Computer Algebra Systems

A Practical Guide

Edited by

Michael J. Wester
Cotopaxi, Albuquerque, New Mexico, USA

JOHN WILEY & SONS, LTD
Chichester · New York · Weinheim · Brisbane · Singapore · Toronto

Table of Systems per OS

	DOS	Mac OS	NeXT Step	Open VMS	OS/2	Unix	Win 3.1	Win 95	Win NT
Axiom						•		•	•
Bergman	•					•			
BigNum	•			•		•			
CoCoA	•	•				•	•	•	•
Derive	•					•	•		•
Dist. Maple		•				•		•	•
FELIX	•					•	•		•
Fermat		•						•	•
FORM	•	•	•	•		•			•
GANITH						•			
GAP	•	•				•		•	•
GNU MP	•					•			
GRB	•					•			
JACAL	•	•		•	•	•			
Kan						•			
KASH/KANT	•				•	•	•	•	
LiDIA		•			•	•			•
LiE						•			
Macaulay						•	•	•	•
Macsyma						•		•	•
Magma	•					•			
Magnus						•			
Maple		•				•		•	•
MAS			•		•	•			
MathCAD		•					•	•	•
Mathematica		•	•	•	•	•		•	•
MathView		•					•	•	•
Maxima						•			
MuPAD		•				•		•	•
NTL						•		•	•
Numbers	•								
PARI/GP	•	•				•			
REDUCE	•	•				•	•	•	•
Risa/Asir	•	•				•			
SACLIB						•			
Schur	•					•			
Scilab						•		•	•
SIMATH						•			
SINGULAR		•				•		•	•
UBASIC	•								•
Zen					•	•			

Appendix A

Major General Purpose CASs

Here, some historical information on the major general purpose computer algebra systems currently in use and their direct historical antecedents is presented. A general purpose system is one that attempts to cover a wide spectrum of mathematical concepts and techniques, as opposed to a special purpose system which concentrates on a particular specialized area such as number theory or tensor analysis. All the major general purpose systems (and most special purpose systems as well) are extensible, providing a user-level programming language with which features not originally anticipated by the designers can be added in a systematic way.

Initially, the first CASs were specialized. The first documented computer programs to do actual symbolic computation (in particular, analytic differentiation) were described in 1953 by H. G. Kahrimanian at Temple University [Kahrimanian53] who wrote an assembly program on the Univac I, and J. F. Nolan of the Massachusetts Institute of Technology [Nolan53] whose program ran on the Whirlwind I ([Caviness93]§1.1), [van Hulzen83].

General purpose systems got their start in the early 1960s, developing both as outgrowths of special purpose systems (like REDUCE, which started out as a tool for elementary particle physics computations [Hearn66]) and as systems designed from the start to be general purpose calculators (like Macsyma, which was the end result of a series of systems developed at M.I.T.: Carl Engleman's MATHLAB [Engelman65] and MATHLAB-68 [Engelman69], William Martin's Symbolic Mathematical Laboratory [Martin67], and Joel Moses' SIN (Symbolic INtegrator) [Moses67]). Since those days, a great plethora of symbolic computation programs has come and gone.

Generally, CASs are software packages, but special purpose hardware has been designed in one instance to support a symbolic computation language: ANALITIK [Husberg74] and its successors, which ran on the Soviet MIR-2 computers and their successors.

I would like to thank Ellen and Jeffrey P. Golden (Macsyma), Tony Hearn (REDUCE), Richard Jenks (Axiom and Scratchpad), Frank Postel (MuPAD), Mohamed Omar Rayes (TI-92), Albert Rich (muMATH and Derive), David Stoutemyer (TI-92) and Michael Trott (Mathematica) for providing details for this short historical survey. See also Peter Larcombe's chapter in this volume and in addition, [van Hulzen83] and ([Geddes92]§1.3) for many more details on the history of computer algebra systems, [Grosheva98] for information on the history of CASs in the former Soviet Union/Russia, and [Caviness86] for a history of computer algebra algorithms.

