NAME

getstr, **getnstr**, **wgetstr**, **wgetnstr**, **mvgetstr**, **mvgetnstr**, **mvwgetstr**, **mvwgetnstr** − accept character strings from **curses** terminal keyboard

SYNOPSIS

#include <curses.h>

int getstr(char *str);
int getnstr(char *str, int n);
int wgetstr(WINDOW *win, char *str);
int wgetnstr(WINDOW *win, char *str, int n);
int mvgetstr(int y, int x, char *str);
int mvwgetstr(WINDOW *win, int y, int x, char *str);
int mvgetnstr(int y, int x, char *str, int n);
int mvwgetnstr(WINDOW *, int y, int x, char *str, int n);

DESCRIPTION

The function **getstr** is equivalent to a series of calls to **getch**, until a newline or carriage return is received (the terminating character is not included in the returned string). The resulting value is placed in the area pointed to by the character pointer *str*.

wgetnstr reads at most *n* characters, thus preventing a possible overflow of the input buffer. Any attempt to enter more characters (other than the terminating newline or carriage return) causes a beep. Function keys also cause a beep and are ignored. The **getnstr** function reads from the *stdscr* default window.

The user's erase and kill characters are interpreted. If keypad mode is on for the window, **KEY_LEFT** and **KEY_BACKSPACE** are both considered equivalent to the user's kill character.

Characters input are echoed only if **echo** is currently on. In that case, backspace is echoed as deletion of the previous character (typically a left motion).

RETURN VALUE

All routines return the integer **ERR** upon failure and an **OK** (SVr4 specifies only "an integer value other than **ERR**") upon successful completion.

X/Open defines no error conditions.

In this implementation, these functions return an error if the window pointer is null, or if its timeout expires without having any data.

This implementation provides an extension as well. If a SIGWINCH interrupts the function, it will return **KEY_RESIZE** rather than **OK** or **ERR**.

Functions with a "mv" prefix first perform a cursor movement using **wmove**, and return an error if the position is outside the window, or if the window pointer is null.

NOTES

Note that **getstr**, **mvgetstr**, and **mvwgetstr** may be macros.

PORTABILITY

These functions are described in the XSI Curses standard, Issue 4. They read single-byte characters only. The standard does not define any error conditions. This implementation returns ERR if the window pointer is null, or if the lower-level **wgetch** call returns an ERR.

SVr3 and early SVr4 curses implementations did not reject function keys; the SVr4.0 documentation claimed that "special keys" (such as function keys, "home" key, "clear" key, *etc*.) are "interpreted", without giving details. It lied. In fact, the 'character' value appended to the string by those implementations was predictable but not useful (being, in fact, the low-order eight bits of the key's KEY_ value).

The functions **getnstr**, **mvgetnstr**, and **mvwgetnstr** were present but not documented in SVr4.

SEE ALSO

ncurses(3NCURSES), **getch**(3NCURSES), **curses_variables**(3NCURSES).

