

[Skip site navigation](#) (1)[Skip section navigation](#) (2)

## Site Navigation

# FreeBSD Manual Pages

## strnstr

[home](#) | [help](#)

STRSTR(3)

Library Functions Manual

STRSTR(3)

### NAME

strstr, strcasestr, strnstr -- locate a substring in a string

### LIBRARY

Standard C Library (libc, -lc)

### SYNOPSIS

```
#include <string.h>
```

```
char *  
strstr(const char *big, const char *little);
```

```
char *  
strcasestr(const char *big, const char *little);
```

```
char *  
strnstr(const char *big, const char *little, size_t len);
```

```
#include <string.h>  
#include <xlocale.h>
```

```
char *  
strcasestr_l(const char *big, const char *little, locale_t loc);
```

### DESCRIPTION

The **strstr()** function locates the first occurrence of the null-terminated string *little* in the null-terminated string *big*.

The **strcasestr()** function is similar to **strstr()**, but ignores the case of both strings.

The **strcasestr\_l()** function does the same as **strcasestr()** but takes an explicit locale rather than using the current locale.

The **strnstr()** function locates the first occurrence of the null-terminated string *little* in the string *big*, where not more than *len* characters are searched. Characters that appear after a `'\0'` character are not searched. Since the **strnstr()** function is a FreeBSD specific API, it should only be used when portability is not a concern.

### RETURN VALUES

If *little* is an empty string, *big* is returned; if *little* occurs nowhere in *big*, NULL is returned; otherwise a pointer to the first character of the first occurrence of *little* is returned.

## EXAMPLES

The following sets the pointer *ptr* to the "**Bar Baz**" portion of *largestring*:

```
const char *largestring = "Foo Bar Baz";
const char *smallstring = "Bar";
char *ptr;

ptr = strstr(largestring, smallstring);
```

The following sets the pointer *ptr* to NULL, because only the first 4 characters of *largestring* are searched:

```
const char *largestring = "Foo Bar Baz";
const char *smallstring = "Bar";
char *ptr;

ptr = strnstr(largestring, smallstring, 4);
```

## SEE ALSO

[memchr\(3\)](#), [memmem\(3\)](#), [strchr\(3\)](#), [strcspn\(3\)](#), [strpbrk\(3\)](#), [strrchr\(3\)](#), [strsep\(3\)](#), [strspn\(3\)](#), [strtok\(3\)](#), [wcsstr\(3\)](#).

## STANDARDS

The **strstr()** function conforms to ISO/IEC 9899:1990 ("ISO C90").

## HISTORY

The **strnstr()** function was introduced by FreeBSD 4.5 and is non-standard.

FreeBSD 13.2

October 11, 2001

STRSTR(3)

---

[NAME](#) | [LIBRARY](#) | [SYNOPSIS](#) | [DESCRIPTION](#) | [RETURN VALUES](#) | [EXAMPLES](#) | [SEE ALSO](#) | [STANDARDS](#) | [HISTORY](#)

Want to link to this manual page? Use this URL:

<<https://man.freebsd.org/cgi/man.cgi?query=strnstr&sektion=3&manpath=FreeBSD+14.0-RELEASE+and+Ports>>

[home](#) | [help](#)

---

[Legal Notices](#) | © 1995-2023 The FreeBSD Project. All rights reserved.  
[Contact](#)