

NAME

RNGLST – Expand a range specification and append it to a list of integers.

SYNOPSIS

CALL RNGLST(TOKEN,LT, LIST,LL, NL,RC)

TOKEN(LT)	is a CHARACTER*1 string containing the range specification
LT	is the INTEGER*4 dimensioned size of TOKEN
LIST(LL)	is the INTEGER*4 list of integers input and then returned
LL	is the INTEGER*4 dimensioned size of LIST
NL	is the INTEGER*4 number of integers added to LIST on this call
RC	is the INTEGER*4 return code; see below

DESCRIPTION

If LT or LL is less than 1, the routine returns with NL=0 and LIST unchanged. Otherwise it examines the characters of TOKEN from left to right in search of a hyphen.

If a hyphen is found it uses DTB to convert the integer preceding the hyphen into a starting value and the integer following the hyphen into an ending value. Then it determines whether the range specifies increasing or decreasing values, and appends to LIST the values from starting to ending in the appropriate direction.

If no hyphen is found the routine uses DTB to convert the single integer into a single value and appends it to LIST.

SEE ALSO

ARGLST, which uses this routine.

DIAGNOSTICS

On output these are the possible RC values:

- 0 all went well
- 1 parse error
- 2 more than LL entries for LIST
- 3 the input parameters did not make sense

LINKAGE

gfortran source.f -L\${HOME}/lib -lmisc

AUTHOR

Michael Kupferschmid

EXAMPLE

```

      CHARACTER*12 TOKEN
      INTEGER*4 LIST(20)/20*0/,RC
      N=0
2     CALL PROMPT('token:',6)
      READ(5,901,END=1) TOKEN
901   FORMAT(A12)
      LT=LENGTH(TOKEN,12)
      CALL RNGLST(TOKEN,LT, LIST(N+1),20, NL,RC)
      N=N+NL
      WRITE(6,902) RC,NL,(LIST(I),I=1,N)
902   FORMAT('RC=',I1,' NL=',I2,' LIST=',20I3)
      GO TO 2
1     WRITE(6,*)
      STOP
      END

```

When compiled and run this program produces the following output.

```

unix[1] a.out
token: 123
RC=0 NL= 1 LIST=123
token: 8-2
RC=0 NL= 7 LIST=123  8  7  6  5  4  3  2
token: -1
RC=1 NL= 0 LIST=123  8  7  6  5  4  3  2
token: ^D
unix[2]

```