COMPUTER ALGEBRA, Algorithms, Systems and Applications

Richard Liska, Ladislav Drska, Jiri Limpouch, Milan Sinor, Michael Wester, Franz Winkler

February 11, 1999

Contents

Chapter 1

Introduction

1.1 What are algebraic calculations?

- comparison of numerical to algebraic, or symbolic, computation

Numerical	Symbolic
$2/6 \to 0.333333$	$2/6 \to 1/3$
$2+3 \to 5$	$x+2x \to 3x$
$\cos(3.14159) \to -0.999999$	$\cos(\pi) \to -1$
	$\sin(2x) \to 2\sin x \cos x$
	$\frac{d\,x^2}{d\,x} \to 2x$
$\int_0^{1/2} \frac{x}{x^2-1} d\,x \to 0.1438$	$\int \frac{x}{x^2-1} \to \frac{\ln\|x^2-1\|}{2}$
	$a^2-b^2 \to (a+b)(a-b)$
numerical evaluation	algebraic simplification

- typical features of algebraic computation
  - computation with arbitrary precision numbers—no rounding
  - computation with symbols and variables (e.g., x, y)
  - computation with functions (e.g., sin, cos, f)
  - manipulation of formulas
  - symbolic operations ((see chapter2.8) , (see chapter2.10) , (see chapter2.12) , etc.)
- algebraic computation or symbol manipulation or computer algebra is the field of scientific computation which develops, analyzes, implements and uses algebraic algorithms

1.2 Why do algebraic calculations?

- in many research fields, one often needs to process very large algebraic expressions (of perhaps hundreds of thousands or more terms, e.g., (see chapter5.5.2), computations from the (see chapter5.4.1), (see chapter5.1) calculations) or perform long analytical operations
- compared to a human, a computer does not make errors (assuming the programming is correct! This unfortunately is not always the case ...)
- there exist algorithms which cannot easily be performed by a human with pencil and paper (e.g., (see chapter2.12) , (see chapter2.13))
- many others:

– algebraic solutions are usually more compact than a set of numerical solutions; algebraic solution gives more direct information about the relationship between the variables than figures

– algebraic solutions are always exact, numerical solutions will normally be approximations; this can arise from rounding and truncation errors, further errors can creep in when the user interpolates data given in tabular form

– computer algebra can save both time and effort in solving a wide range of problems; much larger problems can be investigated than by using traditional methods

– computer algebra reduces the need for tables of functions, series and integrals; symbolic computation using computers have highlited many errors in such materials

– traditional teaching of applied mathematics has to involve much time in teaching techniques of solution; computer algebra systems tend to produce solutions quickly and without errors, so they enable more time to be devoted to studying the properties of the solution

– using of computer algebra allows also very effective construction of numerical algorithms and their semi-automatic programming by code generation, the effectiveness of the work and reliability of the results can be strongly increased

1.3 History

- the field originated from the needs of physics researchers
- the first programs dealing with formulas were written by physicists in order to save them from performing long, tedious, error prone calculations by hand
- 1955 - first programs for formula derivation
- 1965 - first general purpose computer systems working with algebraic expressions
- 1975 - new research field with its own conferences, journals, etc.
- 1990 - general spreading of (see chapter3) into almost all branches of science

1.3.1 Chronology of computer algebra systems

- table by Brian Evans ¡evans@eedsp.gatech.edu¿

System name	Year	Related systems	e-mail address, tel.
ALPAK	1964	ALTRAN	(Bell Labs)
ALTRAN	1968		(Bell Labs)
FORMULA (Algol).			
FORMAC		FORMAC (PL/I)	(IBM)
FORMAC (PL/I)			(IBM)
MATHLAB (DECUS)	1968	MACSYMA	(DEC)
CAMAL			
REDUCE	1968		reduce-netlib@rand.org
MACSYMA	1970	Symbolics Macsyma, VAXIMA, DOE-Macsyma, ALJABR, ParaMacs	(See Below)
SchoonShip	1971		archive.umich.edu (FTP)
muMath	1979	Derive	
VAXIMA	1980		(312) 972-7250
SMP	1982	Mathematica	NOT ON MARKET
Symbolics MACSYMA	1983		macsyma-service@symbolics.com
DOE-Macsyma	1984	ALJABR	gcook@llnl.gov
Maple	1985		wmsi@daisy.waterloo.edu

MathCAD	1985(?)	Mathcad	1-800-MATHCAD
Powermath	1985		NOT ON MARKET
REDUCE/PC	1986		reduce-netlib@rand.org
Derive	1988		(Soft Warehouse Inc.)
Mathematica	1988		info@wri.com
Theorist	1988		(415) 543-2252 (Prescience Corp)
PARI	1988(?)		ftp to math.ucla.edu
FORM	1989		form@can.nl
MACSYMA/PC	1989		macsyma-service@symbolics.com
ALJABR	1991		aljabr@fpr.com
Mathcad	1991		1-800-MATHCAD
SymbMath	1991		chen@deakin.oz.au
Axiom	1991		(708) 971-2337
ParaMacs	1991		lph@paradigm.com
SIMATH	1992		marc@math.uni-sb.de

Chapter 2

Algorithms for algebraic computation

2.1 Algebraic structures

- basic requirements
  - precise representation of algebraic structures
  - precise arithmetic with algebraic structures
  - other analytical operations with these structures (e.g., (see chapter2.8) , (see chapter2.10))

2.1.1 Number domains

- $\{Z, +, -, \times\}$integers with operations of addition, subtraction, multiplication; see the examples of (see chapter4.6.1)
- $\{Z_m, +_m, -_m, \times_m\}$integers under modular arithmetic where m is a positive integer
- $\{Q, +, -, \times, /\}$rational numbers with operations of addition, subtraction, multiplication and division; see the examples of (see chapter4.6.1)
- $\{\{x = a+ib, a, b \in Z\}, +, -, \times\}$Gaussian integers, i.e., complex numbers which have integer real and imaginary parts; see the (see chapter4.6.1) examples
- $\{Q(a), +, -, \times, /\}, p(a) = 0$algebraic extension field; the algebraic number $a$is defined by the polynomial $p$with integer coefficients which can be represented precisely; the algebraic number $a$is a root of the polynomial p, e.g., $\sqrt{7}$is represented by the polynomial $a^2 - 7$and the algebraic number $b$could be defined by the polynomial $3b^2 - 5b + 1$; see the examples of (see chapter4.6.1)
- $\{R_f^n, +, -, \times, /\}$floating point numbers with a precision of $n$decimal digits where $n$is an arbitrary positive integer—it can be 100or 1000; see the examples of (see chapter4.6.1)

2.1.2 Algebraic expression domains

- ring of polynomials $O[x_1, ..., x_n]$in $n$variables with operations of addition, subtraction, multiplication and exponentiation by a nonnegative integer; see the examples of (see chapter4.6.2) ; polynomial coefficients can be numbers from a (see chapter2.1.1)
- power series, other kinds of series
- rational functions $K(x_1, ..., x_n)$(extension of polynomials by the operation of division) with operations of addition, subtraction, multiplication, division and exponentiation by an integer; see the examples of (see chapter4.6.3)
- extension of rational functions by radicals (rational exponents), with operations of addition, subtraction, multiplication, division and exponentiation by a rational number

- algebraic functions $\{y_i, p_i(X, y_1, ..., y_m) = 0, i = 1, ..., m\}$implicitly defined by polynomials with integer coefficients $p_i$which depend on algebraic functions $y_i$and variables from $X = \{x_1, x_2, ..., x_n\}$, e.g., the algebraic function $y$defined by the polynomial $x^2 + y^2 - 1$

- elementary transcendental functions $\exp, \ln$, extension of rational functions by elementary transcendental functions; if we have only one variable $x$and an expression contains $\exp x, \ln x$, we can denote $y = \exp x, z = \ln x$and work with a rational function of x, y, z— the extension is given only by rules like $x = \exp \ln x, \ln x^2 = 2 \ln x$, etc.

- transcendental functions: e.g., $\sin, \cos, rmerf$; extension of rational functions by transcendental functions

- matrix rings

- differential fields $(K,')$

- finite groups

- a user can use algebraic expressions from an arbitrary domain for the most part; the program will decide which domain the expression belongs in and use an appropriate algorithm

2.2 Representation of algebraic structures

- to work on a computer with algebraic structures, we need to represent them by some sort of data structures

- representation is very important because often the effectiveness of an algorithm will depend on the representation that is used

2.2.1 Representation of integers

- integers are standardly represented by one computer word of $n$bits; in such a representation, the size of an integer $a$is limited by $a < 2^{n-1}$; for example,

  if $n = 16, a < 32768$,

  if $n = 32, a < 2147483648$and

  if $n = 64, a < 9223372036854775808$

- however, we also need to represent arbitrarily big integers; e.g., see (see chapter2.5.3)

- one possibility to to represent a big integer $a$by an array $A$where all elements of the array are limited by $A_i < 2^{n-1}$and

$$a = \sum_{i=0}^{m} A_i \left(2^{n-1}\right)^i$$

- thus, the big integer is expressed in a number system with base 2^{n-1}

- of course, memory is finite and so we can store only a number of a limited size, however using this representation we can work with quite large integers, e.g., in 1 kB of memory we can store an integer of size 10^{2588}

2.2.2 Representation of polynomials

- representation by a string of characters is not advantageous as it is not a dynamical structure and algorithm implementation would be difficult

Prefix representation

- polynomials are represented by lists using e.g., the prefix operators PLUS for addition, DIFFERENCE for subtraction (or unary MINUS), TIMES for multiplication and EXPT for exponentiation; the first element of a list is the prefix operator and the rest of the elements are its arguments
- the polynomial

$$4x^3 + 2x - 5$$

- is represented by the list `(PLUS (TIMES 4 (EXPT X 3) ) (TIMES 2 X) (MINUS 5) )`
- this representation can be used for arbitrary algebraic expressions
- algorithms using this representation are not particularly fast

Dense representation

- a polynomial in one variable x

$$\sum_{i=0}^{n} a(i)x^i$$

  is represented by the array of all its coefficients $a(i), i = 0, ..., n$

- in this representation, one needs for the polynomial

$$2x^{1000} - 1$$

  to store 1001coefficients even though only 2nonzero coefficients are really necessary; for polynomials in more variables, the situation is even worse

- the time complexity of the algorithm for adding two polynomials of degree $n$in this representation is of the order $O(n)$; polynomials can have high degree and only a few terms so that most of the operations are unnecessary (such as the addition of two zero coefficients)
- most polynomials which we encounter in real life are sparse, that is, most of their coefficients are zero

Sparse representation

- a polynomial in one variable x

$$\sum_{i=0}^{n} a(i)x^i$$

  is represented by pairs of corresponding exponents and coefficients $(i, a(i))$for each term of the polynomial, i.e., all coefficients $a(i)$in this representation are nonzero

- in this representation, the polynomial

$$2x^{1000} - 1$$

  would be represented by the list `( (1000 2) (0 -1) )`

- the time complexity of the algorithm for adding two polynomials with $n$terms is $O(n)$
- to increase the effectiveness of algorithms using this sparse representation, a rule is usually applied which specifies the order of pairs in the list based on the exponents, e.g., in the example above, the lists are sorted in descending order of exponents

Recursive representation

- recursive representation is sparse representation with exponent ordering used to represent multivariate polynomials
- for recursive representation, an ordering of variables has to be chosen, e.g., alphabetically
- the variable which is chosen first in the ordering is called the main variable of the polynomial
- coefficients of the powers of the main variable will be polynomials in the other variables
- for simplicity, we will consider a polynomial in 2variables x, y, with variable ordering $x > y$so that the main variable is x

$$\sum_{i=0}^{n}\sum_{j=0}^{m} a(i,j)x^i y^j = \sum_{i=0}^{n} c(i)x^i \;, \quad \text{where} \quad c(i) = \sum_{j=0}^{m} a(i,j)y^j$$

- such a polynomial is represented by a list of pairs $(i, c(i))$where each coefficient $c(i)$is a polynomial in $y$and is represented by the list of pairs $(j, a(i,j))$
- to distinquish a polynomial in $x, y$from a polynomial in a, c, we replace each pair of (exponent, coefficient) by the triple (variable, exponent, coefficient)
- the polynomial

$$x^3(y^2 + 8y) - x^2(7y + 3) - 5xy + 4$$

  is then represented by the list `((x 3 ((y 2 1) (y 1 8))) (x 2 ((y 1 7) (y 0 3))) (x 1 ((y 1 5))) (x 0 4))`

Recursive representation in Macsyma

- Macsyma Canonical Rational Expression (CRE):

$$\begin{aligned} x^7 + 4x^3 - 2x + 11 &\longrightarrow ((x\ 7\ 1\ 3\ 4\ 1\ {-2}\ 0\ 11)\ .\ 1) \\ (4zy^2 + 5)x^3 - 7x &\longrightarrow ((x\ 3\ (y\ 2\ (z\ 1\ 4)\ 0\ 5)\ 1\ {-7})\ .\ 1) \end{aligned}$$

Recursive representation in Reduce

- Reduce standard form of a polynomial

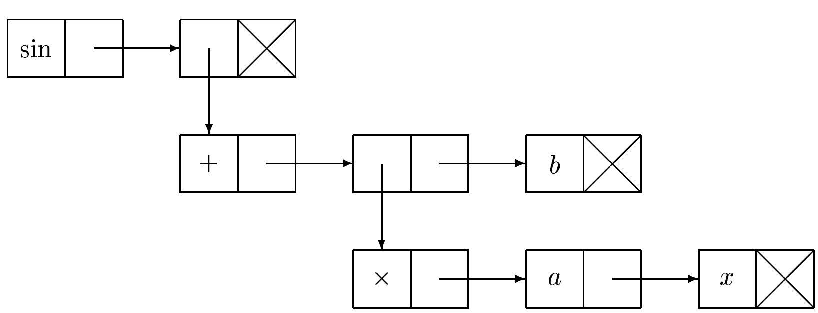
$$\begin{aligned} x^7 + 4x^3 - 2x + 11 &\longrightarrow (((x\ .\ 7)\ .\ 1)\ ((x\ .\ 3)\ .\ 4)\ ((x\ .\ 1)\ .\ {-2})\ .\ 11) \\ (4zy^2 + 5)x^3 - 7x &\longrightarrow (((x\ .\ 3)\ ((y\ .\ 2)((z\ .\ 1)\ .\ 4))\ .\ 5)\ ((x\ .\ 1)\ .\ -7)) \end{aligned}$$

2.2.3 Representation of Expressions

- general prefix representation (external form vs internal form):

$$\sin(ax+b) \longrightarrow (\sin\ \underbrace{(+\ \overbrace{(\times\ a\ x)}\ b)})$$

- internal representation in terms of linked lists:



2.3 Arithmetic

- by arithmetic, we understand this to mean operations of addition, subtraction, multiplication and division

2.3.1 Numeric vs symbolic arithmetic

numeric arithmetic

- normally inexact
- usually produces some sort of answer (may suffer loss of accuracy, fail to converge, etc.)
- arithmetic relatively cheap
- zeroes usually not given special treatment except to avoid computational singularities
- zero detection: number sufficiently small?
- maximize size of pivots to promote stability

symbolic arithmetic

- always exact
- frequently cannot produce an answer (problem is intractable, expression swell, etc.)
- arithmetic often expensive
- zeroes can greatly simplify calculations by making some of them unnecessary
- zero detection: expression equivalent to zero?
- minimize complexity of pivots to stunt expression growth

2.3.2 Arithmetic of integers

- algorithms for addition, subtraction and multiplication of integers are analogous to procedures used by humans for hand calculations; the difference is that with hand calculations, we calculate in the decimal number system while these algorithms use the number system with base 2^{n-1}
- there exists more effective algorithms for multiplication of integers
- the standard procedure for division of integers requires an initial estimate—there exist algorithms that do this estimate with reasonable precision

2.3.3 Arithmetic of polynomials

- polynomial in one variable in (see chapter2.2.2)
- (see chapter2.3.3)
- algorithm for the addition of two polynomials

```
PLUSPOL(a, b) :=
  if  a = ( )  then return b
    else if  b = ( )  then return a
    else
      ea := first first a
      eb := first first b
      ca := second first a                      %a=((ea ca) ...)
      cb := second first b                      %b=((eb cb) ...)
      return(
        if ea > eb then
            cons(first a, pluspol(rest a, b))
          else if ea < eb then
            cons(first b, pluspol(rest b, a))
          else if ca + cb = 0 then
            pluspol(rest a, rest b)
          else cons(list(ea, ca + cb), pluspol(rest a, rest b))
        fi fi fi)
  fi fi
```

- for example, adding the two polynomials $4x^2 - 3x$and $5x + 7$would be performed (after converting them into sparse representation) by `PLUSPOL[((2 4) (1 -3)),((1 5) (0 7))] => ((2 4) (1 2) (0 7))` which is the sparse representation of the sum $4x^2 + 2x + 7$
- algorithm for the multiplication of two polynomials a, b; if $a = a_1 + a_2$and $b = b_1 + b_2$then $ab = a_1b_1 + a_1b_2 + a_2b_1 + a_2b_2$

```
TIMESPOL(a, b) :=
  if  a = ( ) or b = ( )  then return ( )
    else
      ea := first first a
      eb := first first b
      ca := second first a                    %a=((ea ca) ...)
      cb := second first b                    %b=((eb cb) ...)
      return(cons(list(ea + eb, ca * cb),
                  pluspol(timespol(list first a, rest b),
                          timespol(rest a, b))
                 )
            )
  fi
```

- for example, multiplying the two polynomials $4x - 3$a $5x + 2$would be performed (after converting them into sparse representation) by `TIMESPOL[((1 4) (0 -3)),((1 5) (0 2))] => ((2 20) (1 -7) (0 -6))` which is the sparse representation of their product $20x^2 - 7x - 6$

Lists

- a list is an ordered set of elements `(a 1 b ...)`
- an element of a list can be a number, an identifier or another list, e.g., `(a (2))`
- empty list `( )`
- selection operations
  - first returns the first element of a list: `first[(a (2))] => a`
  - second returns the second element of a list: `second[(a (2))] => (2)`
  - rest returns everything but the first element of a list: `rest[(a (2))] => ((2))`
- constructive operations
  - cons adds an element to the beginning of a list: `cons[1, (a b)] => (1 a b)`
  - list builds a list from its arguments: `list[a, b] => (a b)`

2.4 Simplification

- most operations in computer algebra are some form of simplification
- here is a question on what representation to use for algebraic expressions: which form is simpler?

$$(a + b)^2 \Leftrightarrow a^2 + 2ab + b^2$$

- $S$is a canonical simplification operator if and only if

$$\forall t, \quad S(t) \sim t$$

$$\forall t_1, t_2 \;\; t_1 \sim t_2 \Rightarrow S(t_1) = S(t_2)$$

- the canonical simplification operator gives us an unique form for equivalent formulas: it defines a unique representative for each class of equivalent algebraic expressions
- most often non-canonical simplification for general algebraic expressions is used in computer algebra programs—typically, a combination of canonical simplification and pattern matching

2.4.1 Canonical simplification on algebraic domains

- for polynomials, it is necessary to define a term ordering—often the ordering of polynomial variables is sufficient and terms are ordered according to the variables present and their degrees
- for rational functions
  - canonical simplification of numerator and denominator
  - dividing numerator and denominator by their greatest common divisor
  - ensuring that the leading coefficient (coefficient of the first term of a polynomial in a given ordering) satisfies some condition (e.g., making sure that the leading coefficient of the denominator of a rational function is positive)
- for rational functions extended by rational exponents (radicals)
  - unnested radicals (a radical cannot be inside another radical): there exist theories for simplifying these forms
  - nested radicals: the theories are very complicated—it is possible to make use of the same methods as for algebraic functions
- algebraic functions: simplifications are performed "modulo" the Groebner basis of the system of polynomials defining the algebraic functions
- elementary transcendental functions and transcendental functions: there exist structural theorems—these are very complicated

2.4.2 Complexity of expressions

- which is simpler?

$$
\begin{aligned}
(x-1)^{10} &= x^{10} - 10x^9 + 45x^8 - 120x^7 + 210x^6 - 252x^5 + 210x^4 - 120x^3 + 45x^2 - 10x + 1 \\
x^{10} - 1 &= (x-1)(x+1)(x^4 - x^3 + x^2 - x + 1)(x^4 + x^3 + x^2 + x + 1)
\end{aligned}
$$

- how about here?

