Author(s) : M. Goossens, A. Petrilli, M. Marquina **Library:** KERNLIB
Submitter : **Submitted:** 11.02.1986
Language : Fortran **Revised:** 28.09.94

Utility Package for Character Manipulation

M433 is a comprehensive package for the manipulation of type CHARACTER strings.

Structure:

SUBROUTINE and FUNCTION subprograms
User Entry Names: INDEXA, INDEXB, INDEXC, INDEXN, INDEXS, INDXAC, INDXBC, INDXNC, ISCAN, REPEAT, SPACES, STRIP, SUBWORD, VERIFY, WORD, WORDS, WORDSEP

Usage:

In what follows, the parameters STR, SSTR, SET, the functions REPEAT, SPACES and the variables CHD, CHOPT and H are of type CHARACTER. The function VERIFY is of type INTEGER.

```
I = INDEXA(STR)
```

sets I equal to the position of the first alphabetic character (upper or lower case) in STR. I = 0 if no such character is present.

```
I = INDEXB(STR,SSTR)
```

sets I equal to the position of the first occurrence of string SSTR in string STR scanning backwards. I = 0 if no such string is present.

```
I = INDEXC(STR,SSTR)
```

sets I equal to the leftmost position where string SSTR does not match a substring in STR. I = 0 if there is no such mismatch.

```
I = INDEXN(STR)
```

sets I equal to the position of the first numeric character in STR. I = 0 if no such character is present.

```
I = INDEXS(STR)
```

sets I to the position of the first special (i.e. non-alphanumeric) character in STR. I = 0 if no such character is present.

```
I = INDXAC(STR)
```

sets I equal to the position of the first non-alphabetic character (upper or lower case) in STR. I = 0 if no such character is present.

```
I = INDXBC(STR,SSTR)
```

sets `I` equal to the position of the first mismatch of string `SSTR` with respect to string `STR` scanning backwards. `I` = `0` if there is no such mismatch.

```
I = INDXNC(STR)
```

sets `I` equal to the position of the first non-numeric character in `STR`. `I` = `0` if no such character is present.

```
I = ISCAN(STR,SET)
```

sets `I` to the leftmost position where any of the characters in `SET` matches a character in `STR`. `I` = `0` if there is no such match.

```
H = REPEAT(STR,NTIMES)
```

sets `H` equal to `NTIMES` concatenated copies of the string `STR`.

```
H = SPACES(STR,NSPACE)
```

sets `H` equal to a character string equivalent to `STR` with leading blanks removed and each occurence of one or more blanks inside `STR` replaced by `NSPACE` blanks.

```
H = STRIP(STR,CHOPT,CHD)
```

sets `H` to a character string equivalent to `STR` with leading and trailing occurances of the character `CHD` removed. If `CHOPT` is equal to `'L'`, only leading characters will be removed. If `CHOPT` is equal to `'T'`, only trailing characters will be removed.

```
H = SUBWORD(STR,IW,NW)
```

sets `H` equal to the character string starting with word `IW` of `STR` and containing `NW` words.

```
I = VERIFY(STR,SET)
```

sets `I` to the leftmost position of any character in `STR` which is not part of `SET`.

```
H = WORD(STR,IW)
```

sets `H` equal to the word `IW` of `STR`.

```
I = WORDS(STR)
```

sets `I` to the number of words in `STR`.

```
CALL WORDSEP(STR)
```

sets the word separator for `SUBWORD`, `WORD` and `WORDS` to the first character of the string `STR`.

Examples:

Assume the following declarations:

```
CHARACTER STR*41,REP10*10,REP17*17
CHARACTER REPEAT*16,SPAC17*17,SPAC30*30,SPACES*20
INTEGER VERIFY
```

and a string `STR` defined as:

```
DATA STR /'A B C 1 2 3 A B C 1 2 3 A B C 1 2 3 A B C'/
```

The following results are obtained:

Statement/ Expression	Yields the value
`REP10 = REPEAT('ABC',5)`	`'ABCABCABCA'`
`REP17 = REPEAT('ABC',5)`	`'ABCABCABCABCABC  '`
`REP17 = REPEAT('ABC',6)`	`'ABCABCABCABCABCAB'`

```
INDEXB(STR,' ')              40
INDEXC(STR,' ')               1
INDXBC(STR,' ')              41
INDEXA(STR)                   1
INDXAC(STR)                   2
INDEXN(STR)                   7
INDXNC(STR)                   1
INDEXS(STR)                   2
ISCAN(STR,' ')                2
VERIFY(STR,' ')               1

INDEXB(STR,'1 2 3')          31
INDEXC(STR,'1 2 3')           1
INDXBC(STR,'1 2 3')          37
ISCAN(STR,'123')              7
VERIFY(STR,'123')             1
INDEXB(STR,'A B C')          31
INDEXC(STR,'A B C')           2
INDXBC(STR,'A B C')          36
ISCAN(STR,'ABC')              1
VERIFY(STR,'ABC')             2

SPAC17=SPACES(STR,0)          'ABC123ABC123ABC12'
SPAC30=SPACES(STR,2)          'A  B  C  1  2  3  A  B  C  1  2'
```

•