Null-terminated wide strings

A null-terminated wide string is a sequence of valid wide characters, ending with a null-character.

Functions

Character classification

Defined in header <wctype.h>

iswalnum (C95)	checks if a wide character is alphanumeric (function)
iswalpha (C95)	checks if a wide character is alphabetic (function)
iswlower (C95)	checks if a wide character is an lowercase character (function)
iswupper (C95)	checks if a wide character is an uppercase character (function)
iswdigit (C95)	checks if a wide character is a digit (function)
iswxdigit (C95)	checks if a character is a hexadecimal character (function)
iswcntrl (C95)	checks if a wide character is a control character (function)
iswgraph (C95)	checks if a wide character is a graphical character (function)
iswspace (C95)	checks if a wide character is a space character (function)
iswblank (C99)	checks if a wide character is a blank character (function)
iswprint (C95)	checks if a wide character is a printing character (function)
iswpunct (C95)	checks if a wide character is a punctuation character (function)
iswctype (C95)	classifies a wide character according to the specified LC_CTYPE category (function)
wctype (C95)	looks up a character classification category in the current C locale (function)

Character manipulation

Defined in header <wctype.h>

towlower (C95)	converts a wide character to lowercase (function)
towupper (C95)	converts a wide character to uppercase (function)
towctrans (C95)	performs character mapping according to the specified LC_CTYPE mapping category (function)
wctrans (C95)	looks up a character mapping category in the current C locale (function)

ASCII values			characters	iscntrl iswcntrl	isprint iswprint	isspace iswspace	isblank iswblank	isgraph iswgraph	ispunct iswpunct	isalnum iswalnum	isalpha iswalpha	isupper iswupper	islower iswlower	isdigit iswdigit	isxdigit iswxdigit
decimal	hexadecimal	octal													
0–8	\x0-\x8	\0-\10	control codes (NUL, etc.)	≠0	0	0	0	0	0	0	0	0	0	0	0
9	\x9	\11	tab (\t)	≠0	0	≠0	≠0	0	0	0	0	0	0	0	0
10–13	\xA-\xD	\12-\15	whitespaces (\n, \v, \f, \r)	≠0	0	≠0	0	0	0	0	0	0	0	0	0
14–31	\xE-\x1F	\16-\37	control codes	≠0	0	0	0	0	0	0	0	0	0	0	0
32	\x20	\40	space	0	≠0	≠0	≠0	0	0	0	0	0	0	0	0
33–47	\x21-\x2F	\41-\57	!"#$%&'()*+,-./	0	≠0	0	0	≠0	≠0	0	0	0	0	0	0
48–57	\x30-\x39	\60-\71	0123456789	0	≠0	0	0	≠0	0	≠0	0	0	0	≠0	≠0
58–64	\x3A-\x40	\72-\100	:;<=>?@	0	≠0	0	0	≠0	≠0	0	0	0	0	0	0
65–70	\x41-\x46	\101-\106	ABCDEF	0	≠0	0	0	≠0	0	≠0	≠0	≠0	0	0	≠0
71–90	\x47-\x5A	\107-\132	GHIJKLMNOP QRSTUVWXYZ	0	≠0	0	0	≠0	0	≠0	≠0	≠0	0	0	0
91–96	\x5B-\x60	\133-\140	[\]^_`	0	≠0	0	0	≠0	≠0	0	0	0	0	0	0
97–102	\x61-\x66	\141-\146	abcdef	0	≠0	0	0	≠0	0	≠0	≠0	0	≠0	0	≠0
103–122	\x67-\x7A	\147-\172	ghijklmnop qrstuvwxyz	0	≠0	0	0	≠0	0	≠0	≠0	0	≠0	0	0
123–126	\x7B-\x7E	\172-\176	{|}~	0	≠0	0	0	≠0	≠0	0	0	0	0	0	0
127	\x7F	\177	backspace character (DEL)	≠0	0	0	0	0	0	0	0	0	0	0	0

Conversions to numeric formats

Defined in header <wchar.h>

wcstol (C95) **wcstoll** (C99)	converts a wide string to an integer value (function)
wcstoul (C95) **wcstoull** (C99)	converts a wide string to an unsigned integer value (function)
wcstof (C99) **wcstod** (C95) **wcstold** (C99)	converts a wide string to a floating-point value (function)

Defined in header <inttypes.h>

wcstoimax (C99) **wcstoumax** (C99)	converts a wide string to `intmax_t` or `uintmax_t` (function)

