### сеТребования к функциям sqrt() в стандарте POSIX (IEEE 1003.1-2013)

**NAME**

sqrt, sqrtf, sqrtl - square root function

**SYNOPSIS**

#include <[math.h](http://pubs.opengroup.org/onlinepubs/9699919799/basedefs/math.h.html)>  
  
double sqrt(double *x*);  
float sqrtf(float *x*);  
long double sqrtl(long double *x*);

**DESCRIPTION**

[[CX](javascript:open_code('CX'))] The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1-2008 defers to the ISO C standard. 

These functions shall compute the square root of their argument *x*, 

An application wishing to check for error situations should set *errno* to zero and call *feclearexcept*(FE\_ALL\_EXCEPT) before calling these functions. On return, if *errno* is non-zero or *fetestexcept*(FE\_INVALID | FE\_DIVBYZERO | FE\_OVERFLOW | FE\_UNDERFLOW) is non-zero, an error has occurred.

**RETURN VALUE**

Upon successful completion, these functions shall return the square root of *x*.

For finite values of *x* < -0, a domain error shall occur, and [[MX](javascript:open_code('MX'))]  either a NaN (if supported), or  an implementation-defined value shall be returned.

[[MX](javascript:open_code('MX'))] If *x* is NaN, a NaN shall be returned.

If *x* is ±0 or +Inf, *x* shall be returned.

If *x* is -Inf, a domain error shall occur, and a NaN shall be returned. 

**ERRORS**

These functions shall fail if:

Domain Error

The finite value of *x* is < -0, [[MX](javascript:open_code('MX'))]  or *x* is -Inf. 

If the integer expression (*math\_errhandling* & MATH\_ERRNO) is non-zero, then *errno* shall be set to [EDOM]. If the integer expression (*math\_errhandling* & MATH\_ERREXCEPT) is non-zero, then the invalid floating-point exception shall be raised.

### Требования в стандарте IEEE 754-2008

The operation **squareRoot**(*x*) computes √ *x.* It has a positive sign for all operands ≥ 0, except that **squareRoot**(−0) shall be −0.

The preferred exponent is floor(Q(*x*) / 2).

Operations on infinite operands are usually exact and therefore signal no exceptions, including, among others,

― squareRoot(+∞)

Except that squareRoot(−0) shall be −0, every numeric squareRoot result shall have a positive sign.

For operations producing results in floating-point format, the default result of an operation that signals the invalid operation exception shall be a quiet NaN that should provide some diagnostic information (see 6.2). These operations are:

g) squareRoot if the operand is less than zero

### Изложение требований IEEE 754 2008 для sqrt()

Двоичные числа с плавающей точкой представляются в виде массивов бит длиной n, которые делятся на три части: один знаковый бит S, порядок E из k бит, мантисса M из (n-k-1) бит. Число B = 2k-1-1 называется смещением порядка.

Представляемое число при этом вычисляется по следующим правилам

если E  0 и E  2k-1 (порядок не состоит из одних нулей или одних единиц)   
x = (-1)S·2(E-B)·(1+M/2n-k-1)  
это нормализованные числа

если E = 0  
x = (-1)S·2(-B+1)·(M/2n-k-1)  
это денормализованные числа

если E = 2k-1  
при M = 0, x = (-1)S· (используются для представления бесконечных или слишком больших по абсолютной величине результатов)  
при M  0, x = NaN (не-число, используется для представления результатов, которым нельзя согласованно с остальными правилами приписать конечное или бесконечное значение). Различают сигнальное NaN и тихое NaN — в сигнальном NaN первый mbn мантиссы равен 1, в тихом — 0.

Для двоичных чисел с плавающей точкой стандарт требует поддерживать три типа

* binary32 (float): n = 32, k = 8
* binary64 (double): n = 64, k = 11
* binary128 (quadruple): n = 128, k = 15

Результат любой поддерживаемой стандартом операции (в т.ч. и sqrt) должен быть корректно округленным к одному из представимых в рамках заданного типа чисел с плавающей точкой точным математическим результатом. При этом должно поддерживаться четыре режима округления.

* К ближайшему (режим по умолчанию) — результат округляется к ближайшему представимому числу.  
  Если таких чисел два, выбирается то, которое имеет бит 0 в конце мантиссы (округление к четному).  
  Если точный результат отличается от максимального/минимального представимого числа меньше, чем на половину величины последнего бита мантиссы (т.е., меньше, чем на 2(2k-B-2)·1/2n-k-1·1/2), то он округляется к максимальному/минимальному представимому числу, иначе, к +/-.
* Вверх — результат округляется к ближайшему сверху представимому числу, или к +, если такого нет.
* Вниз — результат окруляется к ближайшему снизу представимому числу, или к -. если такого нет.
* К нулю — для положительных результатов применяется округление вниз, для отрицательных — вверх.

Результатом вычисления sqrt с аргументом NaN должно быть NaN (тихое или сигнальное — в соответствии с аргументом).

Результатом вычисления sqrt с аргументом + должна быть +.

Результатом вычисления sqrt с отрицательным аргументом (конечным или бесконечным, но не -0) должно быть сигнальное NaN.

Результатом вычисления sqrt с аргументом -0 должен быть -0.

Кроме того, по результатам вычислений должны выставляться следующие флаги.

* INVALID, в том случае, если один из аргументов яляется сигнальным NaN или все аргументы не NaN, а в результате получается NaN (для sqrt — в случае отрицательного аргумента, кроме -0).
* DIVIDE-BY-ZERO, в том случае, результат в точности бесконечен (для sqrt нет таких ситуаций).
* OVERFLOW, в том случае, если получаемый результат конечен, но превосходит по абсолютной величине наибольшее представимое в заданном типе число (для sqrt нет таких ситуаций).
* UNDERFLOW, в том случае, если получаемый результат не 0 и по абсолютной величине меньше наименьшего нормализованного числа (для sqrt нет таких ситуаций).
* INEXACT, в том случае, если получаемый результат меняется при округлении (т.е., округленный результат не равен точному математическому).