NAME

wcstok - split wide-character string into tokens

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

#include <wchar.h>

*wchar_t **
wcstok(*wchar_t * restrict str, const wchar_t * restrict sep, wchar_t ** restrict last*);

DESCRIPTION

The **wcstok**() function is used to isolate sequential tokens in a null-terminated wide character string, *str*. These tokens are separated in the string by at least one of the characters in *sep*. The first time that **wcstok**() is called, *str* should be specified; subsequent calls, wishing to obtain further tokens from the same string, should pass a null pointer instead. The separator string, *sep*, must be supplied each time, and may change between calls. The context pointer *last* must be provided on each call.

The **wcstok**() function is the wide character counterpart of the **strtok_r**() function.

RETURN VALUES

The **wcstok**() function returns a pointer to the beginning of each subsequent token in the string, after replacing the token itself with a null wide character (L'\0'). When no more tokens remain, a null pointer is returned.

EXAMPLES

The following code fragment splits a wide character string on ASCII space, tab and newline characters and writes the tokens to standard output:

```
const wchar_t *seps = L" \t\n";
wchar_t *last, *tok, text[] = L" \none\ttwo\t\tthree  \n";

for (tok = wcstok(text, seps, &last); tok != NULL;
    tok = wcstok(NULL, seps, &last))
        wprintf(L"%ls\n", tok);
```

COMPATIBILITY

Some early implementations of **wcstok**() omit the context pointer argument, *last*, and maintain state across calls in a static variable like **strtok**() does.

SEE ALSO

strtok(3), wcschr(3), wcscspn(3), wcspbrk(3), wcsrchr(3), wcsspn(3)

STANDARDS

The **wcstok**() function conforms to ISO/IEC 9899:1999 ("ISO C99").