

## Re: sin(x) for large x

**Source:** <http://coding.derkeiler.com/Archive/Fortran/comp.lang.fortran/2004-05/0618.html>

---

**From:** Dr Chaos ([mbkennelSPAMBEGONE\\_at\\_NOSPAMyahoo.com](mailto:mbkennelSPAMBEGONE_at_NOSPAMyahoo.com))

**Date:** 05/14/04

Date: Fri, 14 May 2004 21:47:38 +0000 (UTC)

glen herrmannsfeldt <[gah@ugcs.caltech.edu](mailto:gah@ugcs.caltech.edu)> wrote:

- > *There is a discussion in comp.lang.c titled*
- > *Sine code for ANSI C*
- >
- > *regarding the evaluation of the sin() function for large*
- > *arguments. It might be more applicable in this*
- > *group, so I thought I would ask here.*
- >
- > *What result do people expect from the sin() function for*
- > *very large arguments, for example sin(1e100)?*

sin(1e100) is a symptom of a mistake.

I believe that if one's program depends on evaluating that to any degree of accuracy, it is computing something in a mathematically mistaken way.

Generally you need some paper-and-pencil analysis and thinking to avoid this problem.

- > *Hopefully the discussion won't get too nasty, but it*
- > *does seem to be a contentious issue over there.*

Sometimes computer scientists have an odd idea what it means to be "correct", e.g. imagining that the particular rules of IEEE floating point define "correctness" as opposed to solving the problem in a satisfactory way.

Nevertheless, in a legalistic fashion, one could define it thus:  
 $\sin(1.0D100) = \sin(x)$  where  $x$  is the closest IEEE representation, in double precision floating point, to the number  $10^{100}$ .

Now, every IEEE floating point number is a rational number mathematically. That has an exact value of sin, and you can define sin(x) as the IEEE floating point number which is closest to that exact number.

In single precision, 1e100 doesn't have a representation I believe.

comp.lang.fortran: Re: sin(x) for large x

Now this definition is logical, but it may make implementation far too difficult, and waste computational time on the cases that almost everybody cares about, i.e. small arguments.

> -- *glen*