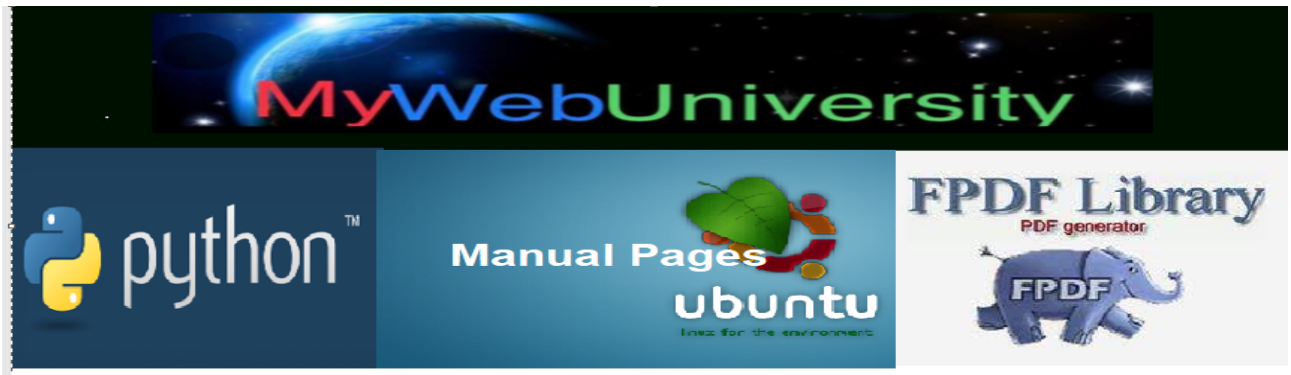

Full credit is given to the above companies including the OS that this PDF file was generated!

Linux Ubuntu 26.04 Manual Pages on command 'mbsnrtowcs.3'

$ man mbsnrtowcs.3

mbsnrtowcs(3) Library Functions Manual mbsnrtowcs(3)

NAME

mbsnrtowcs - convert a multibyte string to a wide-character string

LIBRARY

Standard C library (libc, -lc)

SYNOPSIS

```
#include <wchar.h>

size_t mbsnrtowcs(size_t size;
                  wchar_t dest[restrict size], const char **restrict src,
                  size_t nms, size_t size, mbstate_t *restrict ps);
```

Feature Test Macro Requirements for glibc (see feature_test_macros(7)):

mbsnrtowcs():

Since glibc 2.10:

_POSIX_C_SOURCE >= 200809L

Before glibc 2.10:

_GNU_SOURCE

DESCRIPTION

The mbsnrtowcs() function is like the mbsrtowcs(3) function, except that the number of bytes to be converted, starting at *src, is limited to at most nms bytes.

If dest is not NULL, the mbsnrtowcs() function converts at most nms bytes from the multibyte string *src to a wide-character string starting at dest. At most size wide characters are written to dest. The shift state *ps is updated. The conversion is effectively performed by repeatedly calling mbrtowc(dest, *src, n, ps) where n is some positive number, as long as

this call succeeds, and then incrementing dest by one and *src by the number of bytes con?
sumed. The conversion can stop for three reasons:

? An invalid multibyte sequence has been encountered. In this case, *src is left pointing to the invalid multibyte sequence, (size_t) -1 is returned, and errno is set to EILSEQ.

? The nms limit forces a stop, or size non-L'\0' wide characters have been stored at dest. In this case, *src is left pointing to the next multibyte sequence to be converted, and the number of wide characters written to dest is returned.

? The multibyte string has been completely converted, including the terminating null wide character ('\0') (which has the side effect of bringing back *ps to the initial state). In this case, *src is set to NULL, and the number of wide characters written to dest, ex? cluding the terminating null wide character, is returned.

According to POSIX.1, if the input buffer ends with an incomplete character, it is unspeci? fied whether conversion stops at the end of the previous character (if any), or at the end of the input buffer. The glibc implementation adopts the former behavior.

If dest is NULL, size is ignored, and the conversion proceeds as above, except that the con? verted wide characters are not written out to memory, and that no destination size limit ex? ists.

In both of the above cases, if ps is NULL, a static anonymous state known only to the mbsnr? towcs() function is used instead.

The programmer must ensure that there is room for at least size wide characters at dest.

RETURN VALUE

The mbsnrtowcs() function returns the number of wide characters that make up the converted part of the wide-character string, not including the terminating null wide character. If an invalid multibyte sequence was encountered, (size_t) -1 is returned, and errno set to EILSEQ.

ATTRIBUTES

For an explanation of the terms used in this section, see attributes(7).

Interface	Attribute	Value
mbsnrtowcs()	Thread safety	MT-Unsafe race:mbsnrtowcs/!ps

STANDARDS

POSIX.1-2008.

NOTES

The behavior of mbsnrtowcs() depends on the LC_CTYPE category of the current locale.

Passing NULL as ps is not multithread safe.

SEE ALSO

iconv(3), mbrtowc(3), mbsinit(3), mbsrtowcs(3)