

**NAME**

**wcrtomb**, **c16rtomb**, **c32rtomb** - convert a wide-character code to a character (restartable)

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

```
#include <wchar.h>
```

*size\_t*

```
wcrtomb(char * restrict s, wchar_t c, mbstate_t * restrict ps);
```

```
#include <uchar.h>
```

*size\_t*

```
c16rtomb(char * restrict s, char16_t c, mbstate_t * restrict ps);
```

*size\_t*

```
c32rtomb(char * restrict s, char32_t c, mbstate_t * restrict ps);
```

**DESCRIPTION**

The **wcrtomb()**, **c16rtomb()** and **c32rtomb()** functions store a multibyte sequence representing the wide character *c*, including any necessary shift sequences, to the character array *s*, storing a maximum of MB\_CUR\_MAX bytes.

If *s* is NULL, these functions behave as if *s* pointed to an internal buffer and *c* was a null wide character (L'\0').

The *mbstate\_t* argument, *ps*, is used to keep track of the shift state. If it is NULL, these functions use an internal, static *mbstate\_t* object, which is initialized to the initial conversion state at program startup.

As certain multibyte characters may only be represented by a series of 16-bit characters, the **c16rtomb()** may need to be invoked multiple times before a multibyte sequence is returned.

**RETURN VALUES**

These functions return the length (in bytes) of the multibyte sequence needed to represent *c*, or (*size\_t*)-1 if *c* is not a valid wide character code.

**ERRORS**

The **wcrtomb()**, **c16rtomb()** and **c32rtomb()** functions will fail if:

[EILSEQ] An invalid wide character code was specified.

[EINVAL] The conversion state is invalid.

### SEE ALSO

`mbrtowc(3)`, `multibyte(3)`, `setlocale(3)`, `wctomb(3)`

### STANDARDS

The `wrtomb()`, `c16rtomb()` and `c32rtomb()` functions conform to ISO/IEC 9899:2011 ("ISO C11").