**宽字符处理函数与普通函数对照表**

1. **字符分类：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| iswalnum（） | isalnum（） | 测试字符是否为数字或字母 |
| iswalpha（） | isalpha（） | 测试字符是否是字母 |
| iswcntrl（） | iscntrl（） | 测试字符是否是控制符 |
| iswdigit（） | isdigit（） | 测试字符是否为数字 |
| iswgraph（） | isgraph（） | 测试字符是否是可见字符 |
| iswlower（） | islower（） | 测试字符是否是小写字符 |
| iswprint（） | isprint（） | 测试字符是否是可打印字符 |
| iswpunct（） | ispunct（） | 测试字符是否是标点符号 |
| iswspace（） | isspace（） | 测试字符是否是空白符号 |
| iswupper（） | isupper（） | 测试字符是否是大写字符 |
| iswxdigit（） | isxdigit（） | 测试字符是否是十六进制的数字 |

1. **大小写转换：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| towlower（） | tolower（） | 把字符转换为小写 |
| towupper（） | toupper（） | 把字符转换为大写 |

1. **字符比较：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| wcscoll（） | strcoll（） | 比较字符串 |

1. **日期和时间转换：**

|  |  |
| --- | --- |
| 宽字符函数 | 描述 |
| strftime（） | 根据指定的字符串格式和locale设置格式化日期和时间 |
| wcsftime（） | 根据指定的字符串格式和locale设置格式化日期和时间，并返回宽字符串 |
| strptime（） | 根据指定格式把字符串转换为时间值，是strftime的反过程 |

1. **打印和扫描字符串：**

|  |  |
| --- | --- |
| 宽字符函数 | 描述 |
| fprintf（）/fwprintf（） | 使用var arg参量的格式化输出 |
| fscanf（）/fwscanf（） | 格式化读入 |
| printf（） | 使用var arg参量的格式化输出到标准输出 |
| scanf（） | 从标准输入的格式化读入 |
| sprintf（）/swprintf（） | 根据var arg参量表格式化成字符串 |
| sscanf（）/swscanf（） | 以字符串作格式化读入 |
| vfprintf（）/vfwprintf（） | 使用stdarg参量表格式化输出到文件 |
| vprintf（） | 使用stdarg参量表格式化输出到标准输出 |
| vsprintf（）/vswprintf（） | 格式化stdarg参量表并写到字符串 |

1. **数字转换：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| wcstod（） | strtod（） | 把宽字符的初始部分转换为双精度浮点数 |
| wcstol（） | strtol（） | 把宽字符的初始部分转换为长整数 |
| wcstoul（） | strtoul（） | 把宽字符的初始部分转换为无符号长整数 |

1. **多字节字符和宽字符转换及操作：**

|  |  |
| --- | --- |
| 宽字符函数 | 描述 |
| mblen（） | 根据locale的设置确定字符的字节数 |
| mbstowcs（） | 把多字节字符串转换为宽字符串 |
| mbtowc（）/btowc（） | 把多字节字符转换为宽字符 |
| wcstombs（） | 把宽字符串转换为多字节字符串 |
| wctomb（）/wctob（） | 把宽字符转换为多字节字符 |

1. **输入和输出：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| fgetwc（） | fgetc（） | 从流中读入一个字符并转换为宽字符 |
| fgetws（） | fgets（） | 从流中读入一个字符串并转换为宽字符串 |
| fputwc（） | fputc（） | 把宽字符转换为多字节字符并且输出到标准输出 |
| fputws（） | fputs（） | 把宽字符串转换为多字节字符并且输出到标准输出串 |
| getwc（） | getc（） | 从标准输入中读取字符，并且转换为宽字符 |
| getwchar（） | getchar（） | 从标准输入中读取字符，并且转换为宽字符 |
| None | gets（） | 使用fgetws（） |
| putwc（） | putc（） | 把宽字符转换成多字节字符并且写到标准输出 |
| putwchar（） | putchar（） | 把宽字符转换成多字节字符并且写到标准输出 |
| None | puts（） | 使用fputws（） |
| ungetwc（） | ungetc（） | 把一个宽字符放回到输入流中 |

1. **字符串操作：**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| wcscat（） | strcat（） | 把一个字符串接到另一个字符串的尾部 |
| wcsncat（） | strncat（） | 类似于wcscat（），而且指定粘接字符串的粘接长度. |
| wcschr（） | strchr（） | 查找子字符串的第一个位置 |
| wcsrchr（） | strrchr（） | 从尾部开始查找子字符串出现的第一个位置 |
| wcspbrk（） | strpbrk（） | 从一字符串中查找另一字符串中任何一个字符第一次出现的位置 |
| wcswcs（）/wcsstr（） | strchr（） | 在一字符串中查找另一字符串第一次出现的位置 |
| wcscspn（） | strcspn（） | 返回不包含第二个字符串的的初始数目 |
| wcsspn（） | strspn（） | 返回包含第二个字符串的初始数目 |
| wcscpy（） | strcpy（） | 拷贝字符串 |
| wcsncpy（） | strncpy（） | 类似于wcscpy（），同时指定拷贝的数目 |
| wcscmp（） | strcmp（） | 比较两个宽字符串 |
| wcsncmp（） | strncmp（） | 类似于wcscmp（），还要指定比较字符字符串的数目 |
| wcslen（） | strlen（） | 获得宽字符串的数目 |
| wcstok（） | strtok（） | 根据标示符把宽字符串分解成一系列字符串 |
| wcswidth（） | None | 获得宽字符串的宽度 |
| wcwidth（） | None | 获得宽字符的宽度 |