Form	Expression	complexity
factored sum	$(x+1)^{10} - (x-1)^{10}$	13
expanded	$20x^9 + 240x^7 + 504x^5 + 240x^3 + 20x$	24
factored	$4x(x^4 + 10x^2 + 5)(5x^4 + 10x^2 + 1)$	25

- one classification scheme:

Symbol	Classification	Complexity
R	rational numbers (integers are considered to have a denominator of 1)	sum of the absolute value of the numerator plus the denominator
F	floating point numbers	absolute value of the number
B	extended precision floating point numbers (bigfloats)	absolute value of the number
E	expressions that are not simple numbers	size of the expression (the number of operators and atomic operands)

2.5 Greatest common divisor

- abbreviated by GCD
- very important algorithm—it is used by many other algorithms

2.5.1 GCD of integers

- for two integers $a, b$for which $a \geq b$, we define their quotient $\mathrm{quot}(a, b)$and remainder $\mathrm{rem}(a, b)$as two integers for which

  $a = \mathrm{quot}(a, b)b + \mathrm{rem}(a, b)$

  where $0 \leq \mathrm{rem}(a, b) < b$

- if we denote the greatest common divisor of a, bas $\gcd(a, b)$and if $\mathrm{rem}(a, b)$is nonzero, then

  $\gcd(a, b) = \gcd(b, \mathrm{rem}(a, b))$

- $\gcd(a, b)$divides both $a$and b; it also divides $\mathrm{quot}(a, b)b$so it also has to divide $\mathrm{rem}(a, b)$
- the Euclidean algorithm for calculating the greatest common divisor is based on this identity
- Euclidean algorithm for calculating the GCD of two integers

```
gcd := GCDI(a, b) :=
  [a, b are integers
   algorithm used:
     rem(a, b) - remainder after dividing integer a by integer b]
1. if  a  <  b  then
      r := a
      a := b
      b := r
   fi
2. while  b != 0  do                              %!= means not equal
      r := rem(a, b)
      a := b
      b := r
   od
3. return a
```

2.5.2 GCD of polynomials with rational coefficients

- here we will consider polynomials from $Q[x]$in one variable $x$with rational coefficients
- the degree of polynomial $a(x)$will be denoted by $\deg a(x)$
- for two polynomials $a(x), b(x)$from $Q[x]$for which $\deg a(x) \geq \deg b(x)$, we define their quotient $\mathrm{quot}(a(x), b(x))$and remainder $\mathrm{rem}(a(x), b(x))$as polynomials from $Q[x]$which satisfy

  $a(x) = \mathrm{quot}(a(x), b(x))b(x) + \mathrm{rem}(a(x), b(x))$

  where $0 \leq \deg(\mathrm{rem}(a(x), b(x))) < \deg(b(x))$

- as was the case for integers, the following identity for the greatest common divisor (GCD) of polynomials holds

  $\gcd(a(x), b(x)) = \gcd(b(x), \mathrm{rem}(a(x), b(x)))$

- Euclidean algorithm for calculating GCD of two polynomials with rational coefficients

```
gcd := GCDPQ(a(x), b(x)) :=
  [a, b are polynomials in Q[x]
   algorithms used:
     deg(a)    - degree of polynomial a
     rem(a, b) - remainder after dividing polynomial a by polynomial b]
1. if deg(a) < deg(b) then
      r := a
      a := b
      b := r
   fi
2. while  b != 0  do
      r := rem(a, b)
      a := b
      b := r
   do
3. return a
```

- this algorithm requires many greatest common divisor calculations of two integers when performing calculations with rational numbers and frequently, the integers can become rather large
- these calculations are time consuming so the algorithm is not so efficient
- we can avoid calculations with rational numbers by doing the computations in the domain of polynomials with integer coefficients

Example of GCD in Q[x]

- we want to calculate $\gcd(a, b)$where

$$a = x^8 + x^6 - 3x^4 - 3x^3 + 8x^2 + 2x - 5$$

$$b = 3x^6 + 5x^4 - 4x^2 - 9x + 21$$

- by using the algorithm GCDPQ, we get the following remainders r

$$\begin{aligned} r_1 &= -\frac{5}{9}x^4 + \frac{1}{9}x^2 - \frac{1}{3} \\ r_2 &= -\frac{117}{25}x^2 - 9x + \frac{441}{25} \\ r_3 &= \frac{233150}{6591}x - \frac{102500}{2197} \\ r_4 &= \frac{1288744821}{543589225} \end{aligned}$$

- so the polynomials $a$and $b$are co-prime since their greatest common divisor is 1

2.5.3 GCD of polynomials with integer coefficients

- now we consider polynomials from $Z[x]$in one variable $x$with integer coefficients
- pseudo-division of polynomials $a(x), b(x)$for which $\deg a(x) \geq \deg b(x)$is defined by

$$\mathrm{lcof}(b(x))^{m-n+1} a(x) = \mathrm{pquot}(a(x), b(x)) b(x) + \mathrm{prem}(a(x), b(x))$$

  where $m = \deg a(x), n = \deg b(x)$and $\mathrm{lcof}(b(x))$is the leading coefficient of the polynomial $b(x)$, i.e., the coefficient of the n-th power of x, so that the pseudo-quotient $\mathrm{pquot}(a(x), b(x))$and the pseudo-remainder $\mathrm{prem}(a(x), b(x))$are also from $Z[x]$

- then $0 \leq \deg \text{prem}(a(x), b(x)) < \deg b(x)$
- the polynomial $a(x)$is primitive if all its coefficients are mutually co-prime

$$a(x) = \text{cont}a(x)\,\text{pp}a(x)$$

- cont$a(x)$is the greatest common divisor of all coefficients of the polynomial and pp$a(x)$is the primitive part of the polynomial a, hence, $a(x) = \text{cont}a(x)\text{pp}a(x)$
- Euclidean algorithm for polynomial reminder sequence

```
gcd := GCDPRS(a(x), b(x)) :=
      [suppose that degree of polynomial a is greater or equal to the
       degree of polynomial b, i.e., deg(a) >= deg(b)
       algorithms used:
         prem(a, b) - pseudo-remainder of polynomial a with polynomial b
         pp(a)      - primitive part of the polynomial a
         gcdi(j, k) - gcd of two integers j, k]
    1. A := pp(a)
       B := pp(b)
    2. while  B != 0  do
          r := prem(A, B)
          A := B
          B := r
       od
    3. return gcdi(cont(a), cont(b)) pp(a)
```

- coefficients and partial results grow very quickly—this algorithm is also not efficient
- it is possible to calculate the primitive part of the remainder in each step, however, the calculation of the primitive part requires the calculation of many greatest common divisors of integer coefficients which can often be large
- modular approach: perform calculations modulo a prime p
  - apply homomorphism to coefficients
  - calculate the greatest common divisor of the transformed polynomials modulo p
  - use the Chinese remainder algorithm for reconstructing the coefficients of the greatest common divisor back in the integers

Example of GCD in Z[x]

- we want to calculate $\gcd(a, b)$, where

$$a = x^8 + x^6 - 3x^4 - 3x^3 + 8x^2 + 2x - 5$$

$$b = 3x^6 + 5x^4 - 4x^2 - 9x + 21$$

- using the algorithm GCDPRS, we get following pseudo-remainders r

pseudo-reminder	primitive part
$r_1 = -15x^4 + 3x^2 - 9$	$-5x^4 + x^2 - 3$
$r_2 = 15795x^2 + 30375x - 59535$	$13x^2 + 25x - 49$
$r_3 = 1254542875143750x - 1654608338437500$	$4663x - 6150$
$r_4 = 12593338795500743100931141992187500$	1

2.6 Resultant

- let a, bbe polynomials with coefficients from the domain O
- the domain Ocan be e.g., integers or polynomials in other variables

$$a = \sum_{i=0}^{n} a_i x^i \ , \qquad b = \sum_{i=0}^{m} b_i x^i$$

- the Sylvester matrix of polynomials $a, b$is the matrix

$$\begin{pmatrix}
a_n & a_{n-1} & \cdots & a_1 & a_0 & 0 & 0 & \cdots & 0 \\
0 & a_n & a_{n-1} & \cdots & a_1 & a_0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & a_n & a_{n-1} & \cdots & a_1 & a_0 & 0 \\
0 & \cdots & 0 & 0 & a_n & a_{n-1} & \cdots & a_1 & a_0 \\
b_m & b_{m-1} & \cdots & b_1 & b_0 & 0 & 0 & \cdots & 0 \\
0 & b_m & b_{m-1} & \cdots & b_1 & b_0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & \cdots & 0 & b_m & b_{m-1} & \cdots & b_1 & b_0 & 0 \\
0 & \cdots & 0 & 0 & b_m & b_{m-1} & \cdots & b_1 & b_0
\end{pmatrix}$$

- where $m$rows are constructed from the coefficients of the polynomial $a$and $n$rows from the coefficients of the polynomial b
- the resultant of the polynomials $a$and $b$with respect to the variable x, denoted by $\mathrm{Res}(x, a, b)$, is the determinant of the Sylvester matrix
- using the special structure of the Sylvester matrix, we can calculate the resultant by the following recursive algorithm
- Algorithm

```
res := RES(x, a, b) :=
  [a, b are polynomials
   algorithms used:
     rem(a, b) - remainder after dividing a by b
     deg(a)    - degree of the polynomial a
     lcof(a)   - leading coefficient of the polynomial a, i.e., the
                 coefficient of x^deg(a)]
1. n := deg(a)
   m := deg(b)
2. if n > m then res :=(-1)^(n m) RES(x, b, a)
     else lc := lcof(a)
          if n = 0 then res := lc^m
            else r := rem(b, a)
                 if r = 0 then res := 0
                   else p := deg(r)
                        res := lc^(m - p) RES(x, a, r)
                 fi
          fi
   fi
```

- this algorithm gradually replaces rows in the Sylvester matrix derived from the coefficients of the polynomial $b$by rows derived from the coefficients of the remainder $r = rem(b, a)$; these new rows are computed as linear combinations of the rows of the original matrix

- after the above replacement, the matrix has only zeroes below the diagonal in its first $m-p$columns while the diagonal has the values lc; the matrix block consisting of the last $n-p$rows and $n-p$columns is the Sylvester matrix of the polynomials a, r; thus, in general case of the algorithm, we can use the formula $\text{Res}(x, a, b) := lc^{m-p}\text{Res}(x, a, r)$

2.7 Solving polynomial equations

- solving linear systems is easy as only the inversion of a matrix is required

- system of polynomial equations

$$
\begin{aligned}
f_1(x_1, \ldots, x_n) &= 0 \\
&\vdots \\
f_m(x_1, \ldots, x_n) &= 0
\end{aligned}
$$

- Grobner basis according to lexicographic ordering of variables $x_1 < x_2 < \ldots < x_n$(very high time complexity)

- the system of polynomial equations and its Grobner basis have the same solution

- example

$$
\begin{aligned}
f_1 &= xz - xy^2 - 4x^2 - \frac{1}{4} \\
f_2 &= y^2z + 2x + \frac{1}{2} \\
f_3 &= x^2z + y^2 + \frac{1}{2}x
\end{aligned}
$$

- Grobner basis of these polynomials with variables ordering $x < y < z$

$$
\begin{aligned}
g_1 &= z + \frac{64}{65}x^4 - \frac{432}{65}x^3 + \frac{168}{65}x^2 - \frac{354}{65}x + \frac{8}{5} \\
g_2 &= y^2 - \frac{8}{13}x^4 + \frac{54}{13}x^3 - \frac{8}{13}x^2 + \frac{17}{26}x \\
g_3 &= x^5 - \frac{27}{4}x^4 + 2x^3 - \frac{21}{16}x^2 + x + \frac{5}{32}
\end{aligned}
$$

- this is in "triangular" form

- $x$can be determined by solving the 3rd equation; after substituting $x = x_i$into the 2nd equation, we can determine y, etc.

- one approximate solution is $(-0.128475, 0.321145, -2.356718)$

- solving (even numerically) of a polynomial equation in one variable is much simpler than solving a system of polynomial equations

- solving a polynomial system is transformed into the successive solution of equations in one variable; see the example (see chapter4.6.2)

2.8 Differentiation

- differentiation algorithm is very simple
- for differentiation, the following rules are sufficient:

```
(a + b)' = a' + b'

(a - b)' = a' - b'

(a b)' = a'b + a b'

(a/b)' = (a'b - a b')/b^2

f(g(t))' = f'(g(t)) g'(t)
```

  rules for differentiation of functions like $\ln, \exp, \sin, \cos, f$(these can be implemented using a table look up)

- (see chapter4.6.2)

2.9 Summation

- given the sequence $a_1, a_2, ..., a_n, ...$, what is the sum

$$S(n) = \sum_{i=1}^{n} a_i$$

- discrete analogue of integration

2.9.1 Simple example

- let

$$S(n) = \sum_{i=1}^{n} \frac{1}{i(i+2)}, \text{i.e.,} a_i = \frac{1}{i(i+2)}$$

- then we can express the ratio

$$\frac{a_n}{a_{n-1}} = \frac{(n-1)(n+1)}{n(n+2)} = \frac{q(n)p(n)}{p(n-1)r(n)}$$

- where we have denoted

$$\begin{aligned} p(n) &= n+1 \\ q(n) &= n-1 \\ r(n) &= n+2 \end{aligned}$$

- the idea is to try to express the sum $S(n)$as

$$\begin{aligned} S(n) &= \frac{q(n+1)}{p(n)} a_n f(n) \\ &= \frac{n}{(n+1)} \frac{1}{n(n+2)} f(n) \end{aligned} \tag{2.1}$$

- where $f(n)$is a polynomial in n; note that

$$a_n = S(n) - S(n-1) \tag{2.2}$$

- substituting (2.1) into (2.2), we obtain a recurrence relation for $f(n)$:

$$n+1 = nf(n) - (n+2)f(n-1) \tag{2.3}$$

- to solve the recurrence relation, we need to know the degree of the polynomial $f(n)$
- we can rewrite (2.3) as

$$\begin{aligned} n+1 &= (n-(n+2))\frac{f(n)+f(n-1)}{2} \\ &\quad +(n+(n+2))\frac{f(n)-f(n-1)}{2} \end{aligned} \tag{2.4}$$

- introducing

$$f(n) = c_k n^k + c_{k-1}n^{k-1} + \ldots + c_0$$

and substituting this formula into (2.4), we obtain

$$n+1 = (k-2)c_k n^k + O(n^{k-1})$$

- from which it follows that $k \leq 2$(if $k > 2$, then the previous equation implies that $c_k = 0$)
- therefore, $f(n)$is a polynomial of at most degree two

$$f(n) = c_2 n^2 + c_1 n + c_0$$

- the solution of (2.3) is then

$$\begin{aligned} c_0 &= x \in R \\ c_1 &= \frac{5+6x}{4} \\ c_2 &= \frac{3+x}{4} \end{aligned}$$

where $x$is an arbitrary real parameter

- the value of $x$is obtained from the initial condition $S(0) = 0$, which gives $x = 0$and hence,

$$f(n) = \frac{3n^2+5n}{4}$$

- the final solution is

$$\boxed{S(n) = \sum_{i=1}^{n} \frac{1}{i(i+2)} = \frac{3n^2+5n}{4(n^2+3n+2)}}$$

2.9.2 Gosper algorithm

- given the sequence a_i, what is the sum

$$S(n) = \sum_{i=1}^{n} a_i$$

- we know that $a_i = S(i) - S(i-1)$
- let us assume that $S(n)/S(n-1)$is a rational function
- the quotient of two succesive terms of the sequence can be expressed as

$$\begin{aligned}\frac{a_n}{a_{n-1}} &= \frac{S(n) - S(n-1)}{S(n-1) - S(n-2)} \\ &= \frac{S(n)/S(n-1) - 1}{1 - S(n-2)/S(n-1)}\end{aligned}$$

- thus, $a_n/a_{n-1}$is also a rational function
- on the basis of the following lemma, the quotient can be always written as

$$\frac{a_n}{a_{n-1}} = \frac{p(n)q(n)}{p(n-1)r(n)}$$

- where $p, q, r$are polynomials for which $\gcd(q(n), r(n+j)) = 1$for all $j \geq 0$(gcdis the (see chapter2.5))
- **Lemma** A rational function $a(n)/b(n)$can be always written as

$$\frac{a(n)}{b(n)} = \frac{p(n)q(n)}{p(n-1)r(n)},$$

  where $p, q, r$are polynomials in $n$and

$$\gcd(q(n), r(n+j)) = 1, \forall j \geq 0,$$

- **Theorem** Let $S(n)/S(n-1)$be a rational function as above. Then

$$f(n) = S(n)\frac{p(n)}{q(n+1)a_n}$$

  is a polynomial for which

$$p(n) = q(n+1)f(n) - r(n)f(n-1).$$

- Gosper algorithm

```
(S(n),b):=GOSPER( a(n))
  [assumes that a(n)/a(n-1) is a rational function,
   if S(n)/S(n-1) is also a rational function then b=true
   and S(n) is computed, otherwise b=false
   algorithms used:
     num(a)        - numerator of the rational function a
     den(a)        - denominator of the ratinal function a
     Res(x, p, q)  - resultant of polynomials p, q with respect to the
                     variable x
     gcd(p,q)      - greatest common divisor of the polynomials p and q
```

```
      deg(p(n))     - degree of the polynomial p(n)
      coef(p(n), i) - coefficient of the ith power of n in the polynomial
                      p(n)]
1. b := true
2. if a(n) = 0 then S(n) := 0; return fi
3. p(n) := 1
   q(n) := num(a(n) / a(n-1))
   r(n) := den(a(n) / a(n-1))
4. while (Res(n, q(n), r(n+j)) has nonnegative integer root j = j0) do
     g(n) := gcd(q(n), r(n+j0))
     q(n) := q(n)/g(n)
     r(n) := r(n)/g(n-j0)
     p(n) := p(n) g(n) g(n-1) ... g(n-j0+1)
   od
5. lp := deg(q(n+1) + r(n))
   lm := deg(q(n+1) - r(n))
   if lp <= lm then k := deg(p(n))-lm
   else
     k0 := 2 (-lp coef(q, lp) - coef(q, lp-1) + coef(r, lp-1))/
             (coef(q, lp) + coef(r, lp))
     if (k0 is integer) then k := max(k0, deg(p(n))-lp+1)
     else k := deg(p(n))-lp+1
     fi
   fi
   if k < 0 then b := false; return fi;
6. solve recurrence relation  p(n)=q(n+1)f(n)-r(n)f(n-1)
   with initial condition     f(1)=p(1)/q(2)
   for f(n)=ck n^k + ... + c0
   if (solution does not exist) then b := false; return fi;
7. S(n) := q(n+1) a(n) f(n)/p(n);
   return
```

2.9.3 Examples using the Gosper algorithm

$$\sum_{n=1}^{m} \frac{\prod_{j=1}^{n-1}(b_j^2 + c_j + d)}{\prod_{j=1}^{n}(b_j^2 + c_j + c)} = \frac{1 - \prod_{j=1}^{m} \frac{(b_j^2+c_j+c)}{b_j^2+c_j+c}}{c - d}$$

$$\sum_{n=1}^{m} nx^n = \frac{mx^{m+2} - (m+1)x^{m+1} + x}{(x-1)^2}$$

- (see chapter4.6.5)

2.10 Integration

- as in the summation problem, the basic structure of the algorithm is the following:
  - a structural theorem gives the general form of the solution
  - need to determine the degree bounds on the polynomial appearing in the solution
  - finally, determine the coefficients of these polynomials

2.10.1 Integration of rational functions

- **Theorem** (Rothstein) Let $A(x), B(x)$be polynomials with $B(x)$square free (i.e., there are no squared or higher degree factors in its factorization) and suppose that the degree of $A$is less than the degree of $B($ $\deg(A) < \deg(B))$. Then

$$\int \frac{A(x)}{B(x)} d\,x = \sum_{i=1}^{n} c_i \ln v_i,$$

where $c_1, ..., c_n$are the distinct roots of the polynomial $R(c) = \text{Res}(x, A(x) - cB'(x), B(x))$and $v_i = \gcd(A(x) - c_i B'(x), B(x))$for $i = 1, ..., n($ Res is the (see chapter2.6)).

