ADAS Subroutine xxdtes

```
      subroutine xxdtes( cstrg , leiss  , lstan  ,  nvlce )
c
c-----------------------------------------------------------------------
c
c  ****************** fortran77 subroutine: xxdtes *********************
c
c  purpose: Detects if the configuration string from a specific ion
c           level list line is of eissner form , standard form or
c           neither.
C
C           If neither, the subroutine checks to see if it is a
c           bundle (* in the string) or based on a parent ([..] in
c           the string).  If the string is of Eissner or standard
c           form, the n-shell and l-shell of the outermost
c           (valence)  electron is returned.
c
c           A version of this routine with a more extended return of
c           parameters and bale to handle very long configuration
c           strings is available as 'g5dtes.for'.
c
c  calling programs: general use
c
c  subroutine:
c
c  input : (c*(*)) cstrg    = configuration character string
c  output: (l*4)   leiss    = .true.  => eissner form
c                             .false. => not eissner form
c  output: (l*4)   lstan    = .true.  => standard form
c                             .false. => not standard form
c  output: (i*4)   nvlce    = outer electron n-shell if recognisable
c
c          (l*4)   lbndl    = .true.  => bundled form ('*' found)
c                             .false. => not bundled form
c          (l*4)   lprnt    = .true.  => parent form ('[...]' found)
c                             .false. => not parent form
c          (c*19)  cstr_top = leading part of config. string in Eissner
c                             format (no leading blank, trailing blanks)
c          (c*(*)) cstr_tail= trailing part of config. string in Eissner
c                             format (no leading blank, trailing blanks)
c          (i*4)   lvlce    = outer electron l-shell if recognisable
c
c          (i*4)   i       = general use
c          (i*4)   iabt    = return code (see specific function)
c                            0 => ok
c                            1 => fault detected
c          (i*4)   icfsel  = 1 => standard form out, standard form in
c                            2 => eissner  form out, standard form in
c                            3 => standard form out, eissner  form in
c                            4 => eissner  form out, eissner  form in
c          (i*4)   ishel   = shell counter
c          (i*4)   ip      = parity of configuration
c          (i*4)   maxn    = n_shell sum for configuration
```

```
c           (i*4)   nshel   = number of shells identified ffrom string
c           (i*4)   ndword  = maximum number of words in string
c           (i*4)   nfirst  = first word to be extracted from string
c           (i*4)   nwords  = number of words in string
c           (i*4)   nela()  = number of electrons in each shell
c           (i*4)   ifirst()= position of first char. of word in string
c           (i*4)   ilast() = position of last char. of word in string
c
c           (c*1)   cdelim  = separators for words in string
c           (c*19)  cstrgo  = general use string
c           (c*19)  strg    = standard form configuration string
c           (c*19)  strge   = eissner form configuration string
c           (c*1)   cheisa()= eissner character for orbitals
c           (c*2)   chstda()= standard orbital spec. for each shell
c           (c*2)   cnela() = chars. for no. of equiv. elec. in shell
c                             (eissner form case)
c           (c*1)   chqa()  = index to hexadecimal conversions
c           (c*1)   chra()  = char. for no. of. equiv. elec. in shell
c                             (standard form case)
c
c routines:
c          routine    source   brief description
c          -------------------------------------------------------------
c          i4fctn     adas     converts character string to integer
c          i4ngrp     adas     returns n quantum number in the
c                              eissner single hexadecimal character form
c          i4pgrp     adas     returns parity of orbital given the
c                              eissner single hexadecimal character form
c          i4schr     adas     returns numerical value for number of
c                              equivalent electrons given as hex> char.
c          cstgrp     adas     returns term of orbital given in the
c                              eissner single hexadecimal character form
c          ceigrp     adas     returns eissner code for orbital
c          xxword     adas     finds number of words in a string
c          xxcmps     adas     compare standard config. strings
c
c
c
c author:  h. p. summers, university of strathclyde
c          ja8.08
c          tel. 0141-553-4196
c
c date:    19/02/03
c
C VERSION: 1.1                          DATE: 19-1-96
C MODIFIED: TIM HAMMOND (TESSELLA SUPPORT SERVICES PLC)
C               - PUT UNDER S.C.C.S. CONTROL
C
C VERSION: 1.2                          DATE: 14-10-96
C MODIFIED: WILLIAM OSBORN (TESSELLA SUPPORT SERVICES PLC)
C               - ADDED CHANGES DATED 01/10/96 ABOVE
C
C VERSION: 1.3                          DATE: 28-08-97
```

```
C MODIFIED: HUGH SUMMERS
C               - ADDED CHANGES TO CHECK 'G' STATES
C
C VERSION: 1.4                         DATE: 19/02/03
C MODIFIED: HUGH SUMMERS
C               - Rewrite based on g5dtes.for
C
C VERSION: 1.5                         DATE: 28/09/2004
C MODIFIED: Martin O'Mullane
C               - Incorrect redirection when checking the Eissner pattern.
C                 The if statement block checking ir jumped out of the
C                 current sub-block to the end of the previous sub-block
C                 rather than to the end of its own sub-block.
C
C VERSION: 1.6                         DATE: 17/05/2007
C MODIFIED: Allan Whiteford
C           - Updated comments as part of subroutine documentation
C             procedure.
C
c-----------------------------------------------------------------------
c-----------------------------------------------------------------------
      CHARACTER*(*)       CSTRG
      INTEGER             NVLCE
      LOGICAL             LEISS,       LSTAN
```