

strspn, wcssp, _mbssp

Returns the index of the first character in a string that does not belong to a set of characters.

```
size_t strspn(const char *string,const char *strCharSet );
```

```
size_t wcssp(const wchar_t *string,const wchar_t *strCharSet );
```

```
size_t _mbssp(const unsigned char *string,const unsigned char *strCharSet );
```

Parameters

string

Null-terminated string to search.

strCharSet

Null-terminated character set.

Return Value

Returns an integer value specifying the length of the substring in *string* that consists entirely of characters in *strCharSet*. If *string* begins with a character not in *strCharSet*, the function returns 0. No return value is reserved to indicate an error.

Remarks

The **strspn** function returns the index of the first character in *string* that does not belong to the set of characters in *strCharSet*. The search does not include terminating null characters.

wcssp and **_mbssp** are wide-character and multibyte-character versions of **strspn**. The arguments of **wcssp** are wide-character strings; those of **_mbssp** are multibyte-character strings. These three functions behave identically otherwise.

Generic-Text Routine Mappings

TCHAR.H routine	_UNICODE & _MBCS not defined	_MBCS defined	_UNICODE defined
_tcssp	strspn	_mbssp	wcssp

Requirements

Routine	Required header	Compatibility
---------	-----------------	---------------

strspn	<string.h>	ANSI, Win 98, Win Me, Win NT, Win 2000, Win XP
wcsspn	<string.h> or <wchar.h>	ANSI, Win 98, Win Me, Win NT, Win 2000, Win XP
_mbsspn	<mbstring.h>	Win 98, Win Me, Win NT, Win 2000, Win XP

For additional compatibility information, see [Compatibility](#) in the Introduction.

Libraries

All versions of the [C run-time libraries](#).

Example

```
// crt_strspn.c
/* This program uses strspn to determine
 * the length of the segment in the string "cabbage"
 * consisting of a's, b's, and c's. In other words,
 * it finds the first non-abc letter.
 */

#include <string.h>
#include <stdio.h>

int main( void )
{
    char string[] = "cabbage";
    int result;
    result = strspn( string, "abc" );
    printf( "The portion of '%s' containing only a, b, or c "
           "is %d bytes long\n", string, result );
}

/*
```

Output

The portion of 'cabbage' containing only a, b, or c is 5 bytes long

```
*/
```