ADAS Subroutine cefill

```
       SUBROUTINE CEFILL( MXNENG , MXNSHL ,
     &                    SYMBR  , SYMBD  , IZR    , IZD    ,
     &                    INDD   , NENRGY , NMIN   , NMAX   ,
     &                    LPARMS , LSETL  , ENRGYA ,
     &                    ALPHAA , LFORMA , XLCUTA , PL2A   ,
     &                    PL3A   , SIGTA  , SIGNA  , SIGLA
     &                  )
C-----------------------------------------------------------------------
C  ****************** FORTRAN77 SUBROUTINE: CDFILL *********************
C
C  PURPOSE:  FILL HIGH N ZEROES IN AN ADF01 IF PRESENT.
C
C  CALLING PROGRAM: ADAS314
C  DATA:
C           THE UNITS USED IN THE DATA FILE ARE TAKEN AS FOLLOWS:
C           COLLISION ENERGIES  : KEV/AMU
C           ALPHA               :
C           TOTAL XSECTS.       : CM2
C           N-SHELL XSECTS.     : CM2
C           NL-SHELL DATA       : CM2
C           NLM-SHELL DATA      : CM2
C  SUBROUTINE:
C  INPUT : (I*4)  MXNENG    = MAXIMUM NO. OF ENERGIES.
C  INPUT : (I*4)  MXNSHL    = MAXIMUM NO. OF N SHELLS.
C  INPUT : (C*2)  SYMBR     = READ - RECEIVER ION ELEMENT SYMBOL.
C  INPUT : (C*2)  SYMBD     = READ - DONOR ION ELMENT SYMBOL.
C  INPUT : (I*4)  IZR       = READ - ION CHARGE OF RECEIVER.
C  INPUT : (I*4)  IZD       = READ - ION CHARGE OF DONOR.
C  INPUT : (I*4)  INDD      = READ - DONOR STATE INDEX.
C  INPUT : (I*4)  NENRGY    = NUMBER OF ENERGIES READ.
C  INPUT : (I*4)  NMIN      = LOWEST N-SHELL FOR WHICH DATA READ.
C  INPUT : (I*4)  NMAX      = HIGHEST N-SHELL FOR WHICH DATA READ.
C  INPUT : (L*4)  LPARMS    = FLAGS IF L-SPLITTING PARAMETERS PRESENT.
C                             .TRUE.  => L-SPLITTING PARAMETERS PRESENT.
C                             .FALSE  => L-SPLITTING PARAMETERS ABSENT.
C  INPUT : (L*4)  LSETL     = FLAGS IF L-RESOLVED DATA PRESENT.
C                             .TRUE.  => L-RESOLVED DATA PRESENT.
C                             .FALSE  => L-RESOLVED DATA ABSENT.
C  INPUT : (R*8)  ENRGYA()  = READ - COLLISION ENERGIES.
C                             UNITS: EV/AMU (READ AS KEV/AMU)
C                             DIMENSION: ENERGY INDEX
C  INPUT : (R*8)  ALPHAA()  = READ - EXTRAPOLATION PARAMETER ALPHA.
C                             DIMENSION: ENERGY INDEX
C  INPUT : (I*4)  LFORMA()  = READ - PARAMETERS FOR CALCULATING L-RES
C                                    X-SEC.
C                             DIMENSION: ENERGY INDEX
C  INPUT : (R*8)  XLCUTA()  = READ - PARAMETERS FOR CALCULATING L-RES
C                                    X-SEC.
C                             DIMENSION: ENERGY INDEX
C  INPUT : (R*8)  PL2A()    = READ - PARAMETERS FOR CALCULATING L-RES
C                                    X-SEC.
C                             DIMENSION: ENERGY INDEX
```

```
C  INPUT : (R*8)  PL3A()    = READ - PARAMETERS FOR CALCULATING L-RES
C                                   X-SEC.
C                             DIMENSION: ENERGY INDEX
C  I/O   : (R*8)  SIGTA()   = READ - TOTAL CHARGE EXCHANGE
C                                   CROSS-SECTION.
C                             UNITS: CM2
C                             DIMENSION: ENERGY INDEX
C  I/O   : (R*8)  SIGNA(,)  = READ - N-RESOLVED CHARGE EXCHANGE
C                                   CROSS-SECTIONS.
C                             UNITS: CM2
C                             1ST DIMENSION: ENERGY INDEX
C                             2ND DIMENSION: N-SHELL
C  I/O   : (R*8)  SIGLA(,)  = READ - L-RESOLVED CHARGE EXCHANGE
C                                   CROSS-SECTIONS.
C                             UNITS: CM2
C                             1ST DIMENSION: ENERGY INDEX
C                             2ND DIMENSION: INDEXED BY I4IDFL(N,L)
C  I/O   : (R*8)  SIGMA(,)  = READ - M-RESOLVED CHARGE EXCHANGE
C                                   CROSS-SECTIONS.
C                             UNITS: CM2
C                             1ST DIMENSION: ENERGY INDEX
C                             2ND DIMENSION: INDEXED BY I4IDFM(N,L,M)
C                                            WITH M >= 0 ONLY
C          (I*4)  I        = N QUANTUM NUMBER.
C          (I*4)  J        = L QUANTUM NUMBER.
C          (I*4)  K        = M QUANTUM NUMBER.
C          (I*4)  N        = N QUANTUM NUMBER.
C          (I*4)  L1       = L QUANTUM NUMBER + 1
C          (I*4)  M1       = M QUANTUUM NUMBER + 1
C ROUTINES:
C          ROUTINE    SOURCE    BRIEF DESCRIPTION
C          ------------------------------------------------------------
C          I4IDFL     ADAS      RETURNS UNIQUE INDEX FROM QUANTUM
C                               NUMBERS N AND L.
C          I4IDFM     ADAS      RETURNS UNIQUE INDEX FROM QUANTUM
C                               NUMBERS N, L AND M.
C AUTHOR:  H. P. SUMMERS, UNIVERSITY OF STRATHCLYDE
C          JA8.08
C          TEL 0141-553-4196
C DATE:    21/09/95
C UPDATE:  27/08/97  HP SUMMERS - CHANGED NAME FROM CCFILL TO CDFILL
C
C VERSION: 1.1 DATE: 01-12-98
C MODIFIED: RICHARD MARTIN
C - PUT UNDER SCCS CONTROL
C
C VERSION:      1.2                                    DATE: 17-05-07
C MODIFIED: Allan Whiteford
C               - Updated comments as part of subroutine documentation
C                 procedure.
C
C VERSION  : 1.3
C DATE     : 22-05-2007
```

```
C MODIFIED : Martin O'Mullane
C              - Remove unused m-subshell data possibility.
C
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
      CHARACTER*2         SYMBD,       SYMBR
      INTEGER             INDD,        IZD,         IZR
      INTEGER             LFORMA(MXNENG),           MXNENG,      MXNSHL
      INTEGER             NENRGY,      NMAX,        NMIN
      LOGICAL             LPARMS,      LSETL
      REAL*8              ALPHAA(MXNENG),           ENRGYA(MXNENG)
      REAL*8              PL2A(MXNENG),             PL3A(MXNENG)
      REAL*8              SIGLA(MXNENG,(MXNSHL*(MXNSHL+1))/2)
      REAL*8              SIGNA(MXNENG,MXNSHL),     SIGTA(MXNENG)
      REAL*8              XLCUTA(MXNENG)
```