Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'scalbln.3p' command

$ man scalbln.3p

SCALBLN(3P) POSIX Programmer's Manual SCALBLN(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

scalbln, scalblnf, scalblnl, scalbn, scalbnf, scalbnl ? compute expo? nent using FLT_RADIX

SYNOPSIS

```
#include <math.h>

double scalbln(double x, long n);

float scalblnf(float x, long n);

long double scalblnl(long double x, long n);

double scalbn(double x, int n);

float scalbnf(float x, int n);

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

DESCRIPTION

The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1?2017 defers to the ISO C standard.

These functions shall compute x * FLT_RADIXn efficiently, not normally

by computing FLT_RADIXn explicitly.

An application wishing to check for error situations should set errno to zero and call feclearexcept(FE_ALL_EXCEPT) before calling these functions. On return, if errno is non-zero or fetestexcept(FE_INVALID | FE_DIVBYZERO | FE_OVERFLOW | FE_UNDERFLOW) is non-zero, an error has occurred.

RETURN VALUE

Upon successful completion, these functions shall return x * FLT_RADIXn.

If the result would cause overflow, a range error shall occur and these functions shall return ?HUGE_VAL, ?HUGE_VALF, and ?HUGE_VALL (according to the sign of x) as appropriate for the return type of the function.

If the correct value would cause underflow, and is not representable, a range error may occur, and scalbln(), scalblnf(), scalblnl(), scalbn(), scalbnf(), and scalbnl() shall return 0.0, or (if IEC 60559 Floating-Point is not supported) an implementation-defined value no greater in magnitude than DBL_MIN, FLT_MIN, LDBL_MIN, DBL_MIN, FLT_MIN, and LDBL_MIN, respectively.

If x is NaN, a NaN shall be returned.

If x is ?0 or ?Inf, x shall be returned.

If n is 0, x shall be returned.

If the correct value would cause underflow, and is representable, a range error may occur and the correct value shall be returned.

ERRORS

These functions shall fail if:

Range Error The result overflows.

If the integer expression (math_errhandling & MATH_ERRNO) is non-zero, then errno shall be set to [ERANGE]. If the integer expression (math_errhandling & MATH_ERREXCEPT) is non-zero, then the overflow floating-point exception shall be raised.

These functions may fail if:

Range Error The result underflows.

If the integer expression (math_errhandling & MATH_ERRNO) is non-zero, then errno shall be set to [ERANGE]. If the integer expression (math_errhandling & MATH_ERREXCEPT) is non-zero, then the underflow floating-point exception shall be raised.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

On error, the expressions (math_errhandling & MATH_ERRNO) and (math_errhandling & MATH_ERREXCEPT) are independent of each other, but at least one of them must be non-zero.

RATIONALE

These functions are named so as to avoid conflicting with the historical definition of the scalb() function from the Single UNIX Specification. The difference is that the scalb() function has a second argument of double instead of int. The scalb() function is not part of the ISO C standard. The three functions whose second type is long are provided because the factor required to scale from the smallest positive floating-point value to the largest finite one, on many implementations, is too large to represent in the minimum-width int format.

FUTURE DIRECTIONS

None.

SEE ALSO

feclearexcept(), fetestexcept()

The Base Definitions volume of POSIX.1?2017, Section 4.20, Treatment of Error Conditions for Mathematical Functions, <math.h>

COPYRIGHT

event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at http://www.opengroup.org/unix/online.html .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.ker?nel.org/doc/man-pages/reporting_bugs.html .