1. **memory操作**

|  |  |  |
| --- | --- | --- |
| 宽字符函数 | 普通C函数 | 描述 |
| wmemcpy（） | memcpy（） | 复制缓冲区之间的字节 |
| wmemchr（） | memchr（） | 查找缓冲区的字符 |
| wmemcmp（） | memcmp（） | 比较两种缓冲区的字符 |
| wmemmove（） | memmove（） | 移动一缓冲区到另一个 |
| wmemset（） | memset（） | 设置缓冲区到指定的字符 |

String Function

=====================================

CharLowerWrapW

CharUpperBuffWrapW

ChrCmpI

CompareStringWrapW

GetAcceptLanguages

GetDateFormatWrapW

GetTimeFormatWrapW

IntlStrEqN

IntlStrEqNI

IntlStrEqWorker

IsCharAlphaNumericWrapW

IsCharSpace

OutputDebugStringWrapW

SHLoadIndirectString

SHStrDup

StrCat

StrCatBuff

StrCatChainW

StrChr

StrChrI

StrCmp

StrCmpC

StrCmpI

StrCmpIC

StrCmpLogicalW

StrCmpN

StrCmpNC

StrCmpNI

StrCmpNIC

StrCpy

StrCpyN

StrCSpn

StrCSpnI

StrDup

StrFormatByteSize64

StrFormatByteSizeA

StrFormatByteSizeW

StrFormatKBSize

StrFromTimeInterval

StrIsIntlEqual

StrNCat

StrPBrk

StrRChr

StrRChrI

StrRetToBSTR

StrRetToBuf

StrRetToStr

StrRetToStrN

StrRStrI

StrSpn

StrStr

StrStrI

StrToInt

StrToInt64Ex

StrToIntEx

StrTrim

wnsprintf

wvnsprintf

Path Functions

=====================================

PathAddBackslash

PathAddExtension

PathAppend

PathBuildRoot

PathCanonicalize

PathCombine

PathCommonPrefix

PathCompactPath

PathCompactPathEx

PathCreateFromUrl

PathCreateFromUrlAlloc

PathFileExists

PathFindExtension

PathFindFileName

PathFindNextComponent

PathFindOnPath

PathFindSuffixArray

PathGetArgs

PathGetCharType

PathGetDriveNumber

PathIsContentType

PathIsDirectory

PathIsDirectoryEmpty

PathIsFileSpec

PathIsHTMLFile

PathIsLFNFileSpec

PathIsNetworkPath

PathIsPrefix

PathIsRelative

PathIsRoot

PathIsSameRoot

PathIsSystemFolder

PathIsUNC

PathIsUNCServer

PathIsUNCServerShare

PathIsURL

PathMakePretty

PathMakeSystemFolder

PathMatchSpec

PathMatchSpecEx

PathParseIconLocation

PathQuoteSpaces

PathRelativePathTo

PathRemoveArgs

PathRemoveBackslash

PathRemoveBlanks

PathRemoveExtension

PathRemoveFileSpec

PathRenameExtension

PathSearchAndQualify

PathSetDlgItemPath

PathSkipRoot

PathStripPath

PathStripToRoot

PathUndecorate

PathUnExpandEnvStrings

PathUnmakeSystemFolder

PathUnquoteSpaces

SHSkipJunction

UrlApplyScheme

UrlCanonicalize

UrlCombine

UrlCompare

UrlCreateFromPath

UrlEscape

UrlEscapeSpaces

UrlFixupW

UrlGetLocation

UrlGetPart

UrlHash

UrlIs

UrlIsFileUrl

UrlIsNoHistory

UrlIsOpaque

UrlUnescape

UrlUnescapeInPlace

Registry Functions

=====================================

AssocCreate

AssocGetPerceivedType

AssocIsDangerous

AssocQueryKey

AssocQueryString

AssocQueryStringByKey

SHCopyKey

SHDeleteEmptyKey

SHDeleteKey

SHDeleteValue

SHEnumKeyEx

SHEnumValue

SHGetValue

SHOpenRegStream

SHOpenRegStream2

SHQueryInfoKey

SHQueryValueEx

SHRegCloseUSKey

SHRegCreateUSKey

SHRegDeleteEmptyUSKey

SHRegDeleteUSValue

SHRegDuplicateHKey

SHRegEnumUSKey

SHRegEnumUSValue

SHRegGetBoolUSValue

SHRegGetIntW

SHRegGetPath

SHRegGetUSValue

SHRegGetValue

SHRegOpenUSKey

SHRegQueryInfoUSKey

SHRegQueryUSValue

SHRegSetPath

SHRegSetUSValue

SHRegSetValue

SHRegWriteUSValue

SHSetValue

Color Palette Functions

=====================================

ColorAdjustLuma

ColorHLSToRGB

ColorRGBToHLS

SHCreateShellPalette

SHGetInverseCMAP

正则表达式：[L/N][P][C][W/T][CH/STR]

N：代表near，在16位系统中代表16位的段内地址。

L：代表long(far)，在16位系统中代表16位的段地址+16位的偏移地址。

在32位系统中N和L已经没有作用，只是为了向前兼容（兼容16位机器）才保留这些定义。

P：代表指针。

C：代表CONST(const)。

W：代表WCHAR(wchar\_t)。

T：代表宏\_T（一个兼容的宏，当定义了UNICODE或者\_UNICODE是表示WCHAR，否则表示char）。

CH：char的简写。

STR：string的简写。