

Re: something to chat about, lisp and Mathematica for list processing.

Re: something to chat about, lisp and Mathematica for list processing.

Source: <http://sci.tech-archive.net/Archive/sci.math.symbolic/2007-06/msg00095.html>

- *From:* rjf <fateman@xxxxxxxx>
 - *Date:* Wed, 20 Jun 2007 22:19:50 -0700
-

On Jun 18, 9:49 am, Jon Harrop <j...@xxxxxxxxxxxxxxxxxxxx> wrote:

All good points but...

rjf wrote:

Manipulating Mathematica programs is presumably possible, but I do not know of anyone doing so.

If Lisp macros count then Mathematica replacement rules should count. These are ubiquitous, of course.

--

Dr Jon D Harrop, Flying Frog Consultancy

OCaml for Scientists http://www.ffconsultancy.com/products/ocaml_for_scientists/?usenet

No, I don't see that mathematica rules are similar to macros in lisp. Mathematica rules are part of a rewriting system at the center of mathematica, kind of a read-applyrules-print loop. In lisp there is a read-eval-print loop.

The eval step sometimes requires "expand macros, then eval".

A macro system as usually used in Lisp is via a function that defines functions. A macro system for mathematica would, I suppose, enable a programmer to do something like produce and install in the system a whole bunch of distinct rules by using a single rule repeatedly with different arguments.

I agree that mathematica rules are ubiquitous. they are usually confused by the casual user with programs, e.g. $f[x_]:=x+1$ seems to act like $(\text{defun } f(x)(+ x 1))$ in lisp. But the mathematica notion is rather different in general. but this specific example works almost the same.

A rule-driven system written in Lisp is easy to define, and is done in any number of books on Lisp, e.g. where prolog is defined in Lisp.

Re: something to chat about, lisp and Mathematica for list processing.

Re: something to chat about, lisp and Mathematica for list processing.

Macros are, however, something else. And they are not like rules in mathematica.

RJF

.