Workshop on

Computer Algebra System (CAS)

Session: Introduction and Algebra with CAS

10. 04. 2016

P. K. Yadav & K. Taneja

khusboo.taneja@sharda.ac.in

Department of Civil Engineering
Dept. of Computer Sci. & Engg.

SHARDA UNIVERSITY

Beyond Boundaries

Motivation

Let us start with an Algebra problem:

$$x^2 + 2x - 1$$

Can you factorize this? Of course! readily, and it is $(x+1)^2$. How about this:

$$2x^4 - 5x^3 - 19x^2 + 11x - 1$$

Not readily, but you may find it to be $(x^2 + 2x - 1)(2x^2 - 9x + 1)$

Operation such as factorize, expand, simplify and many other follow a set of non-numerical manipulations before numerical evaluation.

The CAS provides algorithm for non-numerical or **symbolical** manipulations, which can then be numerically evaluated.

So what is CAS?

Computer Algebra, also called Symbolic Computation is a scientific area that refers to the study and development of algorithms and software for manipulating **mathematical expressions** and other **mathematical objects** (from Wikipedia).

Therefore, Software applications that perform symbolic calculations are called computer algebra systems.

Essentially, the CAS require:

i Representation of Mathematical Data

ii A programming Language

iii A Dedicated Memory Manager

iv User interface

v Optionally, a visualization tool

So what is CAS?

Where are CAS?

The earliest work of CAS was by the Physics Nobel Laureate Dr. Martin Veltman in 1963. He was involved with High Energy Physics.

Carl Engelman in 1964, working in Artificial Intelligence field, used LISP programming language to develop the first CAS system called MATHLAB

Commercial softwares - muMATH and its derivatives- **Reduce** and **Derive** followed by 1970.

The big breakthrough was Macsyma from MIT developed from 1968 to 1982.

Macsyma is now developed as open source MAXIMA and it inspired also several present day popular commercial systems- MATHEMATICA, MAPLE.

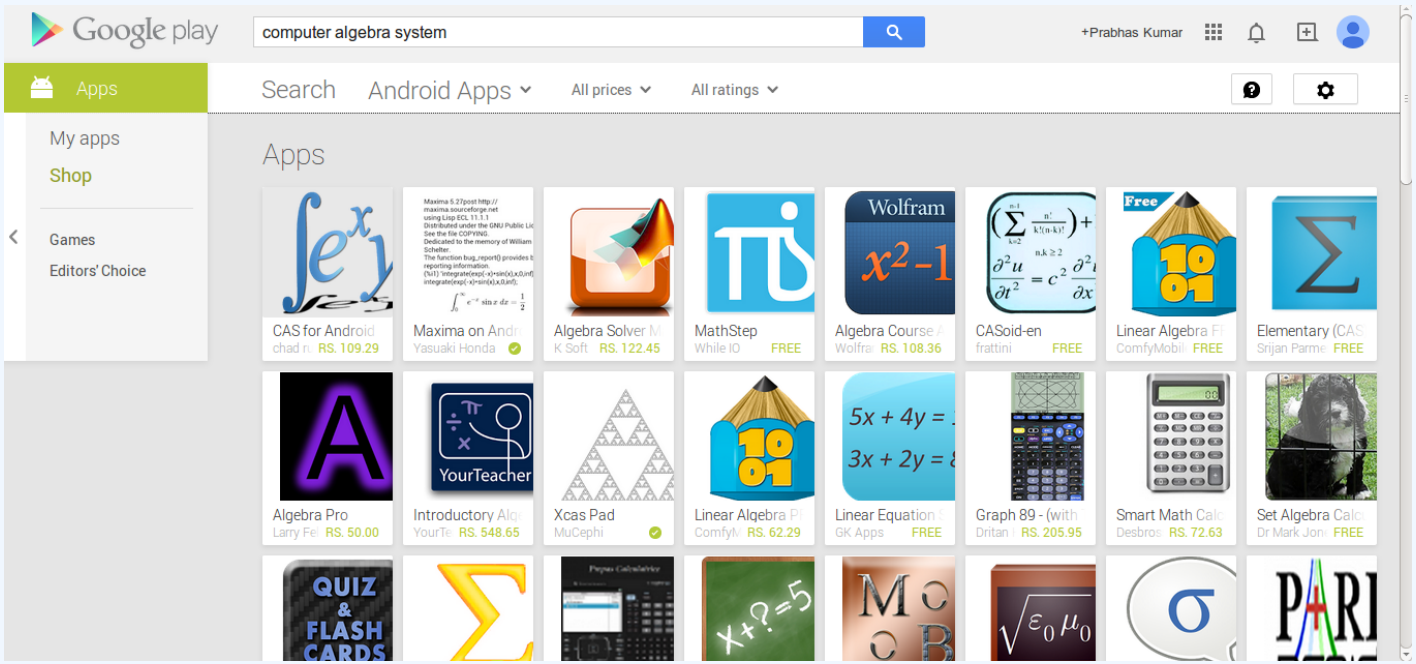
So what is CAS?

