NAME

curs_refresh, refresh, wrefresh, wnoutrefresh, doupdate, redrawwin, wredrawln – refresh curses windows and lines

SYNOPSIS

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

#include <curses.h>

int refresh(void);

int wrefresh(WINDOW **win***);**

int wnoutrefresh(WINDOW **win***);**

int doupdate(void);

int redrawwin(WINDOW **win***);**

int wredrawln(WINDOW **win***, int** *beg_line***, int** *num_lines***);**

MT-LEVEL

Unsafe

DESCRIPTION

The **refresh( )** and **wrefresh( )** routines (or **wnoutrefresh( )** and **doupdate( )**) must be called to get any output on the terminal, as other routines merely manipulate data structures. The routine **wrefresh( )** copies the named window to the physical terminal screen, taking into account what is already there in order to do optimizations. The **refresh( )** routine is the same, using **stdscr** as the default window. Unless **leaveok( )** has been enabled, the physical cursor of the terminal is left at the location of the cursor for that window.

The **wnoutrefresh( )** and **doupdate( )** routines allow multiple updates with more efficiency than **wrefresh( )** alone. In addition to all the window structures, **curses** keeps two data structures representing the terminal screen: a physical screen, describing what is actually on the screen, and a virtual screen, describing what the programmer wants to have on the screen.

The routine **wrefresh( )** works by first calling **wnoutrefresh( )**, which copies the named window to the virtual screen, and then calling **doupdate( )**, which compares the virtual screen to the physical screen and does the actual update. If the programmer wishes to output several windows at once, a series of calls to **wrefresh( )** results in alternating calls to **wnoutrefresh( )** and **doupdate( )**, causing several bursts of output to the screen. By first calling **wnoutrefresh( )** for each window, it is then possible to call **doupdate( )** once, resulting in only one burst of output, with fewer total characters transmitted and less CPU time used. If the *win* argument to **wrefresh( )** is the global variable **curscr**, the screen is immediately cleared and repainted from scratch.

The **redrawwin( )** routine indicates to **curses** that some screen lines are corrupted and should be thrown away before anything is written over them. These routines could be used for programs such as editors, which want a command to redraw some part of the screen or the entire screen. The routine **redrawln( )** is preferred over **redrawwin( )** where a noisy communication line exists and redrawing the entire window could be subject to even more communication noise. Just redrawing several lines offers the possibility that they would show up unblemished.

RETURN VALUES

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

SEE ALSO

curs_outopts(3X), **curses**(3X)

NOTES

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

Note that **refresh( )** and **redrawwin( )** may be macros.