NAME

curs_addch, addch, waddch, mvaddch, mvwaddch, echochar, wechochar – add a character (with attributes) to a curses window and advance cursor

SYNOPSIS

cc [*flag* …] *file* … –**lcurses** [*library* ..]

#include <curses.h>

int addch(chtype *ch***);**

int waddch(WINDOW **win***, chtype** *ch***);**

int mvaddch(int *y***, int** *x***, chtype** *ch***);**

int mvwaddch(WINDOW **win***, int** *y***, int** *x***, chtype** *ch***);**

int echochar(chtype *ch***);**

int wechochar(WINDOW **win,* **chtype** *ch);*

MT-LEVEL

Unsafe

DESCRIPTION

With the **addch( )**, **waddch( )**, **mvaddch( )**, and **mvwaddch( )** routines, the character *ch* is put into the window at the current cursor position of the window and the position of the window cursor is advanced. Its function is similar to that of **putchar( )**. At the right margin, an automatic newline is performed. At the bottom of the scrolling region, if **scrollok( )** is enabled, the scrolling region is scrolled up one line.

If *ch* is a tab, newline, or backspace, the cursor is moved appropriately within the window. A newline also does a **clrtoeol( )** before moving. Tabs are considered to be at every eighth column. If *ch* is another control character, it is drawn in the ^*X* notation. Calling **winch( )** after adding a control character does not return the control character, but instead returns the representation of the control character. See **curs_inch**(3X).

Video attributes can be combined with a character by OR-ing them into the parameter. This results in these attributes also being set. (The intent here is that text, including attributes, can be copied from one place to another using **inch( )** and **addch( )**.) (see **standout( )**, predefined video attribute constants, on the **curs_attr**(3X) page).

The **echochar( )** and **wechochar( )** routines are functionally equivalent to a call to **addch( )** followed by a call to **refresh( )**, or a call to **waddch** followed by a call to **wrefresh( )**. The knowledge that only a single character is being output is taken into consideration and, for non-control characters, a considerable performance gain might be seen by using these routines instead of their equivalents.

Line Graphics

The following variables may be used to add line drawing characters to the screen with routines of the **addch( )** family. When variables are defined for the terminal, the **A_ALTCHARSET** bit is turned on (see **curs_attr**(3X)). Otherwise, the default character listed below is stored in the variable. The names chosen are consistent with the VT100 nomenclature.

Name	*Default*	*Glyph Description*
ACS_ULCORNER	+	upper left-hand corner
ACS_LLCORNER	+	lower left-hand corner
ACS_URCORNER	+	upper right-hand corner
ACS_LRCORNER	+	lower right-hand corner
ACS_RTEE	+	right tee (┤)
ACS_LTEE	+	left tee (├)
ACS_BTEE	+	bottom tee (┴)
ACS_TTEE	+	top tee (┬)
ACS_HLINE	–	horizontal line
ACS_VLINE	|	vertical line
ACS_PLUS	+	plus
ACS_S1	–	scan line 1
ACS_S9	_	scan line 9
ACS_DIAMOND	+	diamond
ACS_CKBOARD	:	checker board (stipple)
ACS_DEGREE	′	degree symbol
ACS_PLMINUS	#	plus/minus
ACS_BULLET	o	bullet
ACS_LARROW	<	arrow pointing left
ACS_RARROW	>	arrow pointing right
ACS_DARROW	v	arrow pointing down
ACS_UARROW	^	arrow pointing up
ACS_BOARD	#	board of squares
ACS_LANTERN	#	lantern symbol
ACS_BLOCK	#	solid square block

RETURN VALUES

All routines return the integer **ERR** upon failure and an integer value other than **ERR** upon successful completion.

SEE ALSO

curs_attr(3X), **curs_clear**(3X), **curs_inch**(3X), **curs_outopts**(3X), **curs_refresh**(3X), **curses**(3X), **putc**(3S)

NOTES

The header **<curses.h>** automatically includes the headers **<stdio.h>** and **<unctrl.h>**.

Note that **addch( )**, **mvaddch( )**, **mvwaddch( )**, and **echochar( )** may be macros.