SHARDA UNIVERSITY
Beyond Boundaries

Where are CAS?

CAS appeared in calculators by 1987 in HP-28 series, and later in more famous TI calculators.

Now it is in your Mobile Phone



P. K. Yadav

So what is CAS?

Cost of CAS?

A very well documented, and with high end visualization and easy to use GUI commercial CAS system, such as MATHEMATICA, MAPLE etc. can cost above ₹ 5,000 per year for limited student version and above ₹ 120,000 for a single full license.

But there are also free and open source CAS, e.g. MAXIMA, XCAS, MATHICS and quite a few more.

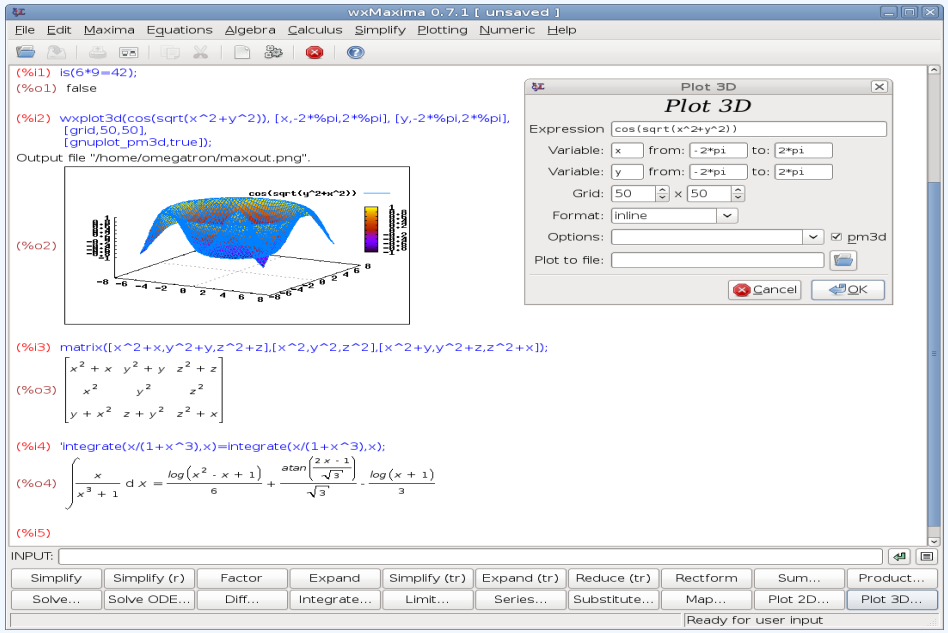
The open source ones typically lack a high end visualization and more often not complete.

We will use **MAXIMA**, an open source and almost complete CAS, for our workshop.

The MAXIMA

The wxMaxima- the MAXIMA GUI

The recent versions of MAXIMA comes with a very intuitive GUI- WxMaxima.



Let us download it from: WxMaxima.
Click here for documentation on Maxima and here for WxMaxima

The wxMaxima- Getting Started

The basic unit of information in Maxima is the expression.

An expression is made up of a combination of operators, numbers, variables, and constants.

Few Maxima Operators

Operator	Description
=	equal to (a=b)
#	not equal to (a # b)
:	assignment operator (a: $x^2 + 2x + 6$)
:=	function definition operator (f(x):= sin(x)

Complete list of **Operators** can be found here.

The wxMaxima– Getting Started

Numbers are the most fundamental unit of expression.

Numbers that are used in Maxima can be one of the different types

Numbers in Maxima
Maxima uses:

(I) Integers, such as **123456**,

(II) Rational numbers, such as **3**/**2**, ratios of integers,

(III) Floats and bigfloats such as **1.234**, **1.234e-6**, and **1.234b5**,

(IV) Complex numbers, such as 4 + 2*%i and a + b*%i. Maxima assumes the symbols *a* and *b* represent real numbers by default.