2.10.2 Integration of elementary transcendental functions

- extend the rational functions by the natural logarithm and exponential function (see (see chapter2.1.2))
- **Theorem** (Liouville) Let Kbe a differential field and $f$be from K. Then, an elementary extension of the field K, which has the same field of constants as $K$and contains an element $g$such that $g' = f$, exists if and only if there exist constants $c_1, ..., c_n$from $K$and functions $u, u_1, ..., u_n$from Ksuch that

$$f = u' + \sum_{i=1}^{n} c_i \frac{u_i'}{u_i}$$

i.e.,

$$g = \int f = u + \sum_{i=1}^{n} c_i \ln u_i$$

- Risch 1968-1969 - first decision procedure for the integration of elementary transcendental and algebraic functions; the procedure determines if the integral exists within a given class of functions and if so, then calculates the value of the integral

2.10.3 Integration examples

$$\int \frac{x}{x^2 - 2} d\,x = \frac{1}{2} \ln(x^2 - 2)$$

$$\begin{aligned} \int \quad & \frac{1}{x^3 + 2} d\,x = -\frac{\ln(x^2 - \sqrt[3]{2}x + \sqrt[3]{2}^2}{6\sqrt[3]{2}^2} \\ & + \frac{\arctan\left(\frac{2x - \sqrt[3]{2}}{\sqrt[3]{2}\sqrt{3}}\right)}{\sqrt[3]{2}^2\sqrt{3}} + \frac{\ln(x + \sqrt[3]{2})}{3\sqrt[3]{2}^2} \end{aligned}$$

$$\begin{aligned} \int \quad & \left[2xe^{x^2} \ln(x) + \frac{e^{x^2}}{x} + \frac{\ln(x) - 2}{(\ln(x)^2 + x)^2} \right. \\ & \left. + \frac{2\ln(x)/x + 1/x + 1}{\ln(x)^2 + x} \right] d\,x \\ & = e^{x^2} \ln(x) - \frac{\ln(x)}{\ln(x)^2 + x} + \ln(\ln(x)^2 + x) \end{aligned}$$

- (see chapter4.6.5)

2.11 Ordinary differential equations

- Kovacic algorithm for differential equations of the type

$$ay'' + by' + cy = 0$$

where $a, b, c$are polynomials in x(implemented in Macsyma)

- Singer algorithm for differential equations of the type

$$y^{(n)} + c_{n-1}y^{(n-1)} + \ldots + c_1 y' + c_0 y = 0$$

where $c_{n-1}, ..., c_0$are rational or algebraic functions (not implemented until now due to its massive complexity)

- algorithms that look for Liouville type solutions (i.e., rational functions extended by algebraic functions, exponential functions and integrals) or decide if such a solution exists
- (see chapter4.6.5)

2.12 Polynomial factorization

- consider polynomials with integer coefficients (it is simple to convert a polynomial with rational coefficients into a problem in this domain)
- factorization splits a given polynomial $a(x)$into the product of polynomials $a_j(x)$:

$$a(x) = \prod_{j=1}^{n} a_j(x)$$

- a polynomial $a(x)$is "square free", i.e., it does not have a factor which is the second or higher power of a polynomial, if and only if $\gcd(a(x), a'(x)) = 1$, that is, there does not exist a polynomial other than the constant 1that divides the polynomial $a(x)$and its derivative
- if the polynomial $a(x)$is not "square free" then

$$a(x) = b(x)^2 c(x)$$

$$a'(x) = 2b(x)b'(x)c(x) + b(x)^2 c'(x)$$

and $\gcd(a(x), a'(x)) = b(x)d(x)$

- Berlekamp algorithm for factorization of "square free" polynomials in one variable modulo a prime p
- Berlekamp-Hensel algorithm for factorization of "square free" polynomials in one variable with integer coefficients
- Kronecker algorithm for factorization of polynomial in more variables
- these algorithms exhibit exponential complexity
- there exists algorithms with only polynomial complexity
- uses of factorization: solving of polynomial equations, integration of rational functions, etc.
- example

$$\begin{aligned} 8x^3y &+ 40x^2y^4 + 2x^2y^2z + 4x^2y \\ &+ 4x^2z^2 + 10xy^5z + 20xy^4 + 20xy^3z^2 \\ &+ xyz^3 + 2xz^2 + 5y^4z^3 + 10y^3z^2 \\ &= (2xy + z^2)(4x + yz + 2)(x + 5y^3) \end{aligned}$$

- (see chapter4.6.2)

2.13 Quantifier elimination

- quantified formula in prenex form over a real field:

$$G = (Q_{f+1}x_{f+1}) \dots (Q_r x_r) F(x_1, \dots, x_r)$$

  where $Q_i$is a general quantifier (for all) or existential quantifier (there exists) and $F$is a logical combination of polynomial equations and inequalities in the variables $x_1, ..., x_r$

- $f$variables are free (not quantified) and $r - f$variables are quantified
- there exists a quantifier free formula Hin free variables $x_1, ..., x_f$which is equivalent to the formula G
- quantifier elimination is the procedure which transforms the quantified formula Ginto the quantifier free formula H
- around 1930 Tarski proved that quantifier elimination is possible
- first quantifier elimination method was proposed by Tarski 1951, however its complexity cannot be bound by any tower of exponentials
- cylindrical algebraic decomposition method by Collins 1975 made a breakthrough having "only" double exponential complexity
- improvements in the method of partial cylindrical algebraic decomposition: Collins and Hong 1991, programm QEPCAD
- example: the quantified formula

$$\begin{aligned} (\forall S_1) \quad (\forall S_2) \quad & (C > 0) \wedge \\ & [(0 \le S_1 \le 1 \wedge 0 \le S_2 \le 1) \Longrightarrow \\ & -2S_1S_2C^3 + 3S_1S_2C^2 + S_1C^2 \\ & -2S_1C + S_2C^2 - 2S_2C + 1 \ge 0] \end{aligned}$$

- is equivalent to the quantifier free formula

$$0 < C \le \frac{1}{2}$$

Example with QEPCAD program

- formula is quantified for all TGX and TGY;

  inputs , `outputs`

```
=======================================================

                Quantifier Elimination
                          in
            Elementary Algebra and Geometry
                          by
      Partial Cylindrical Algebraic Decomposition

               Version 10 (Interactive)
                       June 1992

                          by
                      Hoon Hong
               (hhong@risc.uni-linz.ac.at)
       Research Institute for Symbolic Computation
```

```
===========================================================

<<QEPCAD>> Enter an informal description  between '[' and ']':
```

[Wendr M2, 1. factor¡=0]

```
<<QEPCAD>> Enter a variable list:
```

(A,B,TGX,TGY)

```
<<QEPCAD>> Enter the number of free variables:
```

2

```
<<QEPCAD>> Enter a prenex formula:
```

(A TGX) (A TGY)
(A + B ¡= 1 / A + B ¿= -1 / A - B ¡= 1 / A - B ¿= -1
/ (A^2 TGX^2 (TGY^2 + 1) + 2 A B TGX TGY (- TGX TGY + 1) + A TGX^2 (
TGY^2 + 1) + B^2 TGY^2 (TGX^2 + 1) - B TGY^2 (TGX^2 + 1) ¡= 0
)).

```
===========================================================


<<QEPCAD>> Before Normalization>>
```

finish

```
===========================================================


An equivalent quantifier-free formula:

( A <= 0 /\ B >= 0 /\ B + A - 1 <= 0 /\ B + A + 1 >= 0 /\ \\
  B - A + 1 >= 0 /\ B - A - 1 <= 0 )
```

Chapter 3

Integrated mathematical systems

3.1 Computer algebra systems

- specialized systems
  - TRIGMAN 1970, celestial mechanics
  - SCHOONSCHIP 1971, quantum physics
  - CAMAL 1975, celestial mechanics, general theory of relativity
  - SHEEP 1977, general theory of relativity, very fast
- general purpose systems
  - REDUCE 1968, classical system, delivered with sources
  - MACSYMA 1970, classical system, mainly in USA
  - MAPLE 1985, fast and good system
  - DERIVE 1988, on PC, even on pocket computers
  - MATHEMATICA 1988, nice graphics and user interface (notebook)
  - AXIOM 1991, former Scratchpad II, modern "object oriented" concepts, most complete system
- (see chapter1.3.1)
- for other computer algebra systems and packages, including also public systems, consult the overview at CAIN Netherlands or the collection of symbolic software at the University of Berkeley

3.2 Peculiarities of programming in computer algebra systems

- computer algebra system (CAS) can be used as an algebraic calculator
- one can program in a CAS using a high level language
- one can get VERY LARGE algebraic expressions quite easily (see chapter**??**)
- it is hard to guess the memory and CPU time requirements for a given calculation
- often calculations need to be done only once—once a result is known, it does not need to be calculated again
- efficiency questions (algorithmic)
- questions of representation which can influence the efficiency as well
- experience is very important

3.3 Expression Swell

EXPRESSION SWELL is a common phenomenon of exact computations in which the size of numbers and expressions involved in a calculation grows dramatically as the calculation progresses.

INTERMEDIATE EXPRESSION SWELL is an important special case of expression swell in which, during the middle stages of a calculation, intermediate expressions can expand substantially, but the final results of the calculation are comparatively simple.

3.3.1 An Example of Expression Swell

- consider the Hankel matrix

$$\begin{pmatrix} -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \end{pmatrix}$$

- it has the following characteristic polynomial:

$$\begin{aligned} c(\lambda) &= \lambda^9 + \lambda^8 - 40\lambda^7 - 24\lambda^6 + 240\lambda^5 + 144\lambda^4 \\ &= \lambda^4(\lambda+6)(\lambda^4 - 5\lambda^3 - 10\lambda^2 + 36\lambda + 24) \end{aligned}$$

- the Hankel matrix has four zero eigenvalues, one eigenvalue is -6, and the other four eigenvalues are roots of the seemingly simple looking quartic polynomial

$$\lambda^4 - 5\lambda^3 - 10\lambda^2 + 36\lambda + 24$$

- here is one root of this quartic polynomial

$$-\frac{\sqrt{-\frac{\left(18\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{2/3}-465\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}+1856\right)\sqrt{\frac{36\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{2/3}+465\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}+3712}{\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}}}+999\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25\ldots}{\ldots}\right.\cdots}{\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}\sqrt{\frac{36\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{2/3}+465\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}+3712}{\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}}}}\cdots}}{6\sqrt{2}}\cdots$$

$$-\frac{\sqrt{\frac{36\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{2/3}+465\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}+3712}{\left(\frac{16\sqrt{8071}\,i}{3}+\frac{25136}{27}\right)^{1/3}}}}{12}+\frac{5}{4}$$

- the other three roots are similar in structure.

3.3.2 An Example of Intermediate Expression Swell

- the left-hand side of the (tensor) Bianchi identity for a symmetric connection is

$$K_{j\ \ hk|p}^{\ell} + K_{j\ \ kp|h}^{\ell} + K_{j\ \ ph|k}^{\ell}\ ,$$

where K is the Riemann curvature tensor

- expanding in terms of Christoffel symbols of the second kind, one obtains

$$
\begin{aligned}
& - \Gamma_{\#19}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#19}{}_{\#27}\Gamma_{p}{}^{\#27}{}_{k} + \Gamma_{j}{}^{\ell}{}_{h,\#25}\Gamma_{p}{}^{\#25}{}_{k} + \Gamma_{\#19}{}^{\ell}{}_{\#22}\Gamma_{j}{}^{\#19}{}_{h}\Gamma_{p}{}^{\#22}{}_{k} - \Gamma_{j}{}^{\ell}{}_{\#20,h}\Gamma_{p}{}^{\#20}{}_{k} \\
& + \Gamma_{j}{}^{\ell}{}_{\#18,k}\Gamma_{p}{}^{\#18}{}_{h} + \Gamma_{\#10}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#10}{}_{\#16}\Gamma_{p}{}^{\#16}{}_{h} - \Gamma_{\#10}{}^{\ell}{}_{\#12}\Gamma_{j}{}^{\#10}{}_{k}\Gamma_{p}{}^{\#12}{}_{h} - \Gamma_{j}{}^{\ell}{}_{k,\#11}\Gamma_{p}{}^{\#11}{}_{h} \\
& + \Gamma_{j}{}^{\ell}{}_{\#9,h}\Gamma_{k}{}^{\#9}{}_{p} + \Gamma_{\#1}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#1}{}_{\#7}\Gamma_{k}{}^{\#7}{}_{p} - \Gamma_{\#1}{}^{\ell}{}_{\#3}\Gamma_{j}{}^{\#1}{}_{h}\Gamma_{k}{}^{\#3}{}_{p} - \Gamma_{j}{}^{\ell}{}_{h,\#2}\Gamma_{k}{}^{\#2}{}_{p} + \Gamma_{j}{}^{\ell}{}_{p,\#18}\Gamma_{k}{}^{\#18}{}_{h} \\
& + \Gamma_{\#10}{}^{\ell}{}_{\#15}\Gamma_{j}{}^{\#10}{}_{p}\Gamma_{k}{}^{\#15}{}_{h} - \Gamma_{\#10}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#10}{}_{\#13}\Gamma_{k}{}^{\#13}{}_{h} - \Gamma_{j}{}^{\ell}{}_{\#11,p}\Gamma_{k}{}^{\#11}{}_{h} - \Gamma_{h}{}^{\#20}{}_{k}\Gamma_{j}{}^{\ell}{}_{p,\#20} \\
& + \Gamma_{\#9}{}^{\ell}{}_{k,h}\Gamma_{j}{}^{\#9}{}_{p} + \Gamma_{\#1}{}^{\ell}{}_{h}\Gamma_{\#7}{}^{\#1}{}_{k}\Gamma_{j}{}^{\#7}{}_{p} - \Gamma_{\#1}{}^{\ell}{}_{k}\Gamma_{\#4}{}^{\#1}{}_{h}\Gamma_{j}{}^{\#4}{}_{p} + \Gamma_{\#19}{}^{\ell}{}_{h}\Gamma_{\#27}{}^{\#19}{}_{k}\Gamma_{j}{}^{\#27}{}_{p} \\
& + \Gamma_{\#20}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#20}{}_{p,h} - \Gamma_{\#2}{}^{\ell}{}_{h,k}\Gamma_{j}{}^{\#2}{}_{p} + \Gamma_{\#19}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#19}{}_{p,k} - \Gamma_{\#19}{}^{\ell}{}_{\#26}\Gamma_{h}{}^{\#26}{}_{k}\Gamma_{j}{}^{\#19}{}_{p} \\
& + \Gamma_{\#19}{}^{\#26}{}_{h}\Gamma_{\#26}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#19}{}_{p} - \Gamma_{\#19}{}^{\#26}{}_{k}\Gamma_{\#26}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#19}{}_{p} + \Gamma_{\#19}{}^{\ell}{}_{h,k}\Gamma_{j}{}^{\#19}{}_{p} - \Gamma_{\#18}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#18}{}_{p,k} \\
& - \Gamma_{\#10}{}^{\ell}{}_{k}\Gamma_{\#16}{}^{\#10}{}_{h}\Gamma_{j}{}^{\#16}{}_{p} - \Gamma_{\#10}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#10}{}_{p,h} + \Gamma_{\#10}{}^{\#15}{}_{h}\Gamma_{\#15}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#10}{}_{p} - \Gamma_{\#10}{}^{\#15}{}_{k}\Gamma_{\#15}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#10}{}_{p} \\
& - \Gamma_{\#10}{}^{\ell}{}_{k,h}\Gamma_{j}{}^{\#10}{}_{p} + \Gamma_{h}{}^{\#9}{}_{p}\Gamma_{j}{}^{\ell}{}_{k,\#9} - \Gamma_{\#9}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#9}{}_{k,h} - \Gamma_{\#1}{}^{\ell}{}_{h}\Gamma_{\#7}{}^{\#1}{}_{p}\Gamma_{j}{}^{\#7}{}_{k} \\
& - \Gamma_{\#19}{}^{\ell}{}_{h}\Gamma_{\#27}{}^{\#19}{}_{p}\Gamma_{j}{}^{\#27}{}_{k} + \Gamma_{\#25}{}^{\ell}{}_{h,p}\Gamma_{j}{}^{\#25}{}_{k} + \Gamma_{\#19}{}^{\ell}{}_{p}\Gamma_{\#23}{}^{\#19}{}_{h}\Gamma_{j}{}^{\#23}{}_{k} - \Gamma_{\#20}{}^{\ell}{}_{p,h}\Gamma_{j}{}^{\#20}{}_{k} \\
& + \Gamma_{\#10}{}^{\ell}{}_{p}\Gamma_{\#13}{}^{\#10}{}_{h}\Gamma_{j}{}^{\#13}{}_{k} + \Gamma_{\#11}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#11}{}_{k,p} + \Gamma_{\#10}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#10}{}_{k,h} - \Gamma_{\#10}{}^{\#12}{}_{h}\Gamma_{\#12}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#10}{}_{k} \\
& + \Gamma_{\#10}{}^{\#12}{}_{p}\Gamma_{\#12}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#10}{}_{k} + \Gamma_{\#10}{}^{\ell}{}_{p,h}\Gamma_{j}{}^{\#10}{}_{k} - \Gamma_{\#1}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#1}{}_{k,p} + \Gamma_{\#1}{}^{\ell}{}_{\#6}\Gamma_{h}{}^{\#6}{}_{p}\Gamma_{j}{}^{\#1}{}_{k} \\
& - \Gamma_{\#1}{}^{\#6}{}_{h}\Gamma_{\#6}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#1}{}_{k} + \Gamma_{\#1}{}^{\#6}{}_{p}\Gamma_{\#6}{}^{\ell}{}_{h}\Gamma_{j}{}^{\#1}{}_{k} - \Gamma_{\#1}{}^{\ell}{}_{h,p}\Gamma_{j}{}^{\#1}{}_{k} + \Gamma_{\#1}{}^{\ell}{}_{k}\Gamma_{\#4}{}^{\#1}{}_{p}\Gamma_{j}{}^{\#4}{}_{h} \\
& - \Gamma_{\#25}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#25}{}_{h,p} - \Gamma_{\#19}{}^{\ell}{}_{p}\Gamma_{\#23}{}^{\#19}{}_{k}\Gamma_{j}{}^{\#23}{}_{h} + \Gamma_{\#2}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#2}{}_{h,k} - \Gamma_{\#19}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#19}{}_{h,k} \\
& + \Gamma_{\#19}{}^{\#22}{}_{k}\Gamma_{\#22}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#19}{}_{h} - \Gamma_{\#19}{}^{\#22}{}_{p}\Gamma_{\#22}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#19}{}_{h} - \Gamma_{\#19}{}^{\ell}{}_{p,k}\Gamma_{j}{}^{\#19}{}_{h} + \Gamma_{\#18}{}^{\ell}{}_{p,k}\Gamma_{j}{}^{\#18}{}_{h} \\
& + \Gamma_{\#10}{}^{\ell}{}_{k}\Gamma_{\#16}{}^{\#10}{}_{p}\Gamma_{j}{}^{\#16}{}_{h} - \Gamma_{\#10}{}^{\ell}{}_{p}\Gamma_{\#13}{}^{\#10}{}_{k}\Gamma_{j}{}^{\#13}{}_{h} - \Gamma_{\#11}{}^{\ell}{}_{k,p}\Gamma_{j}{}^{\#11}{}_{h} + \Gamma_{\#1}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#1}{}_{h,p} \\
& + \Gamma_{\#1}{}^{\#3}{}_{k}\Gamma_{\#3}{}^{\ell}{}_{p}\Gamma_{j}{}^{\#1}{}_{h} - \Gamma_{\#1}{}^{\#3}{}_{p}\Gamma_{\#3}{}^{\ell}{}_{k}\Gamma_{j}{}^{\#1}{}_{h} + \Gamma_{\#1}{}^{\ell}{}_{k,p}\Gamma_{j}{}^{\#1}{}_{h} - \Gamma_{\#1}{}^{\ell}{}_{k}\Gamma_{h}{}^{\#4}{}_{p}\Gamma_{j}{}^{\#1}{}_{\#4} \\
& + \Gamma_{h}{}^{\#25}{}_{k}\Gamma_{j}{}^{\ell}{}_{\#25,p} + \Gamma_{\#19}{}^{\ell}{}_{p}\Gamma_{h}{}^{\#23}{}_{k}\Gamma_{j}{}^{\#19}{}_{\#23} - \Gamma_{h}{}^{\#2}{}_{p}\Gamma_{j}{}^{\ell}{}_{\#2,k}
\end{aligned}
$$

- this sum contains 72 terms, each of which is a product of 2 or 3 Christoffel symbols, for a total of 180 Christoffel symbols
- however, upon simplifying this expression by consistently renaming the dummy indices, the simple result of zero is obtained, which verifies the identity

3.3.3 Expression Size

- Integer: (number of base β digits comprising n)

$$\mathcal{N}_\beta(n) \equiv \begin{cases} \lfloor \log_\beta |n| \rfloor + 1, & n \neq 0 \\ 0, & n = 0 \end{cases}$$

- Rational number:

$$\mathcal{N}_\beta\left(\frac{m}{n}\right) \equiv \mathcal{N}_\beta(m) + \mathcal{N}_\beta(n)$$

- General expression:
  - number of terms
  - number of operators and atomic operands
  - number of characters
  - etc.

