

C LIBRARY FUNCTION - MBLEN

http://www.tutorialspoint.com/c_standard_library/c_function_mblen.htm

Copyright © tutorialspoint.com

Description

The C library function **int mblen***constchar * str, size_t n* returns the length of a multi-byte character pointed to, by the argument **str**.

Declaration

Following is the declaration for mblen function.

```
int mblen(const char *str, size_t n)
```

Parameters

- **str** -- This is the pointer to the first byte of a multibyte character.
- **n** -- This is the maximum number of bytes to be checked for character length.

Return Value

The mblen function returns the number of bytes passed from the multi-byte sequence starting at str, if a non-null wide character was recognized. It returns 0, if a null wide character was recognized. It returns -1, if an invalid multi-byte sequence was encountered or if it could not parse a complete multi-byte character.

Example

The following example shows the usage of mblen function.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{
    int len;
    char *pmbnull = NULL;
    char *pmb = (char *)malloc( MB_CUR_MAX );
    wchar_t *pwc = L"Hi";
    wchar_t *pwcs = (wchar_t *)malloc( sizeof( wchar_t ) );

    printf("Converting to multibyte string\n");
    len = wcstombs( pmb, pwc, MB_CUR_MAX );
    printf("Characters converted %d\n", len);
    printf("Hex value of first multibyte character: %#.4x\n", pmb);

    len = mblen( pmb, MB_CUR_MAX );
    printf( "Length in bytes of multibyte character %x: %u\n", pmb, len );

    pmb = NULL;

    len = mblen( pmb, MB_CUR_MAX );
    printf( "Length in bytes of multibyte character %x: %u\n", pmb, len );

    return(0);
}
```

Let us compile and run the above program that will produce the following result:

```
Converting to multibyte string
Characters converted 1
```

Hex value of first multibyte character: 0x168c6010
Length in bytes of multibyte character 168c6010: 1
~~Length in bytes of multibyte character 0: 0~~

Loading [Mathjax]/jax/output/HTML-CSS/jax.js