

**NAME**

atanh, atanhf, atanhL – inverse hyperbolic tangent function

**SYNOPSIS**

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

```
double atanh(double x);
```

```
float atanhf(float x);
```

```
long double atanhL(long double x);
```

Link with `-lm`.

Feature Test Macro Requirements for glibc (see [feature\\_test\\_macros\(7\)](#)):

**atanh()**:

```
_ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L || _XOPEN_SOURCE >= 500 /* Since
glibc 2.19: */ _DEFAULT_SOURCE || /* Glibc versions <= 2.19: */ _BSD_SOURCE ||
_SVID_SOURCE
```

**atanhf()**, **atanhL()**:

```
_ISOC99_SOURCE || _POSIX_C_SOURCE >= 200112L /* Since glibc 2.19: */ _DE-
FAULT_SOURCE || /* Glibc versions <= 2.19: */ _BSD_SOURCE || _SVID_SOURCE
```

**DESCRIPTION**

These functions calculate the inverse hyperbolic tangent of  $x$ ; that is the value whose hyperbolic tangent is  $x$ .

**RETURN VALUE**

On success, these functions return the inverse hyperbolic tangent of  $x$ .

If  $x$  is a NaN, a NaN is returned.

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

If  $x$  is  $+1$  or  $-1$ , a pole error occurs, and the functions return **HUGE\_VAL**, **HUGE\_VALF**, or **HUGE\_VALL**, respectively, with the mathematically correct sign.

If the absolute value of  $x$  is greater than 1, a domain error occurs, and a NaN is returned.

**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:

Domain error:  $x$  less than  $-1$  or greater than  $+1$

`errno` is set to **EDOM**. An invalid floating-point exception (**FE\_INVALID**) is raised.

Pole error:  $x$  is  $+1$  or  $-1$

`errno` is set to **ERANGE** (but see **BUGS**). A divide-by-zero floating-point exception (**FE\_DIVBYZERO**) is raised.

**ATTRIBUTES**

For an explanation of the terms used in this section, see [attributes\(7\)](#).

Interface	Attribute	Value
<b>atanh()</b> , <b>atanhf()</b> , <b>atanhL()</b>	Thread safety	MT-Safe

**CONFORMING TO**

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

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

**BUGS**

In glibc 2.9 and earlier, when a pole error occurs, `errno` as set to **EDOM** instead of the POSIX-mandated **ERANGE**. Since version 2.10, glibc does the right thing.

**SEE ALSO**

[acosh\(3\)](#), [asinh\(3\)](#), [catanh\(3\)](#), [cosh\(3\)](#), [sinh\(3\)](#), [tanh\(3\)](#)

**COLOPHON**

This page is part of release 4.16 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/>.