NAME

bkgdset, **wbkgdset**, **bkgd**, **wbkgd**, **getbkgd** − manipulate background of a *curses* window of characters

SYNOPSIS

```
#include <curses.h>

int bkgd(chtype ch);
int wbkgd(WINDOW *win, chtype ch);

void bkgdset(chtype ch);
void wbkgdset(WINDOW *win, chtype ch);

chtype getbkgd(WINDOW *win);
```

DESCRIPTION

Every *curses* window has a *background character* property: in the library's non-wide-character configuration, it is a *curses* character (*chtype*) that combines a set of attributes (and, if colors are enabled, a color pair identifier) with a character code. When erasing (parts of) a window, *curses* replaces the erased cells with the background character.

curses also uses the background character when writing characters to a populated window.

- The attribute part of the background character combines with all non-blank character cells in the window, as populated by the **waddch**(3X) and **winsch**(3X) families of functions (and those that call them).
- Both the character code and attributes of the background character combine with blank character cells in the window.

The background character's set of attributes becomes a property of the character cell and move with it through any scrolling and insert/delete line/character operations. To the extent possible on the terminal type, *curses* displays the attributes of the background character as the graphic rendition of a character cell on the display.

bkgd, wbkgd

bkgd and **wbkgd** set the background property of **stdscr** or the specified window and then apply this setting to every character cell in that window.

- The rendition of every character in the window changes to the new background rendition.
- Wherever the former background character appears, it changes to the new background character.

ncurses updates the rendition of each character cell by comparing the character, non-color attributes, and color pair selection. The library applies the following procedure to each cell in the window, whether or not it is blank.

- *ncurses* first compares the cell's character to the previously specified background character; if they match, *ncurses* writes the new background character to the cell.
- *ncurses* then checks whether the cell uses color; that is, its color pair value is nonzero. If not, it simply replaces the attributes and color pair in the cell with those from the new background character.
- If the cell uses color, and its background color matches that of the current window background, *ncurses* removes attributes that may have come from the current background and adds those from the new background. It finishes by setting the cell's background to use the new window background color.
- If the cell uses color, and its background color does not match that of the current window background, *ncurses* updates only the non-color attributes, first removing those that may have come from the current background, and then adding attributes from the new background.

If the new background's character is non-spacing (for example, if it is a control character), *ncurses* retains the existing background character, except for one special case: *ncurses* treats a background character code of zero (0) as a space.

If the terminal does not support color, or if color has not been initialized with **start_color**(3X), *ncurses* ignores the new background character's color pair selection.

bkgdset, wbkgdset

bkgdset and **wbkgdset** manipulate the background of the applicable window, without updating the character cells as **bkgd** and **wbkgd** do; only future writes reflect the updated background.

getbkgd

getbkgd returns the given window's background character, attributes, and color pair as a *chtype*.

RETURN VALUE

bkgdset and **wbkgdset** do not return a value.

Functions returning an *int* return **ERR** upon failure and **OK** upon success. In *ncurses*, failure occurs if

- the *curses* screen has not been initialized, or
- *win* is *NULL*.

getbkgd's return value is as described above.

NOTES

Unusually, there is no **wgetbkgd** function; **getbkgd** behaves as one would expect **wgetbkgd** to, accepting a *WINDOW* pointer argument.

bkgd and **bkgdset** may be implemented as macros.

X/Open Curses mentions that the character part of the background must be a single-byte value. *ncurses*, like SVr4 *curses*, checks to ensure that it is, and retains the existing background character if the check fails.

PORTABILITY

X/Open Curses Issue 4 describes these functions. It indicates that *bkgd*, *wbkgd*, and *getbkgd* return *ERR* on failure (in the case of the last, this value is cast to *chtype*), but specifies no error conditions for them.

SVr4 documentation says that *bkgd* and *wbkgd* return *OK* "or a non-negative integer if *immedok()* is set", referring to the return value from *wrefresh*, which in SVr4 returns a count of characters written to the window if its *immedok* property is set; in *ncurses*, it does not.

Neither X/Open Curses nor the SVr4 manual pages detail how the rendition of characters in the window updates when *bkgd* or *wbkgd* changes the background character. *ncurses*, like SVr4 *curses*, does not (in its non-wide-character configuration) store the background and window attribute contributions to each character cell separately.

HISTORY

SVr3.1 (1987) introduced these functions.

SEE ALSO

curs_bkgrnd(3X) describes the corresponding functions in the wide configuration of *ncurses*.

curses(3X), **curs_addch**(3X), **curs_attr**(3X)