

NAME

insstr, **winsstr**, **mvinsstr**, **mvinsnstr**, **insnstr**, **winsnstr**, **mvwinsstr**, **mvwinsnstr** – insert a string in a *curses* window

SYNOPSIS

```
#include <curses.h>

int insstr(const char * str);
int winsstr(WINDOW * win, const char * str);
int mvinsstr(int y, int x, const char * str);
int mvwinsstr(WINDOW * win, int y, int x, const char * str);

int insnstr(const char * str, int n);
int winsnstr(WINDOW * win, const char * str, int n);
int mvinsnstr(int y, int x, const char * str, int n);
int mvwinsnstr(WINDOW * win, int y, int x, const char * str, int n);
```

DESCRIPTION

winsstr inserts a string *str* before the character at the cursor in window *win* as if by calling **winsch**(3X) for each *char* in *str*. No line wrapping is performed. Characters to the right of the cursor are shifted right; those at the right edge of the window may be lost. **winsstr** stops inserting if it would have to wrap to the next line to write the next *char* in *str*. The cursor position does not change (after moving to (y, x), if specified). **insnstr** does the same, but inserts at most *n* characters, or as many as possible (up to the end of the line) if *n* is negative. **ncurses**(3X) describes the variants of these functions.

RETURN VALUE

These functions return **OK** on success and **ERR** on failure.

In *ncurses*, they return **ERR** if

- *win* is **NULL**,
- *str* is **NULL**, or
- an internal **winsch**(3X) call returns **ERR**.

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

NOTES

All of these functions except **winsnstr** may be implemented as macros.

EXTENSIONS

insnstr, **winsnstr**, **mvinsnstr**, and **mvwinsnstr**’s acceptance of negative *n* values is an *ncurses* extension.

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.

SVr4 describes a successful return value only as “an integer value other than **ERR**”.

Issue 4 distinguished *insnstr* and *winsnstr* from the other functions documented above by stating they “do not perform wrapping”. This was probably an error, since it makes this group of functions inconsistent. No implementation of *curses* manifests this inconsistency, and Issue 7 removed the claim.

Issue 4 states that the entire string is inserted if *n* is less than 1. This is probably an error, because it is inconsistent with other functions such as *waddstr*, and differs from the SVr4 *curses* and Solaris *xcurses* implementations. Nevertheless, Issue 7 retains the language.

HISTORY

SVr3.1 (1987) introduced *winsstr* and *winsnstr* and their variants.

SEE ALSO

curs_ins_wstr(3X) describes comparable functions of the *ncurses* library in its wide-character configuration (*ncursesw*).

curses(3X), **curs_inch**(3X), **curs_ins_wstr**(3X), **curs_util**(3X)