

NAME

`asprintf`, `vasprintf` – print to allocated string

SYNOPSIS

```
#define _GNU_SOURCE    /* See feature_test_macros(7) */
#include <stdio.h>

int asprintf(char **strp, const char *fmt, ...);
int vasprintf(char **strp, const char *fmt, va_list ap);
```

DESCRIPTION

The functions **`asprintf()`** and **`vasprintf()`** are analogs of **`sprintf(3)`** and **`vsprintf(3)`**, except that they allocate a string large enough to hold the output including the terminating null byte (`'\0'`), and return a pointer to it via the first argument. This pointer should be passed to **`free(3)`** to release the allocated storage when it is no longer needed.

RETURN VALUE

When successful, these functions return the number of bytes printed, just like **`sprintf(3)`**. If memory allocation wasn't possible, or some other error occurs, these functions will return `-1`, and the contents of *strp* are undefined.

ATTRIBUTES

For an explanation of the terms used in this section, see **`attributes(7)`**.

Interface	Attribute	Value
<code>asprintf()</code> , <code>vasprintf()</code>	Thread safety	MT-Safe locale

CONFORMING TO

These functions are GNU extensions, not in C or POSIX. They are also available under *BSD. The FreeBSD implementation sets *strp* to `NULL` on error.

SEE ALSO

`free(3)`, **`malloc(3)`**, **`printf(3)`**

COLOPHON

This page is part of release 5.05 of the Linux *man-pages* project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.