Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'strfmon.3p' command

$ man strfmon.3p

STRFMON(3P) POSIX Programmer's Manual STRFMON(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

strfmon, strfmon_l ? convert monetary value to a string

SYNOPSIS

```
#include <monetary.h>
ssize_t strfmon(char *restrict s, size_t maxsize,
    const char *restrict format, ...);
ssize_t strfmon_l(char *restrict s, size_t maxsize,
    locale_t locale, const char *restrict format, ...);
```

DESCRIPTION

The strfmon() function shall place characters into the array pointed to by s as controlled by the string pointed to by format. No more than maxsize bytes are placed into the array.

The format is a character string, beginning and ending in its initial state, if any, that contains two types of objects: plain characters, which are simply copied to the output stream, and conversion specifica? tions, each of which shall result in the fetching of zero or more argu? ments which are converted and formatted. The results are undefined if

there are insufficient arguments for the format. If the format is ex?
hausted while arguments remain, the excess arguments are simply ig?
nored.

The application shall ensure that a conversion specification consists of the following sequence:

* A '%' character
* Optional flags
* Optional field width
* Optional left precision
* Optional right precision
* A required conversion specifier character that determines the con?
  version to be performed

The strfmon_l() function shall be equivalent to the strfmon() function, except that the locale data used is from the locale represented by lo?
cale.

Flags

One or more of the following optional flags can be specified to control the conversion:

=f An '=' followed by a single character f which is used as the numeric fill character. In order to work with precision or width counts, the fill character shall be a single byte charac?
ter; if not, the behavior is undefined. The default numeric fill character is the <space>. This flag does not affect field width filling which always uses the <space>. This flag is ig?
nored unless a left precision (see below) is specified.

^ Do not format the currency amount with grouping characters. The default is to insert the grouping characters if defined for the current locale.

+ or (Specify the style of representing positive and negative cur?
rency amounts. Only one of '+' or '(' may be specified. If '+' is specified, the locale's equivalent of '+' and '-' are used (for example, in many locales, the empty string if positive and '-' if negative). If '(' is specified, negative amounts are en?

closed within parentheses. If neither flag is specified, the '+' style is used.

! Suppress the currency symbol from the output conversion.

- Specify the alignment. If this flag is present the result of the conversion is left-justified (padded to the right) rather than right-justified. This flag shall be ignored unless a field width (see below) is specified.

Field Width

w A decimal digit string w specifying a minimum field width in bytes in which the result of the conversion is right-justified (or left-justified if the flag '-' is specified). The default is 0.

Left Precision

#n A '#' followed by a decimal digit string n specifying a maximum number of digits expected to be formatted to the left of the radix character. This option can be used to keep the formatted output from multiple calls to the strfmon() function aligned in the same columns. It can also be used to fill unused positions with a special character as in "$***123.45". This option causes an amount to be formatted as if it has the number of digits specified by n. If more than n digit positions are re? quired, this conversion specification is ignored. Digit posi? tions in excess of those actually required are filled with the numeric fill character (see the =f flag above).

If grouping has not been suppressed with the '^' flag, and it is defined for the current locale, grouping separators are in? serted before the fill characters (if any) are added. Grouping separators are not applied to fill characters even if the fill character is a digit.

To ensure alignment, any characters appearing before or after the number in the formatted output such as currency or sign symbols are padded as necessary with <space> characters to make their positive and negative formats an equal length.

Right Precision

.p A <period> followed by a decimal digit string p specifying the number of digits after the radix character. If the value of the right precision p is 0, no radix character appears. If a right precision is not included, a default specified by the current locale is used. The amount being formatted is rounded to the specified number of digits prior to formatting.

Conversion Specifier Characters

The conversion specifier characters and their meanings are:

i The double argument is formatted according to the locale's in? ternational currency format (for example, in the US: USD 1,234.56). If the argument is ?Inf or NaN, the result of the conversion is unspecified.

n The double argument is formatted according to the locale's na? tional currency format (for example, in the US: $1,234.56). If the argument is ?Inf or NaN, the result of the conversion is unspecified.

% Convert to a '%'; no argument is converted. The entire conver? sion specification shall be %%.

Locale Information

The LC_MONETARY category of the current locale affects the behavior of this function including the monetary radix character (which may be dif? ferent from the numeric radix character affected by the LC_NUMERIC cat? egory), the grouping separator, the currency symbols, and formats. The international currency symbol should be conformant with the ISO 4217:2001 standard.

If the value of maxsize is greater than {SSIZE_MAX}, the result is im? plementation-defined.

The behavior is undefined if the locale argument to strfmon_l() is the special locale object LC_GLOBAL_LOCALE or is not a valid locale object handle.

RETURN VALUE

If the total number of resulting bytes including the terminating null

byte is not more than maxsize, these functions shall return the number of bytes placed into the array pointed to by s, not including the ter? minating NUL character. Otherwise, -1 shall be returned, the contents of the array are unspecified, and errno shall be set to indicate the error.

ERRORS

These functions shall fail if:

E2BIG Conversion stopped due to lack of space in the buffer.

The following sections are informative.

EXAMPLES

Given a locale for the US and the values 123.45, -123.45, and 3456.781, the following output might be produced. Square brackets ("[]") are used in this example to delimit the output.

%n	[$123.45]	Default formatting
	[-$123.45]	
	[$3,456.78]	
%11n	[$123.45]	Right align within an 11-character field
	[-$123.45]	
	[$3,456.78]	
%#5n	[$ 123.45]	Aligned columns for values up to 99999
	[-$ 123.45]	
	[$ 3,456.78]	
%=*#5n	[$***123.45]	Specify a fill character
	[-$***123.45]	
	[$*3,456.78]	
%=0#5n	[$000123.45]	Fill characters do not use grouping
	[-$000123.45]	even if the fill character is a digit
	[$03,456.78]	
%^#5n	[$ 123.45]	Disable the grouping separator
	[-$ 123.45]	
	[$ 3456.78]	
%^#5.0n	[$ 123]	Round off to whole units
	[-$ 123]	

[$ 3457]

%^#5.4n [$ 123.4500] Increase the precision

[-$ 123.4500]

[$ 3456.7810]

%(#5n [$ 123.45] Use an alternative pos/neg style

[($ 123.45)]

[$ 3,456.78]

%!(#5n [123.45] Disable the currency symbol

[(123.45)]

[3,456.78]

%-14#5.4n [$ 123.4500] Left-justify the output

[-$ 123.4500]

[$ 3,456.7810]

%14#5.4n [$ 123.4500] Corresponding right-justified output

[-$ 123.4500]

[$ 3,456.7810]

See also the EXAMPLES section in fprintf().

APPLICATION USAGE

None.

RATIONALE

None.

FUTURE DIRECTIONS

Lowercase conversion characters are reserved for future standards use and uppercase for implementation-defined use.

SEE ALSO

fprintf(), localeconv()

The Base Definitions volume of POSIX.1?2017, <monetary.h>

event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at http://www.opengroup.org/unix/online.html .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.ker?nel.org/doc/man-pages/reporting_bugs.html .