String manipulation

Defined in header <wchar.h>

wcscpy (C95) **wcscpy_s** (C11)	copies one wide string to another (function)
wcsncpy (C95) **wcsncpy_s** (C11)	copies a certain amount of wide characters from one string to another (function)
wcscat (C95) **wcscat_s** (C11)	appends a copy of one wide string to another (function)
wcsncat (C95) **wcsncat_s** (C11)	appends a certain amount of wide characters from one wide string to another (function)
wcsxfrm (C95)	transform a wide string so that wcscmp would produce the same result as wcscoll (function)

String examination

Defined in header <wchar.h>

wcslen (C95) **wcsnlen_s** (C11)	returns the length of a wide string (function)
wcscmp (C95)	compares two wide strings (function)
wcsncmp (C95)	compares a certain amount of characters from two wide strings (function)
wcscoll (C95)	compares two wide strings in accordance to the current locale (function)
wcschr (C95)	finds the first occurrence of a wide character in a wide string (function)
wcsrchr (C95)	finds the last occurrence of a wide character in a wide string (function)
wcsspn (C95)	returns the length of the maximum initial segment that consists of only the wide characters found in another wide string (function)
wcscspn (C95)	returns the length of the maximum initial segment that consists of only the wide chars *not* found in another wide string (function)
wcspbrk (C95)	finds the first location of any wide character in one wide string, in another wide string (function)
wcsstr (C95)	finds the first occurrence of a wide string within another wide string (function)
wcstok (C95) **wcstok_s** (C11)	finds the next token in a wide string (function)

Wide character array manipulation

Defined in header <wchar.h>

wmemcpy (C95) **wmemcpy_s** (C11)	copies a certain amount of wide characters between two non-overlapping arrays (function)
wmemmove (C95) **wmemmove_s** (C11)	copies a certain amount of wide characters between two, possibly overlapping, arrays (function)
wmemcmp (C95)	compares a certain amount of wide characters from two arrays (function)
wmemchr (C95)	finds the first occurrence of a wide character in a wide character array (function)
wmemset (C95)	copies the given wide character to every position in a wide character array (function)

Types

Defined in header <stddef.h>
Defined in header <stdlib.h>
Defined in header <wchar.h>

wchar_t	integer type that can hold any valid wide character (typedef)

Defined in header <wchar.h>
Defined in header <wctype.h>

wint_t (C95)	integer type that can hold any valid wide character and at least one more value (typedef)

Defined in header <wctype.h>

wctrans_t (C95)	scalar type that holds locale-specific character mapping

(typedef)

wctype_t (C95)	iscalar type that holds locale-specific character classification (typedef)

Macros

Defined in header <wchar.h> Defined in header <wctype.h>	
WEOF (C95)	a non-character value of type wint_t used to indicate errors (macro constant)
Defined in header <wchar.h> Defined in header <stdint.h>	
WCHAR_MIN (C95)	the smallest valid value of wchar_t (macro constant)
WCHAR_MAX (C95)	the largest valid value of wchar_t (macro constant)

References

- C11 standard (ISO/IEC 9899:2011):
  - 7.19 Common definitions <stddef.h> (p: 288)
  - 7.29 Extended multibyte and wide character utilities <wchar.h> (p: 402–446)
  - 7.30 Wide character classification and mapping utilities <wctype.h> (p: 447–454)
  - 7.31.16 Extended multibyte and wide character utilities <wchar.h> (p: 456)
  - 7.31.17 Wide character classification and mapping utilities <wctype.h> (p: 457)
  - K.3.3 Common definitions <stddef.h> (p: 585)
  - K.3.9 Extended multibyte and wide character utilities <wchar.h> (p: 627–651)
- C99 standard (ISO/IEC 9899:1999):
  - 7.17 Common definitions <stddef.h> (p: 254)
  - 7.24 Extended multibyte and wide character utilities <wchar.h> (p: 348–392)
  - 7.25 Wide character classification and mapping utilities <wctype.h> (p: 393–400)
  - 7.26.12 Extended multibyte and wide character utilities <wchar.h> (p: 402)
  - 7.26.13 Wide character classification and mapping utilities <wctype.h> (p: 402)
- C89/C90 standard (ISO/IEC 9899:1990):
  - 4.1.5 Common definitions <stddef.h>

See also

C++ documentation for **Null-terminated wide strings**