3.3.4 Another Example of Intermediate Expression Swell

- stages in the expression swell analysis of the computation of the characteristic polynomial of a 5×5 general matrix containing "prime" 4-digit rational numbers (both the numerator and the denominator consist of 4 digits)

1. Initial matrix.

$$A = \begin{pmatrix} \frac{9533}{9539} & \frac{9547}{9551} & \frac{9587}{9601} & \frac{9613}{9619} & \frac{9623}{9629} \\ \frac{9631}{9643} & \frac{9649}{9661} & \frac{9677}{9679} & \frac{9689}{9697} & \frac{9719}{9721} \\ \frac{9733}{9739} & \frac{9743}{9749} & \frac{9767}{9769} & \frac{9781}{9787} & \frac{9791}{9803} \\ \frac{9811}{9817} & \frac{9829}{9833} & \frac{9839}{9851} & \frac{9857}{9859} & \frac{9871}{9883} \\ \frac{9887}{9901} & \frac{9907}{9923} & \frac{9929}{9931} & \frac{9941}{9949} & \frac{9967}{9973} \end{pmatrix}$$

2. The *sizes* of the entries in the upper Hessenberg matrix H that is similar to A. For example, the (5,4) entry is a rational number with a 309-digit numerator and a 313-digit denominator!

$$\mathcal{N}_{10}(H) = \begin{pmatrix} \frac{4}{4} & \frac{33}{32} & \frac{105}{104} & \frac{175}{175} & \frac{4}{4} \\ \frac{4}{4} & \frac{33}{32} & \frac{106}{105} & \frac{175}{175} & \frac{4}{4} \\ 0 & \frac{53}{56} & \frac{113}{117} & \frac{187}{190} & \frac{13}{16} \\ 0 & 0 & \frac{155}{158} & \frac{229}{231} & \frac{69}{72} \\ 0 & 0 & 0 & \frac{309}{313} & \frac{143}{146} \end{pmatrix}$$

3. The *sizes* of the coefficients of the characteristic polynomial $c_H(\lambda) = c_A(\lambda)$. For example, the coefficient of λ^3 is a rational number with a 98-digit numerator and a 100-digit denominator. The numbers involved involved have decreased in size, but are still large!

$$\mathcal{N}_{10}(c_H(\lambda)) = \lambda^5 + \frac{21}{20}\lambda^4 + \frac{98}{100}\lambda^3 + \frac{95}{100}\lambda^2 + \frac{92}{100}\lambda + \frac{88}{100}$$

3.3.5 Expression Swell Analysis

- bounds on expression size provide estimates of
  - memory needed
  - CPU time required

  to perform a given calculation.

- bounds can be determined
  - empirically (statistical survey)
  - theoretically (worst case and best case analyses)

Chapter 4

Basic possibilities of integrated mathematical systems

4.1 Axiom

- **inputs**

  `outputs`

4.1.1 Number domains

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

Big integers

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- integers of arbitrary size

  2312**

```
21914624432020321
                                          Type: PositiveInteger
```

factorial 60

```
832098711274139014427634118322336438075417260636124595244927769640960
0000000000000
                                          Type: PositiveInteger
```

bi:=234*37*59*101**

```
61700183203
                                          Type: PositiveInteger
```

- factorization of integers

  factor bi

```
   4
23 37 59 101
```

Type: Factored Integer

bia:=23*116**

```
40745903
```

Type: PositiveInteger

- integer greatest common divisor

  gcd(bi,bia)

```
23
```

Type: PositiveInteger

Rational numbers

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- exact calculation with rational numbers

  1234567890/98765432

```
617283945
---------
49382716
```

Type: Fraction Integer

rn:=1/2+2/15-64/47

```
  1027
- ----
  1410
```

Type: Fraction Integer

Complex numbers

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- exact calculation with complex numbers

  cn:= (2+3*%i)*(15-6*%i)+2/(2-4*%i)

```
63 + 115%i
----------
  2 + %i
```

Type: Fraction Complex Integer

- real and imaginary part

  real(cn) + %i*imag(cn)

```
    +---+
167\|- 1  + 241
---------------
       5
```

Type: Expression Integer

Algebraic numbers

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter ??) Mathematica (see chapter ??) Reduce (see chapter 4.6.1)

sqrt2:= rootOf(sqrt22-2)**

```
sqrt2
```

Type: AlgebraicNumber

1/(sqrt2+1)

```
    1
---------
sqrt2 + 1
```

Type: AlgebraicNumber

(x2+2*sqrt2*x+2)/(x+sqrt2)**

```
x + sqrt2
```

Type: Fraction Polynomial AlgebraicNumber

on gcd;

(x3+(sqrt2-2)*x**2-(2*sqrt2+3)*x-3*sqrt2)/(x**2-2);**

```
   2
  x  - 2x - 3
---------------
    - sqrt2 - 4
x + -----------
    2sqrt2 + 1
```

Type: Fraction Polynomial AlgebraicNumber

normalize

```
   2                   2
(2x  - 4x - 6)sqrt2 + x  - 2x - 3
---------------------------------
      (2x - 1)sqrt2 + x - 4
```

Type: Expression Integer

sqrt(x2-2*sqrt2*x*y+2*y**2)**

```
  +-----------------------+
  |                 2    2
 \|- 2x y sqrt2 + 2y  + x
                                                    Type: Expression Integer
```

- multiple algebraic extensions

  sqrt5:= rootOf(sqrt52-5)**

```
    sqrt5
                                                    Type: AlgebraicNumber
```

  cbrt3:= rootOf(cbrt33-3)**

```
    cbrt3
                                                    Type: AlgebraicNumber
```

  cbrt33**

```
    3
                                                    Type: AlgebraicNumber
```

  sqrt52;**

```
    5
                                                    Type: AlgebraicNumber
```

  cbrt3;

```
    cbrt3
                                                    Type: AlgebraicNumber
```

  sqrt(x2+2*(sqrt5-cbrt3)*x+5-2*sqrt5*cbrt3+cbrt3**2)**

```
     +--------------------------------------------------+
     |                           2               2
    \|(- 2cbrt3 + 2x)sqrt5 + cbrt3  - 2x cbrt3 + x  + 5
                                                    Type: Expression Integer
```

Big floating point numbers

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

rn:= - 1027/1410

```
  1027
- ----
  1410
```

Type: Fraction Integer

cn:=(167*%i + 241)/5

```
241 + 167%i
-----------
     5
```

Type: Fraction Complex Integer

- computation with floating point numbers

  rn :: Float

  ```
  - 0.7283687943 2624113475
  ```

  Type: Float

  cn :: Complex Float

  ```
  48.2 + 33.4 %i
  ```

  Type: Complex Float

  %pi :: Float

  ```
  3.1415926535 897932385
  ```

  Type: Float

  cos(%pi :: Float)

  ```
  - 1.0
  ```

  Type: Float

  sin 1.0

  ```
  0.8414709848 0789650665
  ```

  Type: Float

- computation with an arbitrary number of digits

  digits 50;

  Type: PositiveInteger

  %pi :: Float

```
3.1415926535 8979323846 2643383279 5028841971 693993751
                                                          Type: Float
```

cos %pi :: Float

```
- 1.0
                                                          Type: Float
```

- should be cos(pi/6) = sqrt(3)/2
  cos(%pi/6 :: Float)

```
0.8660254037 8443864676 3723170752 9361834714 0262690519
                                                          Type: Float
```

```
0.75
                                                          Type: Float
```

digits 20;

```
                                                Type: PositiveInteger
```

- no underflows appears
  exp(-100000.12)**

```
0.1184406313 2021703038 E -4342953504
                                                          Type: Float
```

- complex functions
  tan(1.0 + 1.0*%i)

```
0.2717525853 1951171653 + 1.0839233273 386945435 %i
                                                  Type: Complex Float
```

log(1.0 + 1.0*%i)

```
0.3465735902 7997265471 + 0.7853981633 9744830961 %i
                                                  Type: Complex Float
```

4.1.2 Polynomials

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

Basic operations

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- by default, parentheses are expanded

  pol:= (a+b+c)4**

```
   4             3       2              2  2      3         2       2      3
  c  + (4b + 4a)c  + (6b  + 12a b + 6a )c  + (4b  + 12a b  + 12a b + 4a )c
+
   4      3      2 2     3     4
  b  + 4a b  + 6a b  + 4a b + a
                                                   Type: Polynomial Integer
```

- differentiation

  dpol:= D(pol, a)

```
  3                2        2                2        3         2       2      3
4c  + (12b + 12a)c  + (12b  + 24a b + 12a )c + 4b  + 12a b  + 12a b + 4a
                                                   Type: Polynomial Integer
```

  D(D(pol, a), b, 2)

```
24c + 24b + 24a
                                                   Type: Polynomial Integer
```

- integration

  integrate(dpol, a)

```
    3                2  2          2       2      3         3     2 2     3     4
4a c  + (12a b + 6a )c  + (12a b  + 12a b + 4a )c + 4a b  + 6a b  + 4a b + a
                                          Type: Polynomial Fraction Integer
```

- polynomial greatest common divisor

  (a2-b**2)/(a**2-2*a*b+b**2)**

```
- b - a
-------
 b - a
                                          Type: Fraction Polynomial Integer
```

  g := 34*x19-91*x+70*x**7-25*x**16+20*x**3-86**

```
   19      16      7      3
34x   - 25x   + 70x  + 20x  - 91x - 86
                                                   Type: Polynomial Integer
```

f1:= g * (64*x34-21*x**47-126*x**8-46*x**5-16*x**60-81)**

```
         79        76         67         66        63         61         60         54
  - 544x   + 400x   - 1120x   - 714x   + 205x   + 1456x   + 1376x   - 1470x
 +
        53         50         48         47         41         37         35
  2176x   - 2020x   + 1911x   + 1806x   + 4480x   + 1280x   - 5824x
 +
          34         27         24         21         19         16         15
  - 5504x   - 4284x   + 1586x   + 1150x   - 2754x   + 2025x   - 8820x
 +
          12         11          9         8         7         6         5         3
  - 3220x   - 2520x   + 11466x  + 9916x  - 5670x  + 4186x  + 3956x  - 1620x
 +
  7371x + 6966
                                                       Type: Polynomial Integer
```

f2:= g * (72*x60-25*x**25-19*x**23-22*x**39-83*x**52+54*x**10+81)**

```
        79         76         71         68         67         63         61
  2448x   - 1800x   - 2822x   + 2075x   + 5040x   + 1440x   - 6552x
 +
          60         59        58         55         53         52         46
  - 6192x   - 5810x   - 748x   - 1110x   + 7553x   + 7138x   - 1540x
 +
         44         42        41         40         39         32         30
  - 850x   - 1086x   + 625x   + 2002x   + 2367x   - 1750x   - 1330x
 +
        29        28        26         25         24         23         19         17
  1836x   - 500x   + 545x   + 2150x   + 1729x   + 1634x   + 2754x   + 3780x
 +
          16         13         11         10         7         3
  - 2025x   + 1080x   - 4914x   - 4644x   + 5670x  + 1620x  - 7371x - 6966
                                                       Type: Polynomial Integer
```

gcd(f1,f2)

```
      19       16       7      3
  34x   - 25x   + 70x  + 20x  - 91x - 86
                                                       Type: Polynomial Integer
```

Factorization

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- factorization is the transformation of a polynomial into a product of polynomials

  factor(a2-b**2)**

```
  - (b - a)(b + a)
                                              Type: Factored Polynomial Integer
```

factor(a2+b**2, [rootOf(i^2 + 1)])**

```
         1        1
    (b - - a)(b + - a)
         i        i
                                        Type: Factored Polynomial AlgebraicNumber
```

fa:= (x2*z+y**4*z**2+5)* _**

(x*y3+z**2)* _**

(-x3*y+z**2+3)* _**

(x3*y**4+z**2)**

```
      4 8    2 7     3 8      7    3 5     4      6
     y z  + x z  + (x y  + x y  - x y  + 3y  + 5)z
   +
       5 4    3 3    5      2  5
     (x y  + x y  - x y + 3x )z
   +
       4 11    6 9       4     3  8       7     3 4       3     3        4
     (x y   - x y  + (- x  + 3x )y  + 3x y  + 5x y  + 5x y  - 5x y + 15)z
   +
       6 7    8 5       6     5  4     3 3  3
     (x y  - x y  + (- x  + 3x )y  + 3x y )z
   +
         7 12     4 11     4 7     6 5        4      3  4        3  2
     (- x y   + 3x y   + 5x y  - 5x y  + (- 5x  + 15x )y  + 15x y )z
   +
         9 8     6 7       7 8      4 7
     (- x y  + 3x y )z - 5x y  + 15x y
                                                     Type: Polynomial Integer
```

factor fa

```
      2    3        2      3   2    3 4   4 2    2
    (z  - x y + 3)(z  + x y )(z  + x y )(y z  + x z + 5)
                                                Type: Factored Polynomial Integer
```

Grobner bases

In Axiom

For comparison with other CAS choose from: Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

polys := [45*p + 35*s - 165*b - 36, _
35*p + 40*z + 25*t - 27*s, _
15*w + 25*p*s + 30*z - 18*t - 165*b2, _**
- 9*w + 15*p*t + 20*z*s, _
w*p + 2*z*t - 11*b3, _**
99*w - 11*s*b + 3*b2, _**
b2 + 33/50*b + 2673/10000]**

```
[35s + 45p - 165b - 36, 40z + 25t - 27s + 35p,
                                    2                                      3
 30z + 15w - 18t + 25p s - 165b , 20s z - 9w + 15p t, 2t z + p w - 11b ,
                  2   2   33     2673
 99w - 11b s + 3b , b  + -- b + -----]
                          50     10000
                                       Type: List Polynomial Fraction Integer
```

vars := [w, p, z, t, s, b]

```
 [w,p,z,t,s,b]
                                                           Type: List Symbol
```

groebner(polys)

```
     49     1143       19      1323       37     27        5      9
[z + -- b + ----, w + --- b + -----, t - -- b + ---, s - - b - ---,
     36     2000      120     20000      15     250       2     200
     31     153   2   33     2673
 p - -- b - ---, b  + -- b + -----]
     18     200       50     10000
                                       Type: List Polynomial Fraction Integer
```

- solving a system of polynomial equations by Grobner bases

  solve(polys, vars)

```
 [
      - 9500b - 3969     3100b + 1377     - 24500b - 10287     1850b - 81
   [w= --------------, p= ------------, z= ----------------, t= ----------,
          60000              1800              18000              750
      500b + 9        2
    s= --------, 10000b  + 6600b + 2673= 0]
         200
   ]
                             Type: List List Equation Fraction Polynomial Integer
```

- (see chapter4.1.4)

4.1.3 Rational functions

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.3) Macsyma (see chapter 4.3.3) Maple (see chapter 4.4.3) Mathematica (see chapter 4.5.3) Reduce (see chapter 4.6.3)

rf:= (3*a*b2-5*a**2*b)/(a**4-2)**

```
     2     2
 3a b  - 5a b
 ------------
     4
    a  - 2
                                          Type: Fraction Polynomial Integer
```

- integration

  integrate(rf, a)

```
          +-------------------------------------+
          |       2                       2     4
        (\|- 12%%G1  - 8%%G0 %%G1 - 12%%G0  + 9b   - 2%%G1 - 2%%G0)
     *
        log
               (24a %%G1 + 24a %%G0 - 60b - 50a)
            *
                +-------------------------------------+
                |       2                       2     4
               \|- 12%%G1  - 8%%G0 %%G1 - 12%%G0  + 9b
           +
                        2                                2
             48a %%G1  + (120b + 100a)%%G1 + 48a %%G0  + (120b + 100a)%%G0
           +
                       4
             - 27a b  - 250b
    +
            +-------------------------------------+
            |       2                       2     4
        (- \|- 12%%G1  - 8%%G0 %%G1 - 12%%G0  + 9b   - 2%%G1 - 2%%G0)
     *
        log
               (24a %%G1 + 24a %%G0 - 60b - 50a)
            *
                +-------------------------------------+
                |       2                       2     4
               \|- 12%%G1  - 8%%G0 %%G1 - 12%%G0  + 9b
           +
                          2                                  2
             - 48a %%G1  + (- 120b - 100a)%%G1 - 48a %%G0
           +
                                            4
             (- 120b - 100a)%%G0 + 27a b  + 250b
    +
                           2                             4
      4%%G1 log(96a %%G1  + (240b + 200a)%%G1 - 27a b  + 250b)
    +
                           2                             4
      4%%G0 log(96a %%G0  + (240b + 200a)%%G0 - 27a b  + 250b)
 /
    4
                                             Type: Union(Expression Integer,...)
```

- partial fraction decomposition

  (10*x2-11*x-6)/(x**3-x**2-2*x)**

```
      2
   10x  - 11x - 6
   --------------
     3    2
    x  - x  - 2x
```

Type: Fraction Polynomial Integer

padicFraction(_
partialFraction(numerator(
factor(denominator(
Factored UnivariatePolynomial(x, Fraction Integer)))

```
3     2       5
- + ----- + -----
x   x - 2   x + 1
            Type: PartialFraction UnivariatePolynomial(x,Fraction Integer)
```

4.1.4 Solving equations

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

Linear systems

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

solve([2*x1+x2+3*x3-9, x1-2*x2+x3+2, 3*x1+2*x2+2*x3-7], [x1, x2, x3])

```
[[x1= - 1,x2= 2,x3= 3]]
                    Type: List List Equation Fraction Polynomial Integer
```

Nonlinear equations

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

solve(x8-8*x**7+34*x**6-92*x**5+175*x**4-236*x**3+226*x**2-140*x+46, x)**

```
  8     7       6      5       4        3        2
[x  - 8x  + 34x  - 92x  + 175x  - 236x  + 226x  - 140x + 46= 0]
                         Type: List Equation Fraction Polynomial Integer
```

solve(log(acos(asin(x(2/3)-b)-1))+2, x)**

```
[]
                                  Type: List Equation Expression Integer
```

Nonlinear systems

In Axiom

For comparison with other CAS choose from: Derive (see chapter ??) Macsyma (see chapter ??) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

solve(_

[alpha * c1 - beta * c12 - gamma*c1*c2 + epsilon*c3, _**
-gamma*c1*c2 + (epsilon+theta)*c3 -eta *c2, _
gamma*c1*c2 + eta*c2 - (epsilon+theta) * c3], _
[c3, c2, c1])

```
>> Error detected within library code:
system does not have a finite number of solutions
You are being returned to the top level of the interpreter.
```

- (see chapter4.1.2)

4.1.5 Analytical operations

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

Limits

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

limit(sin(x)/x, x = 0)

```
  1
                              Type: Union(OrderedCompletion Expression Integer,...)
```

limit((3*sin(%pi*x) - sin(3*%pi*x))/x3, x = 0)**

```
      3
  4%pi
                              Type: Union(OrderedCompletion Expression Integer,...)
```

limit((2*x+5)/(3*x-2), x =

```
  2
  -
  3
                    Type: Union(OrderedCompletion Fraction Polynomial Integer,...)
```

Taylor series

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

)set streams calculate 4

series(%ex, x = 0)**

```
          1  2   1  3   1   4      5
  1 + x + - x  + - x  + -- x  + O(x )
          2      6      24
                              Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
```

)set streams calculate 2

```
              2      3
1 + 2x + 2x  + O(x )
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
```

Summation and Products

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

sum(n2*x**n, n)**

```
   2          3        2           2    2   n - 1
((n  - 2n + 1)x  + (- 2n  + 2n + 1)x  + n x)x
--------------------------------------------------
                 3     2
                x  - 3x  + 3x - 1
                                                     Type: Expression Integer
```

sum(cos((2*r-1)*%pi/(2*n+1)), r)

```
 r
--+      (2%A - 1)%pi
>    cos(------------)
--+         2n + 1
%A
                                                     Type: Expression Integer
```

product(%e(sin(n*x)), n)**

```
  n
++-++    sin(%A x)
 | |   %e
 | |
 %A
                                                     Type: Expression Integer
```

for all n,m such that fixp m let factorial(n+m)=if m ¿ 0 then factorial(n+m-1)*(n+m) else factorial(n+m+1)/(n+m+1);

sum(n*2n/factorial(n+2), n)**

```
 n        %A
--+   %A 2
>    ---------
--+  (%A + 2)!
%A
                                                     Type: Expression Integer
```

Integration

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

integrate(x2*(a+b*x)**p, x)**

```
   3 2     3      3  3       2 2      2   2     2          3   p log(b x + a)
((b p  + 3b p + 2b )x  + (a b p  + a b p)x  - 2a b p x + 2a )%e
                          3 3     3 2      3      3
                         b p  + 6b p  + 11b p + 6b
                                          Type: Union(Expression Integer,...)
```

integrate(x2*log(x**2+a**2), x)**

```
      3     2    2      3       x      3     2
    3x log(x  + a ) + 6a atan(- -) - 2x  + 6a x
                                a
    ------------------------------------------
                         9
                                          Type: Union(Expression Integer,...)
```

integrate(x*dx*sin x, x)**

```
                3         2                                        2
         (x log(d)  - log(d)  + x log(d) + 1)sin(x) - x cos(x)log(d)
       +
         2cos(x)log(d) - x cos(x)
    *
         x log(d)
       %e
  /
           4           2
     log(d)  + 2log(d)  + 1
                                          Type: Union(Expression Integer,...)
```

integrate(x*sqrt(a+b*x)p, x)**

```
                                                     +-------+
        2      2  2                 2   p log(\|b x + a )
    ((2b p + 4b )x  + 2a b p x - 4a )%e
    ---------------------------------------------------
                     2 2     2      2
                    b p  + 6b p + 8b
                                          Type: Union(Expression Integer,...)
```

integrate(2*x*%e(x**2)*log(x)+%e**(x**2)/x+(log(x)-2)/(log(x)**2+x)**2+ _
((2/x)*log(x)+(1/x)+1)/(log(x)**2+x), x)**

```
                                             2                2
          2               2           x      3        x
    (log(x)  + x)log(log(x)  + x) + %e  log(x)  + (x %e   - 1)log(x)
    ----------------------------------------------------------------
```

```
                    2
             log(x)  + x
                          Type: Union(Expression Integer,...)
```