System	Released[1]	Language	Principal Original Author(s)	Locale
REDUCE	1968	Lisp	Anthony C. Hearn	Stanford University Stanford, California, USA
Macsyma	1970	Lisp	Joel Moses, William Martin	Massachusetts Institute of Technology Cambridge, Massachusetts, USA
Scratchpad	1971	Lisp	James Griesmer, Richard Jenks	IBM Thomas J. Watson Research Center Yorktown Heights, New York, USA
muMATH[2]	1979	muSIMP[3]	David Stoutemyer, Albert Rich	The Soft Warehouse Honolulu, Hawaii, USA
SMP	1982	C	Chris A. Cole, Stephen Wolfram	California Institute of Technology Pasadena, California, USA
Maple	1983	C[4]	Gaston Gonnet, Keith Geddes	University of Waterloo Waterloo, Ontario, Canada
Derive[5]	1988	Lisp	David Stoutemyer, Albert Rich	Soft Warehouse, Inc. Honolulu, Hawaii, USA
Mathematica[6]	1988	C	Stephen Wolfram	pre-Wolfram Research, Inc. Urbana, Illinois, USA
MuPAD	1990	C, C++	Benno Fuchssteiner et al.	University of Paderborn Paderborn, Germany
Axiom[7]	1991	Lisp	Richard Jenks, Barry Trager et al.	IBM Thomas J. Watson Research Center Yorktown Heights, New York, USA
TI-92[8]	1995	C	David Stoutemyer, TI internal team[9]	Texas Instruments Dallas, Texas, USA[10]

[1] First public release.
[2] First small computer CAS.
[3] micro Symbolic IMPlementation language (a parser written in muLISP).
[4] The kernel is actually written in macros that can be translated into C via the Margay macro processor, originally written by the University of Waterloo.
[5] Successor of muMATH.
[6] Successor of SMP.
[7] Successor of ScratchPad, originally called Scratchpad II.
[8] First calculator CAS. A later model, the TI-89, is a physically smaller vertical-format version of the TI-92 plus TI-92+ (the TI-92+ is a module that enhances the CA and other capabilities of the TI-92). Both the TI-92+ module and the TI-89 were released in 1998.
[9] These are the principal authors of the CA portion of the software.
[10] Development was also done at Soft Warehouse, Inc., Honolulu, Hawaii, USA.

Origins of Names

Axiom	Like an *axiom* of mathematics, the basis for elaborate and comprehensive mathematical developments. This name was proposed by Stephen Hague of The Numerical Algorithms Group Limited as an alternative to Scratchpad II, which was what the system was originally called.
Derive	A name that suggests the dynamic, creative process of doing mathematics on a computer (and not a noun that begins with the letter M, as there were already quite a few such named systems at the time!).
Macsyma	Project MAC SYmbolic MAnipulator. Project MAC's first task was writing a time-sharing system (CTSS—the Compatible Time-Sharing System[11]), so MAC originally stood for Multi-Access Computer. Later on, MAC also represented Machine-Aided Cognition, while Maniacs And Clowns, Minsky Against Corbató, and Moses And Company were other (humorous) interpretations.[12] Consistent with the puns on MAC, Macsyma (makseema) also is the intensive form in Hebrew for "makes magic" (from *mem* [making] + *k'syma* [magic—this is the same root Semitic word that produces *kismet*: fate or destiny]) as well as "wonderful or fascinating".
Maple	Indicates a Canadian origin (for example, the Canadian flag displays a *maple* leaf).
Mathematica	A latinized version of the word "mathematics". This name was suggested by Steven Jobs to Stephen Wolfram, who had independently thought of it. Since both had the same idea, the name was adopted.
muMATH	The Greek letter mu (μ) represents "micro" in the metric system of units—so a math system that runs on the microprocessors used in small computers.
MuPAD	Multi-Processing Algebra Data tool. Also indicates its place of origin (PADerborn).
REDUCE	Computer algebra systems tend to produce very large expressions for many problems—so a system that will (hopefully) *reduce* results to a more manageable form.
Scratchpad	A computer *scratchpad* for jotting down mathematical ideas. The name was suggested by James Griesmer.
SMP	Symbolic Manipulation Program.
TI-92	Texas Instruments model 92 calculator. The number was chosen based on customer feedback.

[11] The first operating system Macsyma ran on was ITS—the Incompatible Time-sharing System, named partially in a punning reaction to CTSS.

[12] Other notable software that came out of Project MAC was Maclisp, a forerunner to Common Lisp, and the visual editor emacs (originally, Editor MACroS).