

**NAME**

**instr**, **winstr**, **mvwinstr**, **mvwinstr**, **innstr**, **winnstr**, **mvinnstr**, **mvwinnstr** – get a string from a *curses* window

**SYNOPSIS**

```
#include <curses.h>

int instr(char * str);
int winstr(WINDOW * win, char * str);
int mvwinstr(int y, int x, char * str);
int mvwinstr(WINDOW * win, int y, int x, char * str);

int innstr(char * str, int n);
int winnstr(WINDOW * win, char * str, int n);
int mvinnstr(int y, int x, char * str, int n);
int mvwinnstr(WINDOW * win, int y, int x, char * str, int n);
```

**DESCRIPTION**

**winstr** extracts a string from a *curses* window *win*, starting at the cursor and stopping at the end of the line, and stores it in *str*, terminating it with a null character and omitting any attributes and color pair identifier that *curses* associates with each character. **winnstr** does the same, but copies at most *n* characters from *win*. A negative *n* implies no limit; **winnstr** then works like **winstr**. **ncurses(3X)** describes the variants of these functions.

**RETURN VALUE**

These functions return the count of characters copied from *win* to *str* (not including the null terminator), or **ERR** upon failure.

In *ncurses*, these functions fail if

- the *curses* screen has not been initialized,
- (for functions taking a *WINDOW* pointer argument) *win* is a null pointer, or
- *str* is a null pointer.

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 **winnstr** may be implemented as macros.

Reading a line that overflows the array pointed to by *str* with and its variants causes undefined results. Instead, use the *n*-infix functions with a positive *n* argument no larger than the size of the buffer backing *str*.

**EXTENSIONS**

**instr**, **winnstr**, **mvinnstr**, and **mvwinnstr**’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.

X/Open Curses Issues 4 and 7 both state that *instr*, *winstr*, *mvwinstr*, and *mvwinstr* return *OK* rather than a character count. This is likely an erratum.

- SVr3.1 and SVr4 implemented *winstr* as a wrapper around *winnstr*, returning the latter’s return value. X/Open Curses’s specification thus may have been an editorial solecism copied from System V’s documentation (see below) by X/Open, rather than an intentional change.
- *ncurses* retains compatibility with System V *curses* behavior.

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

SVr4 does not document whether *n* counts the null terminator that these functions write to *str*.

**HISTORY**

SVr3.1 (1987) introduced these functions.

**SEE ALSO**

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

**curses**(3X), **curl\_inch**(3X), **curl\_inchstr**(3X)