NAME

TKNIZE − List the blank-separated tokens in a string.

SYNOPSIS

CALL TKNIZE(STRING,LS,MXLT,MXNT, TKNS,NT,RC)

STRING(LS)	is the CHARACTER*1 string to tokenize
LS	is the INTEGER*4 dimensioned length of STRING
MXLT	is the INTEGER*4 maximum length allowed for a token
MXNT	is the INTEGER*4 maximum number of tokens allowed
TKNS(MXLT,MXNT)	is the CHARACTER*1 list of tokens extracted from STRING
NT	is the INTEGER*4 number of tokens found
RC	is the INTEGER*4 return code; see below

DESCRIPTION

First the routine initializes TKNS to all blanks. Next it uses STRCPY to copy STRING into a workspace, and SPACEL to remove leading and extra blanks from the workspace. Then it copies the blank-separated tokens of the workspace into the columns of TKNS. If STRING contains only blanks, this routine returns NT=0 and RC=1 with TKNS blank; if it contains no blanks it returns NT=1 and RC=0 with the entire input string in the first column of TKNS and the rest of TKNS blank.

DIAGNOSTICS

On output these are the possible RC values:

0 all went well
1 STRING is empty
2 STRING contains more than MXNT tokens
3 some token is longer than MXLT letters

LINKAGE

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

AUTHOR

Michael Kupferschmid

EXAMPLE

```
      CHARACTER*20 STRING/'here are some tokens'/
      CHARACTER*1 TKNS(6,4)
      INTEGER*4 RC
      CALL TKNIZE(STRING,20,6,4, TKNS,NT,RC)
      WRITE(6,901) RC,NT
  901 FORMAT('RC=',I1,' NT=',I2)
      DO 1 I=1,NT
           L=LENGTH(TKNS(1,I),6)
           WRITE(6,902) (TKNS(K,I),K=1,L)
  902      FORMAT(6A1)
    1 CONTINUE
      STOP
      END
```

This example produced the following output:

```
unix[1] a.out
RC=0 NT= 4
here
are
some
tokens
unix[2]
```