Ordinary differential equations

In Axiom

For comparison with other CAS choose from: Derive (see chapter ??) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

y:= operator('y);

```
                                                  Type: BasicOperator
```

solve(D(y(x), x) + y(x) * sin x/cos x - 1/cos x = 0, y, x)

```
 [particular= sin(x),basis= [cos(x)]]
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
```

- Bernoulli equation

  solve(x*(1-x2)*D(y(x), x) + (2*x**2 -1)*y(x) - x**3*y(x)**3 = 0, y, x)**

```
       5    2     4     2
  - 2x y(x)  + 5x  - 5x
  ----------------------
                2
         5y(x)
                                     Type: Union(Expression Integer,...)
```

  solve(D(y(x), x, 2)+4*D(y(x), x)+4*y(x)-x*exp(x) = 0, y, x)

```
                             - 2x   x 3
                  (3x - 2)%e    (%e )             - 2x      - 2x
  [particular= --------------------,basis= [%e     ,x %e     ]]
                            27
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
```

Substitutions - pattern matching

In Axiom

For comparison with other CAS choose from: Derive (see chapter ??) Macsyma (see chapter 4.3.5) Maple (see chapter ??) Mathematica (see chapter ??) Reduce (see chapter 4.6.5)

sincosRules:= rule _
(cos(x)*cos(y) == (cos(x+y) + cos(x-y))/2; _
cos(x)*sin(y) == (sin(x+y) - sin(x-y))/2; _
sin(x)*sin(y) == (cos(x-y) - cos(x+y))/2; _
cos(x)2 == (1 + cos(2*x))/2; _**
sin(x)2 == (1 - cos(2*x))/2)**

```
                       %X cos(y + x) + %X cos(- y + x)
{%X cos(x)cos(y) == -------------------------------,
                                    2
```

```
                          %Y sin(y + x) - %Y sin(- y + x)
    %Y cos(x)sin(y) == ------------------------------,
                                       2
                          - %Z cos(y + x) + %Z cos(- y + x)
    %Z sin(x)sin(y) == --------------------------------,
                                        2
          2    cos(2x) + 1        2    - cos(2x) + 1
    cos(x)  == -----------, sin(x)  == -------------}
                    2                        2
                               Type: Ruleset(Integer,Integer,Expression Integer)
```

sincosRules (a1*cos(wt) + a3*cos(3*wt) + b1*sin(wt) + b3*sin(3*wt))3**

```
       3        3          2                2                  2                2
     b3 sin(3wt)  + (3b1 b3 sin(wt) + 3a3 b3 cos(3wt) + 3a1 b3 cos(wt))sin(3wt)
   +
             2          2
         3b1 b3 sin(wt)  + (6a3 b1 b3 cos(3wt) + 6a1 b1 b3 cos(wt))sin(wt)
       +
             2           2                                 2         2
         3a3 b3 cos(3wt)  + 6a1 a3 b3 cos(wt)cos(3wt) + 3a1 b3 cos(wt)
    *
       sin(3wt)
   +
       3      3          2                2              2
     b1 sin(wt)  + (3a3 b1 cos(3wt) + 3a1 b1 cos(wt))sin(wt)
   +
         2          2                              2         2
     (3a3 b1 cos(3wt)  + 6a1 a3 b1 cos(wt)cos(3wt) + 3a1 b1 cos(wt) )sin(wt)
   +
       3        3         2                2      2          2
     a3 cos(3wt)  + 3a1 a3 cos(wt)cos(3wt)  + 3a1 a3 cos(wt) cos(3wt)
   +
       3      3
     a1 cos(wt)
                                                    Type: Expression Integer
```

int:= operator('int);

```
                                                        Type: BasicOperator
```

intRules:= rule _
(int(x + :y, z) == int(x, z) + int(y, z); _
int(k*x — freeOf?(k, z), z) == k*int(x, z); _
int(y — integer? y, z) == y*z; _
int(x(?p — D(p, x) = 0), x) == x**(p+1)/(p+1))**

```
   {int(y + x,z) == 'int(y,z) + 'int(x,z), int(k x,z) == k'int(x,z),
                                    p + 1
                           p       x
    int(y,z) == y z, int(x ,x) == ------}
                                   p + 1
                               Type: Ruleset(Integer,Integer,Expression Integer)
```

intRules int(a2*b+a**b+3*a-5, a)**

```
 b + 1     2     2               2             2
a      + (a b + a )int(b,a) + (3a  - 5a)b + 3a  - 5a
----------------------------------------------------
                       b + 1
```

Type: Expression Integer

intRules int(a(a+1), a)**

```
 a + 2
a
------
a + 2
```

Type: Expression Integer

4.1.6 Matrices

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.6) Macsyma (see chapter 4.3.6) Maple (see chapter 4.4.6) Mathematica (see chapter 4.5.6) Reduce (see chapter 4.6.6)

xx:= matrix([[a11, a12], [a21, a22]])

```
+a11  a12+
|        |
+a21  a22+
```

Type: Matrix Polynomial Integer

yy:= matrix([[y1], [y2]])

```
+y1+
|  |
+y2+
```

Type: Matrix Polynomial Integer

determinant xx

```
a11 a22 - a12 a21
```

Type: Polynomial Integer

zz:= inverse(xx)*yy

```
+- a12 y2 + a22 y1+
|-----------------|
|a11 a22 - a12 a21|
|                 |
| a11 y2 - a21 y1 |
|-----------------|
+a11 a22 - a12 a21+
```

Type: Matrix Fraction Polynomial Integer

inverse(xx)2**

```
   [
                    2
                 a22  + a12 a21
     [--------------------------------------,
         2   2                       2   2
      a11 a22  - 2a11 a12 a21 a22 + a12 a21
               - a12 a22 - a11 a12
      --------------------------------------]
         2   2                       2   2
      a11 a22  - 2a11 a12 a21 a22 + a12 a21
     ,

               - a21 a22 - a11 a21
     [--------------------------------------,
         2   2                       2   2
      a11 a22  - 2a11 a12 a21 a22 + a12 a21
                                   2
                  a12 a21 + a11
      --------------------------------------]
         2   2                       2   2
      a11 a22  - 2a11 a12 a21 a22 + a12 a21
     ]
                                   Type: Matrix Fraction Polynomial Integer
```

v := matrix([[2, -1, 1], [0, 1, 1], [-1, 1, 1]])

```
    + 2   - 1  1+
    |          |
    | 0    1   1|
    |          |
    +- 1   1   1+
                                                     Type: Matrix Integer
```

eigenvectors v

```
                                  +0+                                 +1+
                                  | |                                 | |
   [[eigval= 2,eigmult= 1,eigvec= [|1|]],[eigval= 1,eigmult= 2,eigvec= [|1|]]]
                                  | |                                 | |
                                  +1+                                 +0+
Type: List Union(Record(algrel: Fraction Polynomial Integer,algmult: Integer,algvec: List Matrix Fraction Po
```

4.1.7 Graphics

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

2D Graphics

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- function of one parameter

  draw(sin(%e^x), x = 0..%pi)

- graph of several functions (Bessel functions $J(n, x), n = 0, 2, 5$)

  draw([besselJ(0, x), besselJ(2, x), besselJ(5, x)], x = 0..10, _

  title == "Bessel Functions BesselJ(n, x)")

3D Graphics

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- graph of a function with 2 parameters

  draw(sin(%pi*sin(x+y)), x = -3..3, y = -3..3)

- graph of another function with 2 parameters

  draw(tan(x*y), x = -2/3*%pi..2/3*%pi, y = -2/3*%pi..2/3*%pi)

Parametric plots

In Axiom

For comparison with other CAS choose from: Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- surface defined parametrically

  draw(surface(sin(v), sin(2*v)*sin(u), sin(2*v)*cos(u)), _

  u = 0..2*%pi, v = -%pi/2..%pi/2)

4.2 Derive

- **inputs**

  `outputs`

4.2.1 Number domains

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

Big integers

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- integers of arbitrary size

  23^12

```
  12
23
```

```
21914624432020321
```

60!

```
60!
```

```
832098711274139014427634118322336438075417260636124595244927769640960
000000000000
```

23^4 37 59 101

```
  4
23  37 59 101
```

```
61700183203
```

- factorization of integers

  factor

```
  4
23  37 59 101
```

Rational numbers

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- precise calculation with rational numbers

  1234567890/98765432

```
 1234567890
------------
  98765432
```

```
 617283945
-----------
 49382716
```

1/2+2/15-64/47

```
 1     2      64
--- + ---- - ----
 2     15     47
```

```
   1027
- ------
   1410
```

Complex numbers

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- precise calculation with complex numbers

  (2 + 3 #i) (15 - 6 #i) + 2/(2 - 4 #i)

```
                              2
(2 + 3 i) (15 - 6 i) + ---------
                          2 - 4 i


 241     167 i
----- + -------
  5        5
```

Radicals

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter ??) Mathematica (see chapter ??) Reduce (see chapter 4.6.1)

- computation with radicals

  (x^2+2 sqrt(2) x+2)/(x+sqrt(2))

```
  2
 x  + 2  2 x + 2
-----------------
     x +  2


x +  2
```

  sqrt(x^2-2 sqrt(2) x y+2 y^2)

```
   2                 2
 (x  - 2  2 x y + 2 y )


  2  2  2 x - 4 y
-------------------
        4
```

Big floating point numbers

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- computation with floating point numbers

  - 1027/1410

```
   1027
- ------
   1410
```

```
-0.7283687943
```

(167 #i + 241)/5

```
 241     167 i
----- + -------
  5        5
```

```
48.2 + 33.4 i
```

pi

```
3.141592920
```

cos(pi)

```
COS ( )
```

```
-1
```

sin(1)

```
SIN (1)
```

```
0.8414709137
```

- computation with arbitrary number of digits
  options, precision, digits 50
  pi

```
3.1415926535897932384626433832795028841971693993751
```

- should be cos(pi/6) = sqrt(3)/2
  cos(pi/6)

```
COS  ---
      6
```

```
0.8660254037844386467637231707529361834714026269051
```

#2^2

```
                                                    2
0.8660254037844386467637231707529361834714026269051
```

```
0.75
```

- complex functions

  options, precision, digits 10

  tan(1.0 + 1.0 #i)

  ```
  TAN (1 + 1 i)
  ```

  ```
  0.2717525893 + 1.083923333 i
  ```

  log(1.0 + 1.0 #i)

  ```
  LOG (1 + 1 i)
  ```

  ```
  0.3465735900 + 0.7853981634 i
  ```

4.2.2 Polynomials

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

Basic operations

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- by default parantheses are not expanded

  p := (a+b+c)^4

  ```
                    4
  P := (a + b + c)
  ```

expand

$$a^4 + 4a^3 b + 4a^3 c + 6a^2 b^2 + 12a^2 b c + 6a^2 c^2 + 4a b^3 + 12a b^2 c$$

$$+ 12 a b c^2 + 4 a c^3 + b^4 + 4 b^3 c + 6 b^2 c^2 + 4 b c^3 + c^4$$

- derivation

d := dif(p,a)

$$D := \frac{d}{da} P$$

$$4a^3 + 12a^2 (b + c) + 12 a (b^2 + 2 b c + c^2) + 4 b^3 + 12 b^2 c + 12 b c^2$$

$$+ 4 c^3$$

$$\frac{d}{db} \frac{d}{db} \frac{d}{da} a^4 + 4a^3 b + 4a^3 c + 6a^2 b^2 + 12a^2 b c + 6a^2 c^2 + 4a b^3$$

$$+ 12 a b^2 c + 12 a b c^2 + 4 a c^3 + b^4 + 4 b^3 c + 6 b^2 c^2 + 4 b c^3 + c^4$$

$$24 a + 24 b + 24 c$$

- integration

i := int(d,a)

$$I := \int D \, da$$

$$a^4 + 4a^3 (b + c) + 6a^2 (b^2 + 2 b c + c^2) + a (4 b^3 + 12 b^2 c + 12 b c^2$$

$$+ 4 c^3)$$

- verification

  i - p

```
I - P


   4       3         2  2         3    4
- (b  + 4 b  c + 6 b  c  + 4 b c  + c )
```

- greatest common divisor of polynomials

  (a^2-b^2)/(a^2-2 a b+b^2)

```
      2    2
     a  - b
-----------------
  2            2
 a  - 2 a b + b


 a + b
-------
 a - b
```

  g := 34 x^19-91 x+70 x^7-25 x^16+20 x^3-86

  f := g (64 x^34-21 x^47-126 x^8-46 x^5-16 x^60-81)

  h := g (72 x^60-25 x^25-19 x^23-22 x^39-83 x^52+54 x^10+81)

  f/h insufficient memory

Factorization

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- factorization is transformation of a polynomial into a product of polynomials

  a^2-b^2

```
 2    2
a  - b
```

  factor

```
(a - b) (a + b)
```

  a^2+b^2

```
 2    2
a  + b
```

factor, complex

```
(a - i b) (a + i b)
```

(x^2 z+y^4 z^2+5) (-x^3 y+z^2+3)

```
   3      2         2      4  2
(- x  y + z  + 3) (x  z + y  z  + 5)
```

expand

```
  5         3  5  2      3      2  3       2      4  4      4  2      2
- x  y z - x  y  z  - 5 x  y + x  z  + 3 x  z + y  z  + 3 y  z  + 5 z  +

 15
```

factor

```
    2      4  2         3       2
 - (x  z + y  z  + 5) (x  y - (z  + 3))
```

for bigger expressions was insufficient memory

4.2.3 Rational functions

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.3) Macsyma (see chapter 4.3.3) Maple (see chapter 4.4.3) Mathematica (see chapter 4.5.3) Reduce (see chapter 4.6.3)

- integration

  w := int((3 a b^2-5 a^2 b)/(a^4-2),a)

```
               2      2
          3 a b  - 5 a  b
W :=     ----------------- da
               4
              a  - 2
```

```
                                                          2
                        3/4               2         2 a  - 2              3/4
          3/4         a 2            3  2 b  LN  -----------       5 b 2    LN
   5 b 2    ATAN  --------                                2
                       2                            2 a  + 2
- -------------------------- + -------------------------- + -------------
               4                            8                            8
```

```
               1/4
```

```
  a + 2
 ----------
       1/4
  a - 2
-------------
```

- verification by derivation

  dif(w,a);

```
d
--  W
da
```

```
            2 a b (5 a - 3 b)
- --------------------------------------
        2             1/4            3/4
   ( 2 a  + 2) (a + 2    ) ( 2 a - 2    )
```

by using build the denominator is chosen

```
     2             1/4            3/4
( 2 a  + 2) (a + 2    ) ( 2 a - 2    )
```

```
   4
2 a  - 4
```

final result

```
 a b (5 a - 3 b)
-----------------
     4
    a  - 2
```

4.2.4 Solving equations

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

Linear systems

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

[2 x + y + 3 z - 9, x - 2 y + z + 2, 3 x + 2 y + 2 z - 7]

```
[2 x + y + 3 z - 9, x - 2 y + z + 2, 3 x + 2 y + 2 z - 7]
```

```
[x = -1, y = 2, z = 3]
```

Nonlinear equations

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

- solving polynomial equations

  x^3 + 5 x^2 - x + 2

```
 3      2
x  + 5 x  - x + 2
```

```
         3777     349  1/3     349      3777  1/3    5
x = -  ------- + -----     -  ----- - -------     - ---
         18       54           54       18           3
```

```
       3777     349  1/3     349      3777  1/3    5         77897
x =  ------- + -----     +  ----- - -------     - --- + i   -------
       144      432          432      144          3         3456

           349  3777  1/6     349  3777     77897  1/6
        - -----------     -  ----------- + -------
             1152               1152        3456
```

```
       3777     349  1/3     349      3777  1/3    5
x =  ------- + -----     +  ----- - -------     - --- + i
       144      432          432      144          3

        349  3777     77897  1/6     77897     349  3777  1/6
       ----------- + -------      - ------- - -----------
          1152        3456           3456        1152
```

- multiple use of inversion functions

  LOG(ACOS(ASIN(x^(2/3) - b) - 1))

```
                  2/3
LOG (ACOS (ASIN (x    - b) - 1))
```

```
 1/3
x    = -  (SIN (COS (1) + 1) + b)
```

```
 1/3
x    =  (SIN (COS (1) + 1) + b)
```

4.2.5 Analytical operations

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

Limits

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

lim(sin(x)/x,x,0)

```
      SIN (x)
lim  ---------
x->0     x


1
```

lim((3 sin(pi x) - sin(3 pi x))/x^3,x,0)

```
      3 SIN (  x) - SIN (3   x)
lim  ---------------------------
x->0             3
                x


   3
4
```

lim((2x+5)/(3x-2),x,inf)

```
      2 x + 5
lim  ---------
x->   3 x - 2


 2
---
 3
```

Taylor series

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

taylor (#e^x, x, 0, 4)

```
         x
TAYLOR (e  , x, 0, 4)


  4      3      2
 x      x      x
---- + ---- + ---- + x + 1
 24     6      2
```

taylor(taylor(#e^(x+y), x, 0, 2), y, 0, 2)

```
                 x + y
TAYLOR (TAYLOR (e     , x, 0, 2), y, 0, 2)


   2              2
 (y  + 2 y + 2) (x  + 2 x + 2)
-------------------------------
               4
```

Summation and Products

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

sum(m^2 x^m,m,1,n);

```
 n   2  m
    m  x
m=1


    2  2         2                      2   n
 x (n  x  - x (2 n  + 2 n - 1) + (n + 1) ) x      x (x + 1)
----------------------------------------------- - -----------
                           3                               3
                    (x - 1)                         (x - 1)
```

sum(cos((2 m-1) pi/(2 n+1)),m,1,r)

```
 r        (2 m - 1)
    COS  -------------
m=1         2 n + 1


       2 r
  SIN  ---------
        2 n + 1
-------------------

 2 SIN  ---------
         2 n + 1
```

product(#e^(sin(m x)),m,1,n)

```
 n   SIN (m x)
    e
m=1


 COT (x) / 2 + 1 / (2 SIN (x)) - COS (x (n + 1/2)) / (2 SIN (x/2))
e
```

Integration

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

int(x d^x sin(x),x)

```
      x
 x d   SIN (x) dx
```

```
                      x                x                   2  x
 x SIN (x) LN (d) d      x COS (x) d      SIN (x) LN (d)   d
--------------------- - -------------- - -------------------- +
           2                  2                    2     2
     LN (d)  + 1        LN (d)  + 1         (LN (d)  + 1)


                      x                x
      2 COS (x) LN (d) d        SIN (x) d
     --------------------- + ----------------
                2     2             2     2
        (LN (d)  + 1)         (LN (d)  + 1)
```

4.2.6 Matrices

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.6) Macsyma (see chapter 4.3.6) Maple (see chapter 4.4.6) Mathematica (see chapter 4.5.6) Reduce (see chapter 4.6.6)

x := [[a,b],[c,d]]

```
       a  b
X :=
       c  d
```

y := [[u],[v]]

```
       u
Y :=
       v
```

det(x)

```
DET (X)


a d - b c
```

z := x^(-1) . y

```
      -1
Z := X     Y
```

```
  d u - b v
-----------
  a d - b c

  a v - c u
-----------
  a d - b c
```

1/x^2

```
 1
----
  2
 X
```

```
                 2
          b c + d                           b (a + d)
  ---------------------------   - ---------------------------
    2  2                2  2        2  2                2  2
   a  d  - 2 a b c d + b  c        a  d  - 2 a b c d + b  c

                                            2
           c (a + d)                       a  + b c
- ---------------------------   ---------------------------
    2  2                2  2      2  2                2  2
   a  d  - 2 a b c d + b  c      a  d  - 2 a b c d + b  c
```

eigenvalues([[2,-1,1],[0,1,1],[-1,1,1]])

```
                2  -1  1

EIGENVALUES     0   1  1  , e

               -1   1  1


[e = 1, e = 2]
```

4.2.7 Graphics

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

2D Graphics

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- function of one parameter

  sin(exp(x)) plot, setting scale and center, plot

- graph of several functions (Bessel functions $J(n, x), n = 0, 2, 5)$

  transfer, load, derive, bessel

  BESSEL_J(0,z)

  BESSEL_J(2,z)

  BESSEL_J(5,z) plot, setting scale a center, plot

3D Graphics

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- graph of a function with 2 parameters

  SIN(pi SIN(x)+y)

- graph of another function with 2 parameters

  TAN(x y)

Parametric plots

In Derive

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- parametricaly given surface

  transfer, load, derive, graphics

  ISOMETRICS([SIN(v), SIN(2 v) SIN(u), SIN(2 v) COS(u)], u,0,2 pi,20,v,-pi/2,pi/2,10)

  ISOMETRICS([SIN(v), SIN(2 v) SIN(u), SIN(2 v) COS(u)], v,-pi/2,pi/2,10,u,0,2 pi,20)

4.3 Macsyma

- **inputs**

  `outputs`

4.3.1 Number domains

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

Big integers

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- integers of arbitrary size

