAGES
C           R8FCTN     ADAS      REAL*8 FUNCTION       -
C                                CONVERT CHARACTER STRING TO REAL*8
C
C AUTHOR:  H. P. SUMMERS, UNIVERSITY OF STRATHCLYDE
C          JA8.08
C          TEL. 0141-553-4196
C
C DATE:    12/11/96
C
C UNIX-IDL PORT: H.P.SUMMERS
C
C VERSION: 1.1                          DATE: 19-11-96
C MODIFIED: WILLIAM OSBORN (TESSELLA SUPPORT SERVICES PLC)
C           - PUT UNDER S.C.C.S. CONTROL
C
C VERSION: 1.2                          DATE: 14-02-97
C MODIFIED: RICHARD MARTIN
C               - CHANGED INITIALISATION 'CKEY2 /'S  '/' TO 'CKEY2 /'S'/ '
C
C
C-----------------------------------------------------------------------
C
C NOTES: Copied from c2data.for.
C        This is v1.1 of xxdata_02.
C
C
C VERSION  : 1.1
C DATE     : 01-12-2005
C MODIFIED : Martin O'Mullane
C              - First version.
C
C-----------------------------------------------------------------------
      CHARACTER*5         CPRIMY(NSTORE),           CSECDY(NSTORE)
      CHARACTER*3         CTYPE(NSTORE)
      CHARACTER*80        DSNAME
      INTEGER             IEA(NSTORE), ISELA(NSTORE),            IUNIT
      INTEGER             NBSEL,       NEDIM,       NSTORE
      REAL*8              ALPHA(NSTORE),            AMPA(NSTORE)
      REAL*8              AMSA(NSTORE),             ETHRA(NSTORE)
      REAL*8              SIA(NEDIM,NSTORE),        TEEA(NEDIM,NSTORE)
```