

Re: demonic problem descriptions

Source: <http://coding.derkeiler.com/Archive/Lisp/comp.lang.lisp/2005-02/0480.html>

From: Cameron MacKinnon (cmackin+nn_at_clearspot.net)

Date: 02/06/05

Date: Sun, 06 Feb 2005 15:52:42 -0500

Kent M Pitman wrote:

> *Floating point is very close to [being fundamentally embedded] in the
> computer world, given that there is floating point hardware on nearly
> every machine, and decades of history involving thousands, perhaps
> even millions, of programmers who are familiar with the "meaning" of
> 0.11111 in computerese.*

I suspect that of all the people who have ever written floating point code (and here I'll be generous and exclude mere spreadsheet users), fewer than half of them even realized that the representation was inexact, and perhaps one in fifty understood enough to be wary of the pitfalls.

There was perhaps a time when the majority of users of general purpose FP hardware wanted inexact math without error bounds calculation by default -- i.e they were willing to live with the occasional need for numerical analysis and the odd model blowing up because they couldn't afford the cycles or the gates to compute cumulative errors along with their shady numbers. That time is long past, I think.

> *What you ARE obliged to do is to write code that is not there by
> default. Lisp enables you to do many things, but one of the things
> you are not enabled to do is to wish for things and have them come
> into existence without any work.*

True... That's what c.l.l is for! The original poster got helpful code and suggestions, but also, disturbingly, comments to the effect that yes, Lisp is a programmable programming language, but you shouldn't program it THAT way.