  23^12;

```
21914624432020321
```

  60!;

```
832098711274139014427634118322336438075417260636124595244927769640960
000000000000
```

bi: 23^4*37*59*101;

```
61700183203
```

- factorization of integers

  factor(bi);

```
  4
23   37  59  101
```

bia: 23*11^6;

```
40745903
```

- integer greatest common divisor

  gcd(bi,bia);

```
23
```

Rational numbers

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- exact calculation with rational numbers

  1234567890/98765432;

```
617283945
---------
49382716
```

rn: 1/2+2/15-64/47;

```
  1027
- ----
  1410
```

Complex numbers

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- exact calculation with complex numbers

  cn: (2+3*%i)*(15-6*%i)+2/(2-4*%i);

```
                                   2
(15 - 6 %i) (3 %i + 2) + --------
                                2 - 4 %i
```

  cn: rectform(cn);

```
167 %i   241
------ + ---
  5       5
```

Algebraic numbers

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter **??**) Mathematica (see chapter **??**) Reduce (see chapter 4.6.1)

- an algebraic number

  algebraic:true;

```
true
```

  tellrat(sqrt22-2);**

```
2
        [sqrt2  - 2]
```

  rat(1/(sqrt2+1));

```
/R/     sqrt2 - 1
```

  ogcd:gcd;

```
spmod
```

  gcd:'algebraic;

```
algebraic
```

rat((x2+2*sqrt2*x+2)/(x+sqrt2));**

```
/R/     x + sqrt2
```

rat((x3+(sqrt2-2)*x**2-(2*sqrt2+3)*x-3*sqrt2)/(x**2-2));**

```
  2
 x  - 2 x - 3
   /R/    ------------
  x - sqrt2
```

gcd:ogcd;

```
spmod
```

radcan(sqrt(x2-2*sqrt2*x*y+2*y**2));**

```
 2     2
      sqrt(2 y  - 2 sqrt2 x y + x )
```

untellrat(sqrt2);

```
[]
```

- multiple algebraic extensions

tellrat(sqrt52-5,cbrt3**3-3);**

```
  3       2
         [cbrt3  - 3, sqrt5  - 5]
```

rat(cbrt33);**

```
/R/         3
```

rat(sqrt52);**

```
/R/         5
```

rat(cbrt3);

```
/R/       cbrt3
```

rat(sqrt5);

```
/R/       sqrt5
```

radcan(sqrt(x2+2*(sqrt5-cbrt3)*x+5-2*sqrt5*cbrt3+cbrt3**2));**

```
  2         2
 sqrt(x  + (2 sqrt5 - 2 cbrt3) x - 2 cbrt3 sqrt5 + cbrt3  + 5)
```

untellrat(sqrt5,cbrt3);

```
[]
```

Big floating point numbers

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

rn: - 1027/1410;

```
  1027
- ----
  1410
```

cn: (167*i + 241)/5;

```
167 %i + 241
------------
     5
```

- computation with floating point numbers

  bfloat(rn);

```
- 7.2836879432624113475b-1
```

bfloat(cn);

```
3.34b1 %i + 4.82b1
```

bfloat(%pi);

```
3.1415926535897932385b0
```

cos(bfloat(%pi));

```
- 1.0b0
```

sin(1.0b0);

```
8.4147098480789650665b-1
```

- computation with an arbitrary number of digits
  fpprec: 50$
  bfloat(%pi);

```
3.1415926535897932384626433832795028841971693993751b0
```

cos(bfloat(%pi));

```
- 1.0b0
```

- should be cos(pi/6) = sqrt(3)/2
  cos(bfloat(%pi/6));

```
8.6602540378443864676372317075293618347140262690519b-1
```

%^2;

```
7.5b-1
```

fpprec: 20$

- with normal defaults, underflows are converted to 0
  exp(-100000.12);**

```
0.0
```

- complex functions
  bfloat(rectform(tan(1.0b0 + 1.0b0*%i)));

```
1.0839233273386945435b0 %i + 2.7175258531951171653b-1
```

bfloat(log(1.0b0 + 1.0b0*%i));

```
7.8539816339744830962b-1 %i + 3.4657359027997265471b-1
```

4.3.2 Polynomials

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

Basic operations

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- by default, parentheses are not expanded

  pol: (a+b+c)^4;

  $$(c + b + a)^4$$

  pol: expand(pol);

  $$c^4 + 4\,b\,c^3 + 4\,a\,c^3 + 6\,b^2\,c^2 + 12\,a\,b\,c^2 + 6\,a^2\,c^2 + 4\,b^3\,c + 12\,a\,b^2\,c + 12\,a^2\,b\,c + 4\,a^3\,c + b^4 + 4\,a\,b^3 + 6\,a^2\,b^2 + 4\,a^3\,b + a^4$$

- differentiation

  dpol: diff(pol, a);

  $$4\,c^3 + 12\,b\,c^2 + 12\,a\,c^2 + 12\,b^2\,c + 24\,a\,b\,c + 12\,a^2\,c + 4\,b^3 + 12\,a\,b^2 + 12\,a^2\,b + 4\,a^3$$

  diff(pol, a, 1, b, 2);

  $$24\,c + 24\,b + 24\,a$$

- integration

  integrate(dpol, a);

  $$4\,a\,c^3 + 12\,a\,b\,c^2 + 6\,a^2\,c^2 + 12\,a\,b^2\,c + 12\,a^2\,b\,c + 4\,a^3\,c + 4\,a\,b^3 + 6\,a^2\,b^2 + 4\,a^3\,b + a^4$$

- verification

  %-pol;

```
    4         3      2  2      3      4
 - c  - 4 b c  - 6 b  c  - 4 b  c - b
```

- polynomial greatest common divisor

  (a^2-b^2)/(a^2-2*a*b+b^2);

```
          2    2
         a  - b
     ---------------
      2            2
     b  - 2 a b + a
```

ratsimp(%);

```
          b + a
        - -----
          b - a
```

g : 34*x^19-91*x+70*x^7-25*x^16+20*x^3-86;

```
       19        16         7        3
   34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86
```

f1 : g * (64*x^34-21*x^47-126*x^8-46*x^5-16*x^60-81);

```
      19        16         7        3
  (34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86)

                              60        47        34         8        5
                         (- 16 x   - 21 x   + 64 x   - 126 x  - 46 x  - 81)
```

f2 : g * (72*x^60-25*x^25-19*x^23-22*x^39-83*x^52+54*x^10+81);

```
      19        16         7        3
  (34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86)

                     60        52        39        25        23        10
                (72 x   - 83 x   - 22 x   - 25 x   - 19 x   + 54 x   + 81)
```

gcd(f1,f2);

```
       19        16         7        3
   34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86
```

Factorization

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- factorization is the transformation of a polynomial into a product of polynomials

  factor(a^2-b^2);

```
- (b - a) (b + a)
```

  gfactor(a^2+b^2);

```
(b - %i a) (b + %i a)
```

  fa: expand((x^2*z+y^4*z^2+5)* (x*y^3+z^2)* (-x^3*y+z^2+3)* (x^3*y^4+z^2));

```
  4  8    2  7    3  8  6      7  6    3  5  6      4  6      6    5  4  5
 y  z  + x  z  + x  y  z  + x y  z  - x  y  z  + 3 y  z  + 5 z  + x  y  z

  3  3  5    5    5      2  5    4  11  4    6  9  4    4  8  4      3  8  4
+ x  y  z  - x  y z  + 3 x  z  + x  y   z  - x  y  z  - x  y  z  + 3 x  y  z

      7  4      3  4  4        3  4      3    4       4    6  7  3    8  5  3
+ 3 x y  z  + 5 x  y  z  + 5 x y  z  - 5 x  y z  + 15 z  + x  y  z  - x  y  z

  6  4  3      5  4  3      3  3  3    7  12  2      4  11  2      4  7  2
- x  y  z  + 3 x  y  z  + 3 x  y  z  - x  y   z  + 3 x  y   z  + 5 x  y  z

    6  5  2      4  4  2       3  4  2          3  2    9  8        6  7
- 5 x  y  z  - 5 x  y  z  + 15 x  y  z  + 15 x y  z  - x  y  z + 3 x  y  z

    7  8       4  7
- 5 x  y  + 15 x  y
```

  factor(fa);

```
  2    3          2      3    2    3  4    4  2    2
(z  - x  y + 3) (z  + x y ) (z  + x  y ) (y  z  + x  z + 5)
```

Decomposition

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- decomposing a polynomial into simpler polynomials which when composed, produce the original polynomial

  polydecomp(x^6+9*x^5+52*x^4+177*x^3+435*x^2+630*x+593, x);

```
                                      2
          3        2                 x  - 9
        [x  + 25 x  + 210 x + 593, ------, 2 x + 3]
                                       4
```

polydecomp(x^4+2*x^3*y + 3*x^2*y^2 + 2*x*y^3 + y^4 + 2*x^2*y +2*x*y^2 + 2*y^3 + 5*x^2 + 5*x*y + 6*y^2 + 5*y + 9, x);

```
            2           2           2
           x  + 11  3 y  + 4 y + x  + 10
          [-------, --------------------, y + 2 x]
              4              2
```

Grobner bases

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter **??**) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- system of polynomials

  polys : [45*p + 35*s - 165*b - 36, 35*p + 40*z + 25*t - 27*s, 15*w + 25*p*s + 30*z - 18*t - 165*b^2, - 9*w + 15*p*t + 20*z*s, w*p + 2*z*t - 11*b^3, 99*w - 11*s*b + 3*b^2, b^2 + 33/50*b + 2673/10000];

```
      [35 s + 45 p - 165 b - 36, 40 z + 25 t - 27 s + 35 p,

   2
30 z + 15 w - 18 t + 25 p s - 165 b , 20 s z - 9 w + 15 p t,

  3          2   2   33 b   2673
2 t z + p w - 11 b , 99 w - 11 b s + 3 b , b  + ---- + -----]
 50    10000
```

  vars : [w, p, z, t, s, b];

```
              [w, p, z, t, s, b]
```

- total degree ordering **grobner_tot_order:true;**

```
             true
```

  gpolys:grobner(polys,vars);

```
     /R/ [- 500 b + 200 s - 9, 1850 b - 750 t - 81, - 9500 b - 60000 w - 3969,

      2
       10000 b  + 6600 b + 2673
- 24500 b - 18000 z - 10287, - 3100 b + 1800 p - 1377, ------------------------]
10000
```

- solve the Grobner basis **solve(gpolys, vars);**

```
        190 sqrt(11) %i + 139  62 sqrt(11) %i + 59
        [[w = - --------------------, p = ------------------,
        10000  300

      490 sqrt(11) %i + 367  148 sqrt(11) %i - 461
z = - ---------------------, t = --------------------,
      3000   500

    15 sqrt(11) %i - 39      12 sqrt(11) %i - 33
s = -------------------, b = -------------------],
    50      100

     190 sqrt(11) %i - 139   62 sqrt(11) %i - 59
[w = ---------------------, p = - -------------------,
     10000   300

    490 sqrt(11) %i - 367  148 sqrt(11) %i + 461
z = ---------------------, t = - ---------------------,
    3000   500

      15 sqrt(11) %i + 39  12 sqrt(11) %i + 33
s = - -------------------, b = - -------------------]]
      50  100
```

4.3.3 Rational functions

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.3) Derive (see chapter 4.2.3) Maple (see chapter 4.4.3) Mathematica (see chapter 4.5.3) Reduce (see chapter 4.6.3)

rf: (3*a*b^2-5*a^2*b)/(a^4-2);

```
       2      2
 3 a b  - 5 a  b
---------------
      4
     a  - 2
```

- integration

  integrate(rf, a);

```
                                                                      1/4
                                                                 a - 2
                                                           5 log(--------)
              2                       2                               1/4
       3 log(a  + sqrt(2)) b   3 log(a  - sqrt(2)) b           a + 2
  - b (--------------------- - --------------------- + ---------------
             4 sqrt(2)               4 sqrt(2)                1/4
                                                           4 2
                                                                    a
                                                           5 atan(----)
                                                                   1/4
                                                                  2
```

```
                                                       + ------------)
                                                               1/4
                                                            2 2
```

- partial fraction decomposition

  partfrac((10*x^2-11*x-6)/(x^3-x^2-2*x), x);

```
  5       3     2
----- + - + -----
x + 1   x   x - 2
```

4.3.4 Solving equations

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

Linear systems

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

solve([2*x1+x2+3*x3-9, x1-2*x2+x3+2, 3*x1+2*x2+2*x3-7], [x1, x2, x3]);

```
[[x1 = - 1, x2 = 2, x3 = 3]]
```

Nonlinear equations

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

- using decomposition to solve high degree polynomials

  solve(x^8-8*x^7+34*x^6-92*x^5+175*x^4-236*x^3+226*x^2-140*x+46, x);

```
       sqrt(- sqrt(4 sqrt(3) + 3) %i - 3) - sqrt(2)
[x = - --------------------------------------------,
                         sqrt(2)

    sqrt(- sqrt(4 sqrt(3) + 3) %i - 3) + sqrt(2)
x = --------------------------------------------,
                      sqrt(2)

      sqrt(sqrt(4 sqrt(3) + 3) %i - 3) - sqrt(2)
x = - ------------------------------------------,
                       sqrt(2)

    sqrt(sqrt(4 sqrt(3) + 3) %i - 3) + sqrt(2)
x = ------------------------------------------,
                     sqrt(2)

      sqrt(sqrt(4 sqrt(3) - 3) + 3) %i - sqrt(2)
x = - ------------------------------------------,
```

```
                              sqrt(2)

         sqrt(sqrt(4 sqrt(3) - 3) + 3) %i + sqrt(2)
     x = -----------------------------------------,
                            sqrt(2)

           sqrt(3 - sqrt(4 sqrt(3) - 3)) %i - sqrt(2)
     x = - ------------------------------------------,
                              sqrt(2)

         sqrt(3 - sqrt(4 sqrt(3) - 3)) %i + sqrt(2)
     x = ------------------------------------------]
                            sqrt(2)
```

- multiple use of inversion functions

  assume(b ¿ 0);

```
                                   [b > 0]
```

solve(log(acos(asin(x^(2/3)-b)-1))+2, x);

```
                        - 2        3/2   1/3                       - 2
    [x = (b + sin(cos(%e    ) + 1))    , x    = - sqrt(b + sin(cos(%e    ) + 1))]
```

forget(b ¿ 0)$

4.3.5 Analytical operations

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

Limits

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

limit(sin(x)/x, x, 0);

```
                                 1
```

limit((3*sin(%pi*x) - sin(3*%pi*x))/x^3, x, 0);

```
                                   3
                               4 %pi
```

limit((2*x+5)/(3*x-2), x, %inf);

```
                              2 %inf + 5
                              ----------
                              3 %inf - 2
```

Taylor series

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

taylor(%e^x, x, 0, 4);

```
                                   2    3    4
                                  x    x    x
     /T/                  1 + x + -- + -- + -- + . . .
                                  2    6    24
```

taylor(%e^(x+y), x, 0, 2, y, 0, 2);

```
                2             2                        2
               y             y                        y    y   1          2
   /T/ 1 + y + -- + . . . + (-- + y + 1 + . . .) x + (-- + - + - + . . .) x
               2             2                        4    2   2

                                                                   + . . .
```

%^2;

```
                      2               2
     /T/ 1 + 2 y + 2 y  + . . . + (4 y  + 4 y + 2 + . . .) x

                                               2                    2
                                       + (4 y  + 4 y + 2 + . . .) x  + . . .
```

Summation and Products

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

load(nusum1)$

closedform(sum(m^2*x^m, m, 1, n));

```
         n + 1   2  2          2                 2
        x      (n  x  + (- 2 n  - 2 n + 1) x + n  + 2 n + 1)   x (x + 1)
        ---------------------------------------------------- - ---------
                                      3                            3
                               (x - 1)                      (x - 1)
```

closedform(sum(cos((2*m-1)*%pi/(2*n+1)), m, 1, r));

```
                            r
                           ====
                           \         %pi (2 m - 1)
                            >    cos(-------------)
                           /            2 n + 1
                           ====
                           m = 1
```

closedform(product(%e^(sin(m*x)), m, 1, n));

```
                                                       %i n x + %i x
                               %i                 %i %e
      expt(%e, - ------------------------------ - ------------------
                     %i n x + %i x       %i n x          %i x
                 2 %e              - 2 %e            2 %e     - 2

                                                     %i x
                                                %i %e          %i
                                             + ------------ + ------------)
                                                   %i x           %i x
                                               2 %e     - 2   2 %e     - 2
```

closedform(sum(m*2^m/(m+2)!, m, 1, n));

```
                                      n
                                   2 2
                             1 - --------
                                 (n + 2)!
```

Integration

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

integrate(x^2*(a+b*x)^p, x);

```
  3   2             3      2   2       2      2            3    p log(b x + a)
(b  (p  + 3 p + 2) x  + a b  (p  + p) x  - 2 a  b p x + 2 a ) %e
                           3   3      2
                          b  (p  + 6 p  + 11 p + 6)
```

integrate(x^2*log(x^2+a^2), x);

```
                                                    3      2
                                       3      x    x  - 3 a  x
                  3      2    2    2 (a  atan(-) + -----------)
                 x  log(x  + a )              a         3
                 --------------- - ----------------------------
                        3                       3
```

integrate(x*d^x*sin(x), x);

```
       3                     2          log(d) x
 (((log (d) + log(d)) x - log (d) + 1) %e         sin(x)

          2                         log(d) x             4           2
 + ((- log (d) - 1) x + 2 log(d)) %e         cos(x))/(log (d) + 2 log (d) + 1)
```

integrate(x*sqrt(a+b*x)^p, x);

```
                                                   p log(b x + a)
                                                   --------------
             2            2                  2          2
            (b  (2 p + 4) x  + 2 a b p x - 4 a ) %e
            ----------------------------------------------------
                                2   2
                               b  (p  + 6 p + 8)
```

radcan(%);

```
                 p/2      2        2   2                  2
            (b x + a)    ((2 b  p + 4 b ) x  + 2 a b p x - 4 a )
            ---------------------------------------------------
                              2  2      2        2
                             b  p  + 6 b  p + 8 b
```

integrate(2*x*%e^(x^2)*log(x)+%e^(x^2)/x+(log(x)-2)/(log(x)^2+x)^2+ ((2/x)*log(x)+(1/x)+1)/(log(x)^2+x), x);

```
                                                     2
                         2             log(x)       x
                log(log (x) + x) - ----------- + %e   log(x)
                                        2
                                    log (x) + x
```

Ordinary differential equations

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter **??**) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

ode('diff(y, x) + y * sin(x)/cos(x) - 1/cos(x), y, x);

```
                      y = cos(x) (tan(x) + %c)
```

- Bernoulli equation

  ode(x*(1-x^2)*'diff(y, x) + (2*x^2 -1)*y - x^3*y^3, y, x);

```
                                    log(x + 1)   log(x - 1)
                                    ---------- + ----------
                                        2            2
                                x %e
                 y = ------------------------------------------
                                   3             5      3
                                  x  + 3 x   3 x  + 5 x  + 15 x
                     sqrt(%c - 2 (-------- - ------------------))
                                     3               15
```

  radcan(%);

```
     sqrt(5) sqrt(x - 1) x sqrt(x + 1)
y = ---------------------------------
                    5
          sqrt(2 x  + 5 %c)
```

ode('diff(y, x, 2)+4*'diff(y, x)+4*y-x*exp(x), y, x);

```
                    x
        (3 x - 2) %e                      - 2 x
y = ------------- + (%k2 x + %k1) %e
             27
```

