



Rocky Enterprise Linux 9.2 Manual Pages on command 'sinh.3'

C:\>man sinh.3

SINH(3) Linux Programmer's Manual SINH(3)

NAME

sinh, sinhf, sinhl - hyperbolic sine function

SYNOPSIS

```
#include <math.h>
```

```
double sinh(double x);
```

```
float sinhf(float x);
```

```
long double sinhl(long double x);
```

Link with -lm.

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

sinhf(), sinhl():

```
  _ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L
```

```
  || /* Since glibc 2.19: */ _DEFAULT_SOURCE
```

```
  || /* Glibc versions <= 2.19: */ _BSD_SOURCE || _SVID_SOURCE
```

DESCRIPTION

These functions return the hyperbolic sine of x, which is defined mathematically

as:

$$\sinh(x) = (\exp(x) - \exp(-x)) / 2$$

RETURN VALUE

On success, these functions return the hyperbolic sine of x.

If x is a NaN, a NaN is returned.

If x is +0 (-0), +0 (-0) is returned.

If x is positive infinity (negative infinity), positive infinity (negative infinity) is returned.

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

ERRORS

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

The following errors can occur:

Range error: result overflow

`errno` is set to `ERANGE`. An overflow floating-point exception (`FE_OVERFLOW`) is raised.

ATTRIBUTES

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

??

?Interface ? Attribute ? Value ?

??

?sinh(), sinhf(), sinhl() ? Thread safety ? MT-Safe ?

??

CONFORMING TO

C99, POSIX.1-2001, POSIX.1-2008.

The variant returning double also conforms to SVr4, 4.3BSD, C89.

SEE ALSO

`acosh(3)`, `asinh(3)`, `atanh(3)`, `cosh(3)`, `csinh(3)`, `tanh(3)`

COLOPHON

This page is part of release 5.05 of the Linux man-pages project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.