

**NAME**

log1p, log1pf, log1pl – logarithm of 1 plus argument

**SYNOPSIS**

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

```
double log1p(double x);
```

```
float log1pf(float x);
```

```
long double log1pl(long double x);
```

Link with `-lm`.

Feature Test Macro Requirements for glibc (see **feature\_test\_macros(7)**):

**log1p()**:

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

```
|| _XOPEN_SOURCE >= 500
```

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

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

**log1pf(), log1pl()**:

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

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

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

**DESCRIPTION**

These functions return a value equivalent to

$$\log(1 + x)$$

The result is computed in a way that is accurate even if the value of  $x$  is near zero.

**RETURN VALUE**

On success, these functions return the natural logarithm of  $(1 + x)$ .

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

If  $x$  is positive infinity, positive infinity is returned.

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

If  $x$  is less than  $-1$  (including negative infinity), a domain error occurs, and a NaN (not a number) 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:  $x$  is less than  $-1$

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

Pole error:  $x$  is  $-1$

`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>log1p(), log1pf(), log1pl()</b>	Thread safety	MT-Safe

**CONFORMING TO**

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

**BUGS**

Before version 2.22, the glibc implementation did not set *errno* to **EDOM** when a domain error occurred.

Before version 2.22, the glibc implementation did not set *errno* to **ERANGE** when a range error occurred.

**SEE ALSO**

**exp(3)**, **expm1(3)**, **log(3)**

**COLOPHON**

This page is part of release 5.02 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/>.