## NAME

BTD − Convert an INTEGER*4 number to a string of numerals.

## SYNOPSIS

**CALL BTD(N,STRING,LS,RC)**

| | |
|---|---|
| N | is the INTEGER*4 value to be converted |
| STRING(LS) | is the CHARACTER*1 string of numerals returned |
| LS | is the INTEGER*4 dimension of STRING |
| RC | is the INTEGER*4 return code |

## DESCRIPTION

LS must be supplied on input, and it is not changed by the routine. First the routine finds the digits of the absolute value of N and inserts them right-justified in STRING, padded with leading zeros. Then it changes the leading zeros to blanks. Finally, if the number is negative, it inserts a minus sign in the character position immediately to the left of the first digit. For N = 0 the routine returns a single "0" right-justified in STRING.

## SEE ALSO

DTB, which converts a string of numerals to an INTEGER*4 number.

## DIAGNOSTICS

RC=0 if all went well

RC=1 if LS ≤ 0, or if the number has more than LS digits, or if N < 0 and there are not enough character positions to insert the minus sign

## LINKAGE

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

## AUTHOR

Michael Kupferschmid

**EXAMPLE**

```
      CHARACTER*1 STRING(11)
      CHARACTER*11 OVLY
      EQUIVALENCE(STRING,OVLY)
      CHARACTER*11 EXX/'xxxxxxxxxxx'/
      INTEGER*4 RC
      OVLY=EXX
      CALL BTD(1,STRING,1,RC)
      WRITE(6,900) STRING,RC
      OVLY=EXX
      CALL BTD(-1,STRING,2,RC)
      WRITE(6,900) STRING,RC
      OVLY=EXX
      CALL BTD(0,STRING,5,RC)
      WRITE(6,900) STRING,RC
      OVLY=EXX
      CALL BTD(1234567,STRING,8,RC)
      WRITE(6,900) STRING,RC
      OVLY=EXX
      CALL BTD(1234567,STRING,6,RC)
      WRITE(6,900) STRING,RC
  900 FORMAT(11A1,1X,I1)
      STOP
      END
```

This example produced the following output:

```
unix[1] a.out
1xxxxxxxxxx 0
-1xxxxxxxxx 0
    0xxxxxx 0
 1234567xxx 0
234567xxxxx 1
unix[2]
```