Substitutions - pattern matching

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter **??**) Maple (see chapter **??**) Mathematica (see chapter **??**) Reduce (see chapter 4.6.5)

matchdeclare([x, y], true)$

letsimp((cos(x)*cos(y), (cos(x+y) + cos(x-y))/2));

```
cos(y + x) + cos(y - x)
-----------------------
           2
```

letsimp((cos(x)*sin(y), (sin(x+y) - sin(x-y))/2));

```
sin(y + x) + sin(y - x)
-----------------------
           2
```

letsimp((sin(x)*sin(y), (cos(x-y) - cos(x+y))/2));

```
cos(y - x) - cos(y + x)
-----------------------
           2
```

letsimp((cos(x)^2, (1 + cos(2*x))/2));

```
cos(2 x) + 1
------------
     2
```

letsimp((sin(x)^2, (1 - cos(2*x))/2));

```
1 - cos(2 x)
------------
     2
```

letsimp(expand((a1*cos(wt) + a3*cos(3*wt) + b1*sin(wt) + b3*sin(3*wt))^3));

```
       3    3                2              2
     b3  sin (3 wt) + 3 a3 b3  cos(3 wt) sin (3 wt)

         2            2                2            2
+ 3 b1 b3  sin(wt) sin (3 wt) + 3 a1 b3  cos(wt) sin (3 wt)

      2       2
+ 3 a3  b3 cos (3 wt) sin(3 wt) + 6 a3 b1 b3 sin(wt) cos(3 wt) sin(3 wt)

                                               2       2
+ 6 a1 a3 b3 cos(wt) cos(3 wt) sin(3 wt) + 3 b1  b3 sin (wt) sin(3 wt)

                                             2       2
+ 6 a1 b1 b3 cos(wt) sin(wt) sin(3 wt) + 3 a1  b3 cos (wt) sin(3 wt)

    3    3             2               2                2            2
+ a3  cos (3 wt) + 3 a3  b1 sin(wt) cos (3 wt) + 3 a1 a3  cos(wt) cos (3 wt)

         2    2
+ 3 a3 b1  sin (wt) cos(3 wt) + 6 a1 a3 b1 cos(wt) sin(wt) cos(3 wt)

      2       2                   3    3              2            2
+ 3 a1  a3 cos (wt) cos(3 wt) + b1  sin (wt) + 3 a1 b1  cos(wt) sin (wt)

      2       2                 3    3
+ 3 a1  b1 cos (wt) sin(wt) + a1  cos (wt)
```

declare(int, linear)$
matchdeclare(p, is(diff(p, x) = 0))$
tellsimp(int(x^p, x), x^(p+1)/(p+1));

```
[intrule1, simpargs1]
```

tellsimp(int(1, x), x);

```
[intrule2, intrule1, simpargs1]
```

int(a^2*b+a^b+3*a-5, a);

```
 b + 1    3        2
a        a  b   3 a
------ + ---- + ---- - 5 a
b + 1     3      2
```

int(a^(a+1), a);

```
 a + 2
a
------
a + 2
```

4.3.6 Matrices

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.6) Derive (see chapter 4.2.6) Maple (see chapter 4.4.6) Mathematica (see chapter 4.5.6) Reduce (see chapter 4.6.6)

xx: matrix([a11, a12], [a21, a22]);

```
[ a11  a12 ]
[          ]
[ a21  a22 ]
```

yy: matrix([y1], [y2]);

```
[ y1 ]
[    ]
[ y2 ]
```

determinant(xx);

```
a11 a22 - a12 a21
```

zz: xx^^(-1).yy;

```
[      a22 y1               a12 y2       ]
[ ----------------- - ----------------- ]
[ a11 a22 - a12 a21   a11 a22 - a12 a21 ]
[                                       ]
[      a11 y2               a21 y1       ]
[ ----------------- - ----------------- ]
[ a11 a22 - a12 a21   a11 a22 - a12 a21 ]
```

xx^^(-2);

```
        [              2                                 ]
        [           a22                   a12 a21        ]
        [  -------------------- + --------------------  ]
        [                     2                      2  ]
Col 1 = [  (a11 a22 - a12 a21)    (a11 a22 - a12 a21)   ]
        [                                                ]
        [         a21 a22                  a11 a21       ]
        [ - -------------------- - -------------------- ]
        [                      2                      2 ]
        [   (a11 a22 - a12 a21)    (a11 a22 - a12 a21)  ]

        [         a12 a22                  a11 a12       ]
        [ - -------------------- - -------------------- ]
        [                      2                      2 ]
        [   (a11 a22 - a12 a21)    (a11 a22 - a12 a21)  ]
Col 2 = [                                                ]
        [                                     2          ]
        [        a12 a21                   a11           ]
        [  -------------------- + --------------------  ]
        [                     2                      2  ]
```

```
                  [  (a11 a22 - a12 a21)    (a11 a22 - a12 a21)   ]
```

factor(%);

```
[        2                                        ]
[     a22  + a12 a21          a12 (a22 + a11)     ]
[  --------------------   - -------------------- ]
[                      2                        2 ]
[  (a11 a22 - a12 a21)      (a11 a22 - a12 a21)  ]
[                                                 ]
[                                          2      ]
[     a21 (a22 + a11)         a12 a21 + a11       ]
[ - --------------------   --------------------  ]
[                        2                     2  ]
[   (a11 a22 - a12 a21)    (a11 a22 - a12 a21)   ]
```

v : matrix([2, -1, 1], [0, 1, 1], [-1, 1, 1]);

```
[  2   - 1  1 ]
[             ]
[  0    1   1 ]
[             ]
[ - 1   1   1 ]
```

eigenvectors(v);

```
[[2, 1, [[0, 1, 1]]], [1, 2, [[1, 1, 0]]]]
```

4.3.7 Code generation

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Maple (see chapter ??) Mathematica (see chapter ??) Reduce (see chapter 4.6.7)

- package gentran

  gentranout ("gentst_mac.f");

```
gentst_mac.f
```

m : matrix([18*COS(Q3)*COS(Q2)*M30*P^2 - 9*SIN(Q3)^2*P^2*M30 - SIN(Q3)^2*J30Y + SIN(Q3)^2*J30Z + P^2*M10 + 18*P^2*M30 + J10Y + J30Y, 9*COS(Q3)*COS(Q2)*M30*P^2 - SIN(Q3)^2*J30Y + SIN(Q3)^2*J30Z - 9*SIN(Q3)^2*M30*P^2 + J30Y + 9*M30*P^2, -9*SIN(Q3)*SIN(Q2)],

[9*COS(Q3)*COS(Q2)*M30*P^2 - SIN(Q3)^2*J30Y + SIN(Q3)^2*J30Z - 9*SIN(Q3)^2*M30*P^2 + J30Y + 9*M30*P^2, -SIN(Q3)^2*J30Y + SIN(Q3)^2*J30Z - 9*SIN(Q3)^2*M30*P^2 + J30Y + 9*M30*P^2, 0],

[-9*SIN(Q3)*SIN(Q2)*M30*P^2, 0, 9*M30*P^2 + J30X]);

```
  2    2     2      2
matrix([- 9 m30 p  sin (q3) + j30z sin (q3) - j30y sin (q3)
```

```
   2         2         2
 + 18 m30 p  cos(q2) cos(q3) + 18 m30 p  + m10 p  + j30y + j10y,

 2    2        2        2      2
- 9 m30 p  sin (q3) + j30z sin (q3) - j30y sin (q3) + 9 m30 p  cos(q2) cos(q3)

  2      2
 + 9 m30 p  + j30y, - 9 m30 p  sin(q2) sin(q3)],

  2    2        2        2       2
[- 9 m30 p  sin (q3) + j30z sin (q3) - j30y sin (q3) + 9 m30 p  cos(q2) cos(q3)

  2      2    2    2    2
 + 9 m30 p  + j30y, - 9 m30 p  sin (q3) + j30z sin (q3) - j30y sin (q3)

  2   2         2
 + 9 m30 p  + j30y, 0], [- 9 m30 p  sin(q2) sin(q3), 0, 9 m30 p  + j30x])
```

- we know that matrix m is symmetric. We wish to generate FORTRAN code to compute numerical values for matrix m, and its inverse matrix, minv.

  mm : copymatrix(m)$

  gentran(literal("C", cr, "C — Calculate Matrix Values —", cr, "C", cr));

  ```
  gentst_mac.f
  ```

  for i:1 thru 3 do for j:i thru 3 do gentran(m[eval(i),eval(j)] : eval(m[i,j]))$

  gentran(literal("C", cr, "C — Assign Non-Zero Matrix Values to Temporary ", "Variables —", cr, "C", cr));

  ```
  gentst_mac.f
  ```

  for i:1 thru 3 do for j:i thru 3 do if m[i,j]#0 then (var : tempvar(false), markvar(var), m[i,j] : var, m[j,i] : var, gentran(eval(var) : m[eval(i),eval(j)]))$

- matrix m contains **m;**

```
[ t0  t1  t2 ]
[            ]
       [ t1  t3  0  ]
[            ]
[ t2  0   t4 ]
```

minv : m^^(-1)$

gentran(literal("C", cr, "C — Calculate Inverse Matrix Values —", cr, "C", cr));

```
gentst_mac.f
```

for i:1 thru 3 do for j:i thru 3 do gentran(minv[eval(i),eval(j)] : eval(minv[i,j]))$

gentran(literal("C", cr, "C — Copy Entries Across Main Diagonals —", cr, "C", cr), for i:1 thru 3 do for j:i+1 thru 3 do (m[j,i] : m[i,j], minv[j,i] : minv[i,j]));

```
gentst_mac.f
```

gentranshut ("gentst_mac.f");

```
true
```

Generated FORTRAN program

Generated program

```
C
C --- Calculate Matrix Values ---
C
      M(1,1)=-(9*M30*P**2*SIN(Q3)**2)+J30Z*SIN(Q3)**2-(J30Y*SIN(Q3)**2)+
     . 18*M30*P**2*COS(Q2)*COS(Q3)+18*M30*P**2+M10*P**2+J30Y+J10Y
      M(1,2)=-(9*M30*P**2*SIN(Q3)**2)+J30Z*SIN(Q3)**2-(J30Y*SIN(Q3)**2)+
     . 9*M30*P**2*COS(Q2)*COS(Q3)+9*M30*P**2+J30Y
      M(1,3)=-(9*M30*P**2*SIN(Q2)*SIN(Q3))
      M(2,2)=-(9*M30*P**2*SIN(Q3)**2)+J30Z*SIN(Q3)**2-(J30Y*SIN(Q3)**2)+
     . 9*M30*P**2+J30Y
      M(2,3)=0
      M(3,3)=9*M30*P**2+J30X
C
C --- Assign Non-Zero Matrix Values to Temporary Variables ---
C
      T0=M(1,1)
      T1=M(1,2)
      T2=M(1,3)
      T3=M(2,2)
      T4=M(3,3)
C
C --- Calculate Inverse Matrix Values ---
C
      MINV(1,1)=(T3*T4)/((T0*T3-T1**2)*T4-(T2**2*T3))
```

```
      MINV(1,2)=-((T1*T4)/((T0*T3-T1**2)*T4-(T2**2*T3)))
      MINV(1,3)=-((T2*T3)/((T0*T3-T1**2)*T4-(T2**2*T3)))
      MINV(2,2)=(T0*T4-T2**2)/((T0*T3-T1**2)*T4-(T2**2*T3))
      MINV(2,3)=(T1*T2)/((T0*T3-T1**2)*T4-(T2**2*T3))
      MINV(3,3)=(T0*T3-T1**2)/((T0*T3-T1**2)*T4-(T2**2*T3))
C
C --- Copy Entries Across Main Diagonals ---
C
      DO 25001 I=1,3
          DO 25002 J=I+1,3
              M(J,I)=M(I,J)
              MINV(J,I)=MINV(I,J)
25002     CONTINUE
25001 CONTINUE
```

4.3.8 Graphics

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

2D Graphics

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- function of one parameter

  plot(sin(%e^x), x, 0, %pi, "Time", "Signal", false);

- graph of several functions (Bessel functions $J(n, x), n = 0, 2, 5$)

  plot([bessel_j[0](x), bessel_j[2](x), bessel_j[5](x)], x, 0, 10, false, "Value", "Bessel Functions BesselJ(n, x)");

3D Graphics

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- graph of a function with 2 parameters

  plot3d(sin(%pi*sin(x+y)), x, -3, 3, y, -3, 3);

- graph of another function with 2 parameters

  plot3d(tan(x*y), x, -2/3*%pi, 2/3*%pi, y, -2/3*%pi, 2/3*%pi);

Parametric plots

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- surface defined parametrically

  plotsurf([[sin(v), sin(2*v)*sin(u), sin(2*v)*cos(u)]], u, 0, 2*%pi, v, -%pi/2, %pi/2);

Contour maps

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- contour map of a function with 2 parameters

  contourplot(sin(x + cos(y)), x, 0, 3/2*%pi, y, 0, 3/2*%pi);

- contour map of another function with 2 parameters

  contourplot(tan(x*y), x, -%pi, %pi, y, -%pi, %pi);

- contour map of yet another function with 2 parameters

  contourplot(%e^(-sqrt(x^2 + y^2))*cos(atan(x/y)), x, -1, 1, y, -1, 1);

4.3.9 Graphical presentation of formulas

In Macsyma

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Maple (see chapter 4.4.8) Mathematica (see chapter 4.5.8) Reduce (see chapter 4.6.9)

4.4 Maple

- **inputs**

  `outputs`

4.4.1 Number domains

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

Big integers

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- integers of arbitrary size

  23^12;

```
21914624432020321
```

  60!;

```
83209871127413901442763411832233643807541726063612459524492776964096\
0000000000000
```

  bi := 23^4*37*59*101;

```
bi := 61700183203
```

- factorization of integers

  ifactor(bi);

```
            4
       (23)  (37) (59) (101)
```

  bia := 23*11^6;

```
       bia := 40745903
```

- greatest common divisor of integers

  gcd(bi, bia);

```
       23
```

Rational numbers

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- precise calculation with rational numbers

  1234567890 / 98765432;

```
       617283945
       ---------
        49382716
```

  rn := 1/2+2/15-64/47;

```
                1027
       rn := - ----
                1410
```

Complex numbers

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

- precise calculation with complex numbers

  cn := (2+3*I)*(15-6*I)+2/(2-4*I);

```
       cn := 241/5 + 167/5 I
```

Big floating point numbers

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Mathematica (see chapter 4.5.1) Reduce (see chapter 4.6.1)

rn := - 1027/1410;

```
               1027
     rn := - ----
               1410
```

cn := (167*I + 241)/5;

```
cn := 241/5 + 167/5 I
```

- computation with floating point numbers

  evalf(rn);

```
-.7283687943
```

  evalf(cn);

```
48.20000000 + 33.40000000 I
```

  Pi;

```
Pi
```

  evalf(Pi);

```
3.141592654
```

  evalf(cos(Pi));

```
-1.
```

  sin(1.);

- computation with arbitrary number of digits

  evalf(rn, 20);

```
-.72836879432624113475
```

Digits := 50;

```
Digits := 50
```

evalf(Pi);

```
3.1415926535897932384626433832795028841971693993751
```

- should be cos(pi/6) = sqrt(3)/2
  evalf(cos(Pi/ 6), 50);

```
0.86602540378443864676372317075293618347140262690520
```

"^2;

```
0.7500000000
```

Digits := 10;

```
Digits := 10
```

- underflow gives an error
  exp(-100000.12);**

```
Error, (in evalf/exp/general) argument too large
```

- complex functions
  tan(1.0 + I);

```
0.2717525853 + 1.083923327 I
```

log(1.0 + I);

```
0.3465735903 + .7853981634 I
```

4.4.2 Polynomials

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

Basic operations

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- by default parantheses are not expanded

  pol := (a+b+c)^4;

$$pol := (a + b + c)^4$$

  expand(pol);

$$12\,a^2\,b\,c + 12\,a\,b^2\,c + 12\,a\,b\,c^2 + a^4 + b^4 + c^4 + 4\,a^3\,b + 4\,a^3\,c + 6\,a^2\,b^2 + 6\,a^2\,c^2 + 4\,a\,b^3 + 4\,a\,c^3 + 4\,b^3\,c + 6\,b^2\,c^2 + 4\,b\,c^3$$

- derivation

  dpol := diff(expand(pol), a);

$$dpol := 24\,a\,b\,c + 12\,b^2\,c + 12\,b\,c^2 + 4\,a^3 + 12\,a^2\,b + 12\,a^2\,c + 12\,a\,b^2 + 12\,a\,c^2 + 4\,b^3 + 4\,c^3$$

  diff(pol, a, b, b);

$$24\,a + 24\,b + 24\,c$$

- integration

  int(dpol, a);

$$12\,a^2\,b\,c + 12\,a\,b^2\,c + 12\,a\,b\,c^2 + a^4 + 4\,a^3\,b + 4\,a^3\,c + 6\,a^2\,b^2 + 6\,a^2\,c^2 + 4\,a\,b^3 + 4\,a\,c^3$$

- verification

  simplify(" - pol);

$$-b^4 - c^4 - 4\,b^3\,c - 6\,b^2\,c^2 - 4\,b\,c^3$$

- greatest common divisor of polynomials

(a^2-b^2)/(a^2-2*a*b + b^2);

```
                         2    2
                        a  - b
                     ---------------
                      2            2
                     a  - 2 a b + b
```

simplify(");

```
                         a + b
                         -----
                         a - b
```

g := 34*x^19 - 25*x^16 + 70*x^7 + 20*x^3 - 91*x - 86;

```
                        19        16        7        3
              g := 34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86
```

f1 := expand(g*(64*x^34 - 21*x^47 - 126*x^8 - 46*x^5 - 16*x^60 - 81));

```
                          7          16          19          3
    f1 := 7371 x - 5670 x  + 2025 x   - 2754 x   - 1620 x  + 6966

              34          47          8          5          60          53
         - 5504 x   + 1806 x   + 9916 x  + 3956 x  + 1376 x   + 2176 x

             66          27          24         79          50         63
         - 714 x   - 4284 x   + 1586 x   - 544 x   - 2020 x   + 205 x

              21         76          41          54          15          12
         + 1150 x   + 400 x   + 4480 x   - 1470 x   - 8820 x   - 3220 x

              67          37          11          35          48           9
         - 1120 x   + 1280 x   - 2520 x   - 5824 x   + 1911 x   + 11466 x

              6          61
         + 4186 x  + 1456 x
```

f2 := expand(g*(72*x^60 - 25*x^25 - 19*x^23 - 22*x^39 - 83*x^52 + 54*x^10 + 81));

```
                            7          16          19          3          60
    f2 := - 7371 x + 5670 x  - 2025 x   + 2754 x   + 1620 x  - 6192 x

              53          24          79          63          76         41
         + 7553 x   + 1729 x   + 2448 x   + 1440 x   - 1800 x   + 625 x

              67          11          61          25          23          39
         + 5040 x   - 4914 x   - 6552 x   + 2150 x   + 1634 x   + 2367 x
```

```
         52          10         44          42         58          71
+ 7138 x   - 4644 x   - 850 x   - 1086 x   - 748 x   - 2822 x

         29          55          68         26          32          30
+ 1836 x   - 1110 x   + 2075 x   + 545 x   - 1750 x   - 1330 x

         46          59          17         28          13          40
- 1540 x   - 5810 x   + 3780 x   - 500 x   + 1080 x   + 2002 x

- 6966
```

gcd(f1, f2);

```
     19       16       7       3
34 x   - 25 x   + 70 x  + 20 x  - 91 x - 86
```

Factorization

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- factorization is transformation of a polynomial into a product of polynomials

  factor(a^2 - b^2);

```
(a - b) (a + b)
```

  factor(a^2 + b^2);

```
 2    2
a  + b
```

  factor(a^2 + b^2, I);

```
(a - I b) (a + I b)
```

  fa := expand((x^2* z + y^4* z^2 + 5) * (x* y^3 + z^2)

- **(-x^3* y + z^2 + 3) * (x^3* y^4 + z^2));**

```
         3  3  3    4  8    7  6      11  4  4    8  4  4    12  2  7
fa := 3 x  z  y  + y  z  + y  z  x + y   z  x  - y  z  x  - y   z  x

       5  3  4      2  5    5  5  4    5  5      8  3  5
  + 3 x  z  y  + 3 x  z  + x  z  y  - x  z  y - x  z  y

       6    7    3  5  3    6  3  7    6  3  4    9    8
  + 3 x  z y  + x  z  y  + x  z  y  - x  z  y  - x  z y
```

```
          3 2       7  4       5  6  3    9  4  6      11  2  4
+ 15 x y   z  + 3 y  z  x - y  z  x  - y  z  x  + 3 y   z  x

       4  7       7  8       2  3  4      4  3  4      4  3
+ 15 x  y  - 5 x  y  + 15 z  x  y  + 5 z  x  y  - 5 z  x  y

      2  6  5         3  4      4  7  2      4  4  2      8  4  3
- 5 z  x  y  + 5 x y  z  + 5 x  y  z  - 5 x  y  z  + 3 y  z  x

    8  6  3      6       4    2  7      4  6
+ y  z  x  + 5 z  + 15 z  + x  z  + 3 y  z
```

factor(fa);

```
     2      4  2        3  4    2      3    2      2        3
- (x  z + y  z  + 5) (x  y  + z ) (x y  + z ) (- z  - 3 + x  y)
```

Decomposition

In Maple

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- finding of polynomial substitution which transforms a polynomial into another polynomial in further polynomials

  readlib(compoly);

  compoly(x^6 + 9*x^5 + 52*x^4 + 177*x^3 + 435*x^2 + 630*x + 593, x);

```
 3    11                              2
x  + ---- + 5/3 x, x = 25/3 + 3 x + x
      27
```

  compoly(x^4 + 2*x^3* y + 3*x^2* y^2 + 2*x* y^3 + y^4 + 2*x^2* y + 2*x* y^2 + 2*y^3 + 5*x^2 + 5*x*y + 6*y^2 + 5*y + 9, x);

```
 2              2                      2
x  + 11/4, x = y  + y + 5/2 + y x + x
```

Grobner bases

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Mathematica (see chapter 4.5.2) Reduce (see chapter 4.6.2)

