



Linux Ubuntu 26.04 Manual Pages on command 'scalbnl.3'

\$ man scalbnl.3

scalbn(3) Library Functions Manual scalbn(3)

NAME

scalbn, scalbnf, scalbnl, scalbln, scalblnf, scalblnl - scale by an integer power of radix

LIBRARY

Math library (libm, -lm)

SYNOPSIS

```
#include <math.h>

double scalbn(double x, long e);

float scalbnf(float x, long e);

long double scalbnl(long double x, long e);

double scalbn(double x, int e);

float scalbnf(float x, int e);

long double scalbnl(long double x, int e);
```

Feature Test Macro Requirements for glibc (see feature_test_macros(7)):

```
scalbn(), scalbnf(), scalbnl():

    _ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L

    || /* Since glibc 2.19: */ _DEFAULT_SOURCE

scalbn(), scalbnf(), scalbnl():

    _ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L

    || /* Since glibc 2.19: */ _DEFAULT_SOURCE

    || /* glibc <= 2.19: */ _BSD_SOURCE || _SVID_SOURCE
```

DESCRIPTION

These functions multiply their first argument *x* by FLT_RADIX (probably 2) to the power of *e*,

$$x * \text{FLT_RADIX} ** e$$

RETURN VALUE

If x is positive infinity (negative infinity), positive infinity (negative infinity) is re?
turned.

If the result overflows, a range error occurs, and the functions return HUGE_VAL, HUGE_VALF, or HUGE_VALL, respectively, with a sign the same as x.

If the result underflows, a range error occurs, and the functions return zero, with a sign the same as x .

See `math_error(7)` for information on how to determine whether an error has occurred when calling these functions.

Range error, overflow

An overflow floating-point exception (FE_OVERFLOW) is raised.

errno is set to ERANGE. An underflow floating-point exception (FE_UNDERFLOW) is raised.

For an explanation of the terms used in this section, see `attributes(7)`.

[illegible]

?	Interface	?	Attribute	?	Value	?
---	-----------	---	-----------	---	-------	---

[illegible]

? scalbn(), scalbnf(), scalbnl(), scalbln(), scalblnf(), ? Thread safety ? MT-Safe ?

? scalbnl()	?	?	?
-------------	---	---	---

[illegible]

C11, POSIX.1-2008.

Page 2/3

glibc 2.1. C99, POSIX.1-2001.

HISTORY

These functions differ from the obsolete functions described in [scalb\(3\)](#) in the type of their second argument. The functions described on this page have a second argument of an integral type, while those in [scalb\(3\)](#) have a second argument of type double.

NOTES

If `FLT_RADIX` equals 2 (which is usual), then `scalbn()` is equivalent to `ldexp(3)`.

BUGS

Before glibc 2.20, these functions did not set `errno` for range errors.

SEE ALSO

[ldexp\(3\)](#), [scalb\(3\)](#)

Linux man-pages 6.17

2026-02-08

[scalbln\(3\)](#)