

NAME

iswalnum_l, **iswalpha_l**, **iswcntrl_l**, **iswctype_l**, **iswdigit_l**, **iswgraph_l**, **iswlower_l**, **iswprint_l**, **iswpunct_l**, **iswspace_l**, **iswupper_l**, **iswxdigit_l**, **towlower_l**, **towupper_l**, **wctype_l**, **iswblank_l**, **iswhexnumber_l**, **iswideogram_l**, **iswnumber_l**, **iswphonogram_l**, **iswrunes_l**, **iswspecial_l**, **nextwctype_l**, **towctrans_l**, **wctrans_l** - wide character classification utilities

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <wctype.h>
```

int

```
iswalnum_l(wint_t wc, locale_t loc);
```

int

```
iswalpha_l(wint_t wc, locale_t loc);
```

int

```
iswcntrl_l(wint_t wc, locale_t loc);
```

int

```
iswctype_l(wint_t wc, locale_t loc);
```

int

```
iswdigit_l(wint_t wc, locale_t loc);
```

int

```
iswgraph_l(wint_t wc, locale_t loc);
```

int

```
iswlower_l(wint_t wc, locale_t loc);
```

int

```
iswprint_l(wint_t wc, locale_t loc);
```

int

```
iswpunct_l(wint_t wc, locale_t loc);
```

int

iswspace_l(*wint_t* wc, *locale_t* loc);

int

iswupper_l(*wint_t* wc, *locale_t* loc);

int

iswxdigit_l(*wint_t* wc, *locale_t* loc);

wint_t

towlower_l(*wint_t* wc, *locale_t* loc);

wint_t

towupper_l(*wint_t* wc, *locale_t* loc);

wctype_t

wctype_l(*wint_t* wc, *locale_t* loc);

int

iswblank_l(*wint_t* wc, *locale_t* loc);

int

iswhexnumber_l(*wint_t* wc, *locale_t* loc);

int

iswideogram_l(*wint_t* wc, *locale_t* loc);

int

iswnumber_l(*wint_t* wc, *locale_t* loc);

int

iswphonogram_l(*wint_t* wc, *locale_t* loc);

int

iswrune_l(*wint_t* wc, *locale_t* loc);

int

iswspecial_l(*wint_t* wc, *locale_t* loc);

wint_t

nextwctype_l(*wint_t* wc, *locale_t* loc);

wint_t

towctrans_l(*wint_t wc, wctrans_t, locale_t loc*);

wctrans_t

wctrans_l(*const char *, locale_t loc*);

DESCRIPTION

The above functions are character classification utility functions, for use with wide characters (*wchar_t* or *wint_t*) in the locale *loc*. They behave in the same way as the versions without the *_l* suffix, but use the specified locale rather than the global or per-thread locale. These functions may be implemented as inline functions in *<wctype.h>* and as functions in the C library. See the specific manual pages for more information.

RETURN VALUES

These functions return the same things as their non-locale versions. If the locale is invalid, their behaviors are undefined.

SEE ALSO

iswalnum(3), *iswalpha*(3), *iswblank*(3), *iswcntrl*(3), *iswctype*(3), *iswdigit*(3), *iswgraph*(3), *iswhexnumber*(3), *iswideogram*(3), *iswlower*(3), *iswnumber*(3), *iswphonogram*(3), *iswprint*(3), *iswpunct*(3), *iswrune*(3), *iswspace*(3), *iswspecial*(3), *iswupper*(3), *iswxdigit*(3), *nextwctype*(3), *towctrans*(3), *towlower*(3), *towupper*(3), *wctrans*(3), *wctype*(3)

STANDARDS

These functions conform to IEEE Std 1003.1-2008 ("POSIX.1"), except for ***iswascii_l***(), ***iswhexnumber_l***(), ***iswideogram_l***(), ***iswphonogram_l***(), ***iswrune_l***(), ***iswspecial_l***() and ***nextwctype_l***() which are FreeBSD extensions.