

**NAME**

**get\_wch**, **wget\_wch**, **mvget\_wch**, **mvwget\_wch**, **unget\_wch** - get (or push back) a wide character from *curses* terminal keyboard

**SYNOPSIS**

```
#include <curses.h>
```

```
int get_wch(wint_t *wch);
int wget_wch(WINDOW *win, wint_t *wch);
int mvget_wch(int y, int x, wint_t *wch);
int mvwget_wch(WINDOW *win, int y, int x, wint_t *wch);

int unget_wch(const wchar_t wc);
```

**DESCRIPTION****Reading Characters**

**wget\_wch** gathers a key stroke *wch* from the terminal keyboard associated with a *curses* window *win*, returning **OK** if a wide character is read, **KEY\_CODE\_YES** if a function key is read, and **ERR** if no key event is available. **ncurses(3X)** describes the variants of this function.

When input is pending, **wget\_wch** stores an integer identifying the key stroke in *wch*; for alphanumeric and punctuation keys, this value corresponds to the character encoding used by the terminal. Use of the control key as a modifier often results in a distinct code. The behavior of other keys depends on whether *win* is in keypad mode; see subsections "Keypad Mode" and "Predefined Key Codes" in **getch(3X)**.

If no input is pending, then if the no-delay flag is set in the window (see **nodelay(3X)**), the function returns **ERR**; otherwise, *curses* waits until the terminal has input. If **cbreak(3X)** has been called, this happens after one character is read. If **nocbreak(3X)** has been called, it occurs when the next newline is read. If **halfdelay(3X)** has been called, *curses* waits until a character is typed or the specified delay elapses.

If **echo(3X)** has been called, and the window is not a pad, *curses* writes *wch* to the window (at the cursor position) per the following rules.

- ⊕ If *wch* matches the terminal's erase character, the cursor moves leftward one position and the new position is erased as if **wmove(3X)** and then **wdelch(3X)** were called. When the window's keypad mode is enabled (see below), **KEY\_LEFT** and **KEY\_BACKSPACE** are handled the same way.
- ⊕ *curses* writes any other *wch* to the window, as with **wecho\_wchar(3X)**.

- ⊕ If the window has been moved or modified since the last call to **wrefresh(3X)**, *curses* calls **wrefresh**.

If *wch* is a carriage return and **nl(3X)** has been called, **wgetch** stores the the character code for newline (line feed) in *wch* instead.

### Ungetting Characters

**unget\_wch** places *wch* into the input queue to be returned by the next call to **wget\_wch**. A single input queue serves all windows.

### RETURN VALUE

**wget\_wch** returns **OK** when it reads a wide character and **KEY\_CODE\_YES** when it reads a function key code. It returns **ERR** if

- ⊕ the *WINDOW* pointer is **NULL**, or
- ⊕ its timeout expires without any data arriving, or
- ⊕ execution was interrupted by a signal, in which case **errno** is set to **EINTR**.

Functions prefixed with "mv" first perform cursor movement and fail if the position (*y*, *x*) is outside the window boundaries.

**unget\_wch** returns **OK** on success and **ERR** if there is no more room in the input queue.

### NOTES

See the "NOTES" section of **wgetch(3X)**.

All of these functions except **wget\_wch** and **unget\_wch** may be implemented as macros.

Unlike **wgetch(3X)**, **wget\_wch** and its variants store the value of the input character in an additional *wch* parameter instead of the return value.

Unlike **ungetch**, **unget\_wch** cannot distinguish function key codes **wget\_wch** from conventional character codes. An application can overcome this limitation by pushing function key codes with **ungetch** and subsequently checking the return value of **wget\_wch** for a match with **KEY\_CODE\_YES**.

### EXTENSIONS

See the "EXTENSIONS" section of **wgetch(3X)**.

**PORTABILITY**

Applications employing *ncurses* extensions should condition their use on the visibility of the **NCURSES\_VERSION** preprocessor macro.

X/Open Curses, Issue 4 describes these functions. It specifies no error conditions for them.

See the "PORTABILITY" section of **wgetch(3X)** regarding the interaction of **wget\_wch** with signal handlers.

**SEE ALSO**

**curs\_getch(3X)** describes comparable functions of the *ncurses* library in its non-wide-character configuration.

**curses(3X)**, **curs\_add\_wch(3X)**, **curs\_inopts(3X)**, **curs\_move(3X)**, **curs\_refresh(3X)**