Math Library Manual

Math Library Manual

Version X.Y

Table of Contents

Chapter 1. Concepts

Chapter 2. Reference

math reference

Exponential and logarithmic functions

exp

```
double exp(double x);
```

Exponential function of X.

frexp

```
double frexp(double x, int* e);
```

Break VALUE into a normalized fraction and an integral power of 2.

ldexp

```
double ldexp(double x, int e);
```

X times (two to the EXP power).

log

```
double log(double x);
```

Natural logarithm of X.

log10

```
double log10(double x);
```

Base-ten logarithm of X.

modf

```
double modf(double x, double* iptr);
```

Break VALUE into integral and fractional parts.

exp10

```
double exp10(double x);
```

A function missing in all standards: compute exponent to base ten.

pow10

```
double pow10(double x);
```

Another name occasionally used.

Hyperbolic functions

cosh

```
double cosh(double x);
```

Hyperbolic cosine of X.

sinh

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

Hyperbolic sine of X.

tanh

```
double tanh(double x);
```

Hyperbolic tangent of X.

acosh

```
double acosh(double x);
```

Hyperbolic arc cosine of X.

asinh

```
double asinh(double x);
```

Hyperbolic arc sine of X.

atanh

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

Hyperbolic arc tangent of X.

LIB_VERSION

```
LIB_VERSION_TYPE LIB_VERSION;
```

This variable can be changed at run-time to any of the values above to affect floating point error handling behavior (it may also be necessary to change the hardware FPU exception settings).

LIB_VERSION_TYPE

```
enum LIB_VERSION_TYPE { IEEE=- 1, SVID, XOPEN, POSIX, ISOC};
```

Nearest integer, absolute value, and remainder functions

ceil

```
double ceil(double x);
```

Smallest integral value not less than X.

fabs

```
double fabs(double x);
```

Absolute value of X.

floor

```
double floor(double x);
```

Largest integer not greater than X.

fmod

```
double fmod(double x, double y);
```

Floating-point modulo remainder of X/Y.

isinf

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

Return 0 if VALUE is finite or NaN, +1 if it is +Infinity, -1 if it is -Infinity.

isfinite

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

Return nonzero if VALUE is finite and not NaN.

isnan

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

Return nonzero if VALUE is not a number.

Power functions

pow

```
double pow(double x, double y);
```

Return X to the Y power.

sqrt

```
double sqrt(double x);
```

Return the square root of X.

hypot

```
double hypot(double x, double y);
```

Return `sqrt(X*X + Y*Y)'.

cbrt

```
double cbrt(double x);
```

Return the cube root of X.

Trigonometric functions

acos

```
double acos(double x);
```

Arc cosine of X.

asin

```
double asin(double x);
```

Arc sine of X.

atan

```
double atan(double x);
```

Arc tangent of X.

atan2

```
double atan2(double y, double x);
```

Arc tangent of Y/X.

cos

```
double cos(double x);
```

Cosine of X.

sin

```
double sin(double x);
```

Sine of X.

tan

```
double tan(double x);
```

Tangent of X.

struct exception

type

```
int type;
```

name

```
 char * name;
```

arg1

```
double arg1;
```

arg2

```
double arg2;
```

retval

```
double retval;
```

math_h_

```
math_h_
```

matherr

```
int matherr(exception* exc);
```

Types

Functions