



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

\$ man fpclassify.3

fpclassify(3) Library Functions Manual fpclassify(3)

NAME

fpclassify, isfinite, isnormal, isnan, isinf - floating-point classification macros

LIBRARY

Math library (libm, -lm)

SYNOPSIS

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

```
int fpclassify(x);
```

```
int isfinite(x);
```

```
int isnormal(x);
```

```
int isnan(x);
```

```
int isinf(x);
```

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

fpclassify(), isfinite(), isnormal():

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

isnan():

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

```
|| _XOPEN_SOURCE
```

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

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

isinf():

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

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

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

DESCRIPTION

Floating point numbers can have special values, such as infinite or NaN. With the macro `fpclassify(x)` you can find out what type `x` is. The macro takes any floating-point expression as argument. The result is one of the following values:

`FP_NAN` `x` is "Not a Number".

`FP_INFINITE` `x` is either positive infinity or negative infinity.

`FP_ZERO` `x` is zero.

`FP_SUBNORMAL` `x` is too small to be represented in normalized format.

`FP_NORMAL` if nothing of the above is correct then it must be a normal floating-point number.

The other macros provide a short answer to some standard questions.

`isfinite(x)` returns a nonzero value if

`(fpclassify(x) != FP_NAN && fpclassify(x) != FP_INFINITE)`

`isnormal(x)` returns a nonzero value if `(fpclassify(x) == FP_NORMAL)`

`isnan(x)` returns a nonzero value if `(fpclassify(x) == FP_NAN)`

`isinf(x)` returns 1 if `x` is positive infinity, and -1 if `x` is negative infinity.

ATTRIBUTES

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

??

? Interface ? Attribute ? Value ?

??

? `fpclassify()`, `isfinite()`, `isnormal()`, `isnan()`, `isinf()` ? Thread safety ? MT-Safe ?

??

STANDARDS

C11, POSIX.1-2008.

HISTORY

POSIX.1-2001, C99.

In `glibc 2.01` and earlier, `isinf()` returns a nonzero value (actually: 1) if `x` is positive infinity or negative infinity. (This is all that C99 requires.)

NOTES

For `isinf()`, the standards merely say that the return value is nonzero if and only if the argument has an infinite value.

SEE ALSO

[finite\(3\)](#), [INFINITY\(3\)](#), [isgreater\(3\)](#), [signbit\(3\)](#)

Linux man-pages 6.17

2026-02-08

[fpclassify\(3\)](#)