

# Re: Steps towards writing a computer algebra system

**Source:** <http://sci.tech-archive.net/Archive/sci.math.symbolic/2005-01/0132.html>

---

**From:** Richard J. Fateman (*fateman\_at\_eecs.berkeley.edu*)

**Date:** 01/18/05

Date: Tue, 18 Jan 2005 11:55:31 -0800

These books would definitely not tell you how to write a computer algebra system.

They have some interesting material, but a careful reading of some of these will give you reliably bad advice on what makes sense to include in an actual program.

In fact it might make an interesting chart to fill out

recommendations implemented by  
as "best" algorithm Macsyma/Reduce/Mma/Maple/NTL/GiNAC/.

Then we would have to admit that most of the "optimal" algorithms are not used by some of the most intensively developed systems.

Because they are not useful enough (slow on practical sizes, not general enough) and sometimes difficult to write out in full detail and debug.

RJF

Klueless wrote:

> "Mike" <mike\_in\_england2000@yahoo.co.uk> wrote in message  
news:f151b240.0501180835.6a31093c@posting.google.com...

>

>>I am rather lost on where to start.

>

>

> *Computer Algebra, Systems and Algorithms for Algebraic*

> *Computation* by J. H. Davenport, Y. Siret, E. Tournier

>

> *Modern Computer Algebra* by Joachim von zur Gathen and

> *Jurgen Gerhard*

>

> *Algorithms for Computer Algebra* by Keith O. Geddes,

> *Stephen R. Czapor, George Labahn*

>

sci.math.symbolic: Re: Steps towards writing a computer algebra system

- > *Computer Algebra Symbolic and Algebraic Computation*
- > *edited by B. Buchberger, G. E. Collins, and R. Loos in*
- > *cooperation with R. Albrecht*
- >
- >
- >