

**NAME**

**isdigit**, **isnumber**, **isdigit\_l**, **isnumber\_l** - decimal-digit character test

**LIBRARY**

Standard C Library (libc, -lc)

**SYNOPSIS**

```
#include <ctype.h>
```

*int*

```
isdigit(int c);
```

*int*

```
isnumber(int c);
```

*int*

```
isdigit_l(int c, locale_t loc);
```

*int*

```
isnumber_l(int c, locale_t loc);
```

**DESCRIPTION**

The **isdigit**() and **isdigit\_l**() functions test for a decimal digit character. Regardless of locale, this includes the following characters only:

```
“0”      “1”      “2”      “3”      “4”  
“5”      “6”      “7”      “8”      “9”
```

The **isnumber**() and **isnumber\_l**() functions behave similarly to **isdigit**() and **isdigit\_l**(), but may recognize additional characters, depending on the current locale setting.

The value of the argument must be representable as an *unsigned char* or the value of EOF.

The *\_l*-suffixed versions take an explicit locale argument, whereas the non-suffixed versions use the current global or per-thread locale.

**RETURN VALUES**

The **isdigit**(), **isdigit\_l**(), **isnumber**(), and **isnumber\_l**() functions return zero if the character tests false and return non-zero if the character tests true.

**COMPATIBILITY**

The 4.4BSD extension of accepting arguments outside of the range of the *unsigned char* type in locales with large character sets is considered obsolete and may not be supported in future releases. The **iswdigit()** or **iswdigit\_l()** function should be used instead.

**SEE ALSO**

`ctype(3)`, `ctype_l(3)`, `iswdigit(3)`, `iswdigit_l(3)`, `multibyte(3)`, `xlocale(3)`, `ascii(7)`

**STANDARDS**

The **isdigit()** function conforms to ISO/IEC 9899:1990 ("ISO C90"). The **isdigit\_l()** function conforms to IEEE Std 1003.1-2008 ("POSIX.1").

**HISTORY**

The **isnumber()** function appeared in 4.4BSD.