The wxMaxima- Getting Started

Constants in Maxima The % sign normally proceeds constants

in Maxima, the most common constants are:

Constant	Description
%e	Base of the natural logarithms (e)
%i	The square root of (-1) (i)
%pi	The constant π
inf	Real positive infinity (∞)
minf	Real negative infinity ($-\infty$)
%phi	The golden mean ($(1+\sqrt{5})/2$)
%gamma	The Euler-Mascheroni constant

SHARDA UNIVERSITY
Beyond Boundaries

The wxMaxima- Getting Started

Reserve words in Maxima

Reserved words are special functional words as such they are blocked for naming of variable. A small list is provided below, detail can be obtained from Maxima manual

and	else	if	psi
args	erf	in	exp
at	f90	ind	rem
carg	fib	inrt	some
col	from	li	then
cv	gd	min	und
del	get	next	unless
diff	hav	op	while
do	ic1	or	zeta

The WxMaxima- Input and Output

The Wxmaxima GUI provides symbol (%i1) as input prompt. **1** here refers to the input number, which changes as more input is added. Obviously **i** refers to input.

The corresponding output is (%o1), where **o** refers to output and the number **1** is the output number.

```
(%i1) expr: x/(x^2+1);
```

$$(\%o1)\quad \frac{x}{x^2+1}$$

The keystroke SHIFT + ENTER executes input in wxMaxima

The WxMaxima- Power Tools

Power Tools: Those **functions** that you will very often need.

1. For **Help exact**: **? foo** Enter (help with foo, do not forget space between ? and foo.)
2. For **Help inexact**: **?? foo**- This will provide all information **exact and inexact** about foo.
3. Describe function: **describe(e)**, describes function e.
4. **values** will list of all user defined variables; **remvalue(var)** will delete the variable **var** and **remvalue(all)** will clear all variables.

The WxMaxima- **Variable Assignment**

Symbol	Assignment
:	assigns variable, e.g., $x : 5$, i.e. $x = 5$
::	e.g., if $x : a$, then $x :: 1$ is $a = 1$
:=	defines a function, $f(x) := x^2$, $f(5) = 25$
::=	defines a macro

To **Un-assign** variables, we use
kill(var1, var2,. . .);
kill(all);
To **reset** WxMaxima
wxMaxima: $\rightarrow$ **Maxima** $\rightarrow$Restart **Maxima**

The WxMaxima- **List**

Lists are for creating a set of variables or assign a variable a set of numbers.

Once the list is created it has to be manipulated- e.g. define the position of the element add more elments, remove a element, and so on.

List is the essence of computational methods, we learn how they are done in **Maxima** .

list: [el1, el2, . . .]

Creates a variable **list** that has elements, **el1, el2, . . .** . Take note of square bracket (**[ ]**)

P. K. Yadav

The WxMaxima- **List**

Let, **x: [1,3,6,8]**
be the list **x**, then **x[2]=3, x[4]=8**

i.e., [integer] = indexes the elements of the list x.
The **index** starts from **1**. More on lists:

functions	Output
cons(expr, x)	prepend **expr** to the list **x**
e.g., cons(0.5, x) = [**0.5**, 1,2,3,4]	
endcons(expr, x)	appends expr to to the list **x**
e.g., endcons(8.5, x) = [1,2,3,4,**8.5**]	
append(x1, x2, x3)	merge lists x1, x2. . .
e.g., let x2: [8,9], append(x, x2) = [1,2,3,4 **8,9**]	
length(x)	equals the number of elements of list **x**
e.g., length(x) = 4	

Useful Information-**Maxima**

SHARDA UNIVERSITY
Beyond Boundaries

Information that are just very important

1) Maxima has no `log10` function. We will use `load(log10)` to get that function.
2) Maxima does not provide numerical results by default, i.e., `%pi` = π, to get numerical result, we use the command, `numer`, e.g., `%pi, numer` = 3.14....
3) Number of decimal digits can be fixed using: `fpprintprec: digits`, where digits = positive integer.
4) Maxima evaluates expression automatically, to avoid that we use ‘`expr` = do not evaluate and ‘‘`expr` = do evaluate.
5) Other very useful functions (you should explore) are: `ratsimp`, `map`, `assume`, `load` etc.

A very good resource can be found here, and the **Maxima** manual can be found here.

That was introduction to CAS
Let us get advanced

and learn to solve Equations...,

SHARDA UNIVERSITY
Beyond Boundaries