- moduls groebner and gbasis dealing with Grobner bases

  with(grobner, gbasis);

```
[gbasis]
```

polys := [45*p + 35*s - 165*b - 36, 35*p + 40*z + 25*t - 27*s, 15*w + 25*p*s + 30*z - 18*t - 165*b^2, -9*w + 15*p*t + 20*z*s, w*p + 2*z*t - 11*b^3, 99*w - 11*s*b + 3*b^2, b^2 + 33/50*b + 2673/10000] ;

```
      polys := [45 p + 35 s - 165 b - 36, 35 p + 40 z + 25 t - 27 s,

                                                  2
          15 w + 25 p s + 30 z - 18 t - 165 b , - 9 w + 15 p t + 20 z s,

                                3                      2    2     33        2673
          w p + 2 z t - 11 b , 99 w - 11 s b + 3 b , b  + ---- b + -----]
                                                                  50       10000
```

vars := [w, p, z, t, s, b];

```
                           vars := [w, p, z, t, s, b]
```

gbasis(polys, vars, tdeg);

```
        [1800 p - 3100 b - 1377, 18000 z + 24500 b + 10287,

                                                    2
            60000 w + 9500 b + 3969, 10000 b  + 6600 b + 2673,

            750 t - 1850 b + 81, - 500 b - 9 + 200 s]
```

4.4.3 Rational functions

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.3) Derive (see chapter 4.2.3) Macsyma (see chapter 4.3.3) Mathematica (see chapter 4.5.3) Reduce (see chapter 4.6.3)

- integration

  int((3*a*b^2 - 5*a^2* b) / (a^4 - 2), a);

```
                                       2  1/2
            2  1/2     - 2 + a  2                3/4                 3/4
       3/8 b  2    ln(-------------) - 5/4 b 2    arctan(1/2 a 2   )
                                       2  1/2
                      - 2 - a  2

                                   1/4
                        3/4     a + 2
            + 5/8 b 2    ln(--------)
                                   1/4
                            a - 2
```

- verification by derivation

  diff(", a);

```
           /        1/2                 2  1/2     1/2\
  2  1/2 |      a 2           (- 2 + a  2   ) a 2   |          2  1/2
b  2     |2 ------------- + 2 ---------------------| (- 2 - a  2   )
         |          2  1/2                2  1/2 2   |
         \  - 2 - a  2           (- 2 - a  2   )    /
                                     2  1/2
                              - 2 + a  2

                  1/2
               b 2
- 5/4 ---------------
                   2  1/2
      1 + 1/2 a  2

             /                  1/4 \
         3/4 |    1        a + 2    |       1/4
      b 2    |-------- - -----------| (a - 2   )
             |     1/4         1/4 2|
             \a - 2      (a - 2   ) /
+ 5/8 -----------------------------------------
                              1/4
                         a + 2
```

simplify(");

```
                              1/2
           (5 a - 3 b) b 2    a
- ------------------------------------
          2  1/2        1/4        1/4
  (2 + a  2   ) (a - 2   ) (a + 2   )
```

simplify(numer(")/expand(denom(")));

```
  (5 a - 3 b) b a
- ---------------
        4
       a  - 2
```

- partial fraction decomposition
  convert((10*x^2- 11*x- 6) / (x^3- x^2- 2*x),parfrac,x);

```
         5       2
3/x + ----- + -----
      x + 1   x - 2
```

4.4.4 Solving equations

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

Linear systems

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

solve(2*x1 + x2 + 3*x3 - 9 = 0, x1 - 2*x2 + x3 + 2 = 0, 3*x1 + 2*x2 + 2*x3 - 7 = 0, x1, x2, x3);

```
{x1 = -1, x2 = 2, x3 = 3}
```

Nonlinear equations

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

- using decompositioni for solving high degree polynomials

  solve(x^8 - 8*x^7 + 34*x^6 - 92*x^5 + 175*x^4 - 236*x^3 + 226*x^2 - 140*x + 46);

```
             1/2             1/2             1/2             1/2
    1 + 1/2 2     (- 3 + %2)    , 1 - 1/2 2     (- 3 + %2)    ,

                 1/2             1/2             1/2             1/2
        1 + 1/2 2     (- 3 - %2)    , 1 - 1/2 2     (- 3 - %2)    ,

                 1/2             1/2             1/2             1/2
        1 + 1/2 2     (- 3 + %1)    , 1 - 1/2 2     (- 3 + %1)    ,

                 1/2             1/2             1/2             1/2
        1 + 1/2 2     (- 3 - %1)    , 1 - 1/2 2     (- 3 - %1)

                                          1/2 1/2
%1 :=                             (- 3 - 4 3    )

                                          1/2 1/2
%2 :=                             (- 3 + 4 3    )
```

- multiple use of inversion functions

  solve(log(arccos(arcsin(x^(2/3)-b)- 1)) + 2);

```
              3                                    2
{x = \_XX1 , b = - sin(cos(exp(-2)) + 1) + \_XX1 }
```

Nonlinear systems

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter **??**) Macsyma (see chapter **??**) Mathematica (see chapter 4.5.4) Reduce (see chapter 4.6.4)

- polynomial systems by the use of (see chapter2.7)

  solve(alpha* c1 - beta* c1^2 - gamma* c1*c2 + epsilon* c3 = 0, -gamma* c1*c2 + (epsilon+theta)* c3 - eta* c2 = 0, gamma* c1*c2 + eta* c2 - (epsilon+theta)* c3 = 0 ,c3,c2,c1);

```
{c1 = c1,

c2 =

  c1 (- epsilon alpha + c1 epsilon beta - theta alpha + c1 theta beta)
- --------------------------------------------------------------------
                    - eta epsilon + c1 theta gamma

,

c3 =

                                                              2
      c1 (- alpha eta + beta c1 eta - gamma c1 alpha + gamma c1  beta)
    - ----------------------------------------------------------------
                      - eta epsilon + c1 theta gamma

}
```

4.4.5 Analytical operations

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

Limits

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

limit(sin(x) / x, x = 0);

```
                                       1
```

limit((3*sin(Pi*x)- sin(3*Pi*x)) / x^3, x = 0);

```
                                         3
                                     4 Pi
```

limit((2*x + 5) / (3*x - 2), x = infinity);

```
                                      2/3
```

Taylor series

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

series(E^x, x=0, 5);

```
                      2         3          4       5
1 + x + 1/2 x  + 1/6 x  + 1/24 x  + O(x )
```

series(E^(x+y), x=0, 3);

```
                                           2       3
              exp(y) + exp(y) x + 1/2 exp(y) x  + O(x )
```

series(", y=0, 3);

```
                                    O(1)
```

Summation and Products

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

sum(m^2 * x^m, m=1..n);

```
   (n + 1)              2                  2              2  2
  x        (- 2 (n + 1) x  + 2 (n + 1) x + x  + x + (n + 1)  x

                    2           2    /        3   x (x + 1)
       - 2 (n + 1)  x + (n + 1) )  /  (x - 1)  - ---------
                                  /                      3
                                                  (x - 1)
```

sum(cos((2*m-1)* Pi / (2*n+1)), m=1..r);

```
        Pi (r + 1)              Pi   2     Pi (r + 1)
    - cos(----------) (- 2 cos(-------)  sin(----------)
           2 n + 1              2 n + 1       2 n + 1

                 Pi        Pi (r + 1)         Pi          Pi (r + 1)
        + 2 cos(-------) cos(----------) sin(-------) + sin(----------))
                2 n + 1       2 n + 1       2 n + 1         2 n + 1

          /       Pi              Pi
         /  sin(-------) + cos(-------)
        /       2 n + 1         2 n + 1
```

simplify(");

```
          Pi (r + 1)          Pi   2     Pi (r + 1)
     (2 cos(----------) cos(-------)  sin(----------)
             2 n + 1       2 n + 1        2 n + 1

                  Pi        Pi (r + 1) 2        Pi
         - 2 cos(-------) cos(----------)  sin(-------)
                 2 n + 1       2 n + 1        2 n + 1

              Pi (r + 1)      Pi (r + 1)          Pi           Pi
         - cos(----------) sin(----------) + cos(-------) sin(-------))
                2 n + 1         2 n + 1          2 n + 1      2 n + 1
```

```
           /        Pi
          /  sin(-------)
         /       2 n + 1
```

product(exp(sin(m*x)), m=1..n);

```
    (exp(

         - sin((n + 1) x) cos(x) + sin((n + 1) x) + cos((n + 1) x) sin(x)
     1/2 ----------------------------------------------------------------
                                 cos(x) - 1

         /           sin(x)
     ))  /  exp(1/2 ----------)
        /           cos(x) - 1
```

Integration

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

int(x^2 * (a+b*x)^p, x);

```
                 3          p                   2              p
                a  (a + b x)                   a  p x (a + b x)
          2 ------------------------- - 2 -------------------------
             3   3      2                   2   3      2
            b  (p  + 6 p  + 11 p + 6)      b  (p  + 6 p  + 11 p + 6)

                     2          p    3          p
                p a x  (a + b x)    x  (a + b x)
              + ----------------- + -------------
                     2                  p + 3
                b (p  + 5 p + 6)
```

simplify(");

```
             p   2    2  2    3  3  2        2  2      2           3  3
    (a + b x)  (p  a x  b  + x  b  p  + p a x  b  - 2 a  p x b + 3 x  b  p

             3      3  3    /                              3
         + 2 a  + 2 x  b )  /  ((p + 3) (p + 2) (p + 1) b )
                           /
```

int(x^2* log(x^2 + a^2), x);

```
              3     2    2         2          3        3
         1/3 x  ln(x  + a ) + 2/3 a  x - 2/9 x  - 2/3 a  arctan(x/a)
```

int(x* d^x * sin(x), x);

```
        x            2             2       x
     x d  tan(1/2 x)    (- 2 ln(d)  + 2) d  tan(1/2 x)
    (---------------- + -----------------------------
              2                          2     2
        ln(d)  + 1                  (ln(d)  + 1)

                    x                      x           2         x
          ln(d) x d  tan(1/2 x)     ln(d) d  tan(1/2 x)       x d
     + 2 --------------------- - 2 -------------------- - ----------
                        2                   2     2            2
                 ln(d)  + 1            (ln(d)  + 1)       ln(d)  + 1

                   x
             ln(d) d       /              2
     + 2 -------------)  /  (1 + tan(1/2 x) )
                2     2  /
          (ln(d)  + 1)
```

simplify(");

```
                                                2
     - (- 2 ln(d) cos(x) + x cos(x) ln(d)  + x cos(x) - sin(x)

                3                                          2   x
          - ln(d)  x sin(x) - ln(d) x sin(x) + sin(x) ln(d) ) d

            /       4          2
           /  (ln(d)  + 2 ln(d)  + 1)
          /
```

int(x* sqrt(a + b*x)^p, x);

```
                          /
                         |                1/2 p
                         |  x ((a + b x)     )  dx
                         |
                        /
```

int(2*x* exp(x^2)* log(x)+ exp(x^2)/x + (log(x)-2)/(log(x)^2+x)^2 + ((2/x)* log(x)+ (1/x)+1)/ (log(x)^2+x), x);

```
                  2            ln(x)             2
            exp(x ) ln(x) - ---------- + ln(ln(x)  + x)
                                 2
                            ln(x)  + x
```

Ordinary differential equations

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter **??**) Macsyma (see chapter 4.3.5) Mathematica (see chapter 4.5.5) Reduce (see chapter 4.6.5)

dsolve(diff(y(x),x) + y(x)*sin(x)/cos(x) - 1/cos(x) = 0, y(x));

```
y(x) = sin(x) + cos(x) \_C1
```

- Bernoulli equation

  dsolve(x*(1-x^2)* diff(y(x),x) + (2*x^2-1)* y(x) - x^3* y^3 = 0, y(x));

```
                              5
  1             2 x  + 5 \_C1
----- = 1/5 -------------
    2         2          2
y(x)         x  (- 1 + x )
```

solve(", y(x));

```
 1/2         5      7                    2 1/2
5    x (- 2 x  + 2 x  - 5 \_C1 + 5 \_C1 x )
------------------------------------------,
                     5
              - 2 x  - 5 \_C1

   1/2         5      7                    2 1/2
  5    x (- 2 x  + 2 x  - 5 \_C1 + 5 \_C1 x )
- ------------------------------------------
                       5
                - 2 x  - 5 \_C1
```

dsolve(diff(y(x),x,x) + 4* diff(y(x),x) + 4*y(x) - x* exp(x) = 0,y(x));

```
y(x) = 1/9 x exp(x) - 2/27 exp(x) + \_C1 exp(- 2 x) + \_C2 exp(- 2 x) x
```

4.4.6 Matrices

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.6) Derive (see chapter 4.2.6) Macsyma (see chapter 4.3.6) Mathematica (see chapter 4.5.6) Reduce (see chapter 4.6.6)

xx := array([[a11,a12], [a21,a22]]);

```
      [ a11  a12 ]
xx := [          ]
      [ a21  a22 ]
```

yy := array([y1,y2]);

```
yy := [ y1, y2 ]
```

evalm(xx^(-1) * yy);

```
     a22 y1 - a12 y2      a21 y1 - a11 y2
  [ -----------------, - ----------------- ]
    a11 a22 - a12 a21    a11 a22 - a12 a21
```

with(linalg);
det(xx);

```
                 a11 a22 - a12 a21
```

evalm(xx^(-2));

```
[          2                                      ]
[       a22  + a12 a21          a12 (a22 + a11)   ]
[  --------------------   - -------------------- ]
[                     2                         2 ]
[  (a11 a22 - a12 a21)      (a11 a22 - a12 a21)  ]
[                                                 ]
[                                           2     ]
[       a21 (a22 + a11)       a12 a21 + a11       ]
[ - --------------------   --------------------  ]
[                      2                       2  ]
[   (a11 a22 - a12 a21)    (a11 a22 - a12 a21)   ]
```

eigenvals([[2,-1,1], [0,1,1], [-1,1,1]]);

```
                      2, 1, 1
```

4.4.7 Graphics

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

2D Graphics

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- function of one parameter
- graph of several functions (Bessel functions $J(n,x), n = 0,2,5$)

3D Graphics

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- graph of a function with 2 parameters
- graph of another function with 2 parameters

Parametric plots

In Maple

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- parametricaly given surface

Contour maps

In Maple

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Macsyma (see chapter 4.3.8) Mathematica (see chapter 4.5.7) Reduce (see chapter 4.6.8)

- map of a function with 2 parameters
- map of another function with 2 parameters
- map of further function with 2 parameters
- map of transformation of cartesian coordinate grid by a complex function

Polytopes

In Maple

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Macsyma (see chapter ??) Mathematica (see chapter 4.5.7) Reduce (see chapter ??)

- graphical presentation of polyhedrons

4.4.8 Graphical presentation of formulas

In Maple

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Macsyma (see chapter 4.3.9) Mathematica (see chapter 4.5.8) Reduce (see chapter 4.6.9)

4.5 Mathematica

- **inputs**

  `outputs`

4.5.1 Number domains

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Reduce (see chapter 4.6.1)

Big integers

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Reduce (see chapter 4.6.1)

- integers of arbitrary size

  23^12

  ```
  21914624432020321
  ```

60!

```
8320987112741390144276341183223364380754172606361245952\
  449277696409600000000000000
```

Factorial[60]

```
8320987112741390144276341183223364380754172606361245952\
  449277696409600000000000000
```

bi = 23^4*37*59*101

```
61700183203
```

- factorization of integers

FactorInteger[bi]

```
{{23, 4}, {37, 1}, {59, 1}, {101, 1}}
```

bia = 23*11^6

```
40745903
```

- greatest common divisor of integers

GCD[bi, bia]

```
23
```

Rational numbers

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Reduce (see chapter 4.6.1)

- precise calculation with rational numbers

1234567890/98765432

```
617283945
---------
49382716
```

rn = 1/2+2/15-64/47

```
  1027
-(----)
  1410
```

Complex numbers

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Reduce (see chapter 4.6.1)

- precise calculation with complex numbers

  cn = (2+3I)*(15-6I)+2/(2-4I)

```
241   167 I
--- + -----
 5      5
```

Big floating point numbers

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Reduce (see chapter 4.6.1)

rn = - 1027/1410

```
  1027
-(----)
  1410
```

cn = (167*I + 241)/5

```
241   167 I
--- + -----
 5      5
```

- computation with floating point numbers with arbitrary number of digits

  N[rn, 20]

```
-0.72836879432624113475
```

N[cn, 10]

```
48.2 + 33.4 I
```

N[Pi, 40]

```
3.141592653589793238462643383279502884197 2
```

Cos[Pi]

```
-1
```

N[Sin[1], 20]

```
0.84147098480789650665
```

- should be cos(pi/6) = sqrt(3)/2

  N[Cos[Pi/6], 50]

  ```
  0.86602540378443864676372317075293618347140262690519
  ```

  %ˆ2

  ```
  0.75
  ```

- complex functions

  Tan[1.0 + 1.0I]

  ```
  0.271753 + 1.08392 I
  ```

  Log[N[1,30]+I]

  ```
  0.346573590279972654708616060729 +
   0.785398163397448309615660845820 I
  ```

  Precision[%]

  ```
  30
  ```

  N[Log[N[1,40] + I], 60]

  ```
  0.3465735902799726547086160607290882840378 +
   0.7853981633974483096156608458198757210493 I
  ```

  Precision[%]

  ```
  40
  ```

4.5.2 Polynomials

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Reduce (see chapter 4.6.2)

Basic operations

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Reduce (see chapter 4.6.2)

- by default parantheses are not expanded

  pol := (a+b+c)ˆ4

  Expand[pol]

$$a^4 + 4 a^3 b + 6 a^2 b^2 + 4 a b^3 + b^4 + 4 a^3 c +$$

$$12 a^2 b c + 12 a b^2 c + 4 b^3 c + 6 a^2 c^2 + 12 a b c^2 +$$

$$6 b^2 c^2 + 4 a c^3 + 4 b c^3 + c^4$$

- derivation

  dpol = D[Expand[pol], a]

$$4 a^3 + 12 a^2 b + 12 a b^2 + 4 b^3 + 12 a^2 c + 24 a b c +$$

$$12 b^2 c + 12 a c^2 + 12 b c^2 + 4 c^3$$

D[pol, a, b, 2]

$$24 (a + b + c)$$

- integration

  Integrate[dpol, a]

$$a^4 + 4 a^3 (b + c) + 6 a^2 (b + c)^2 + 4 a (b + c)^3$$

- verification

  % - pol

$$a^4 + 4 a^3 (b + c) + 6 a^2 (b + c)^2 + 4 a (b + c)^3 -$$

$$(a + b + c)^4$$

Expand[%]

$$-b^4 - 4 b^3 c - 6 b^2 c^2 - 4 b c^3 - c^4$$

- greatest common divisor of polynomials

  (a^2-b^2)/(a^2-2 a b+b^2)

$$\frac{a^2 - b^2}{a^2 - 2 a b + b^2}$$

Simplify[%]

```
a + b
-----
a - b
```

g := 34 x^19 - 25 x^16 + 70 x^7 + 20 x^3 - 91 x - 86

f1 := g (64 x^34 - 21 x^47 - 126 x^8 - 46 x^5 - 16 x^60 - 81)

f1

```
                     3        7        16        19
(-86 - 91 x + 20 x  + 70 x  - 25 x   + 34 x  )

            5          8        34        47        60
 (-81 - 46 x  - 126 x  + 64 x   - 21 x   - 16 x  )
```

f1 = Expand[%]

```
                         3          5          6          7
6966 + 7371 x - 1620 x  + 3956 x  + 4186 x  - 5670 x  +

       8           9          11          12          15
 9916 x  + 11466 x  - 2520 x   - 3220 x   - 8820 x   +

       16          19          21          24          27
 2025 x   - 2754 x   + 1150 x   + 1586 x   - 4284 x   -

       34          35          37          41          47
 5504 x   - 5824 x   + 1280 x   + 4480 x   + 1806 x   +

       48          50          53          54          60
 1911 x   - 2020 x   + 2176 x   - 1470 x   + 1376 x   +

       61         63         66          67         76
 1456 x   + 205 x   - 714 x   - 1120 x   + 400 x   -

      79
 544 x
```

f2 := g (72 x^60 - 25 x^25 - 19 x^23 -22 x^39 - 83 x^52 + 54 x^10 + 81)

f2 = Expand[f2]

```
                          3          7          10
-6966 - 7371 x + 1620 x  + 5670 x  - 4644 x   -

       11          13          16          17          19
 4914 x   + 1080 x   - 2025 x   + 3780 x   + 2754 x   +

       23          24          25         26         28
 1634 x   + 1729 x   + 2150 x   + 545 x   - 500 x   +

       29          30          32          39          40
 1836 x   - 1330 x   - 1750 x   + 2367 x   + 2002 x   +
```

```
          41          42         44          46          52
     625 x   - 1086 x   - 850 x   - 1540 x   + 7138 x   +

           53          55         58          59          60
     7553 x   - 1110 x   - 748 x   - 5810 x   - 6192 x   -

           61          63          67          68          71
     6552 x   + 1440 x   + 5040 x   + 2075 x   - 2822 x   -

           76          79
     1800 x   + 2448 x
```

PolynomialGCD[**f1, f2**]

```
                      3       7        16        19
    -86 - 91 x + 20 x  + 70 x  - 25 x   + 34 x
```

Factorization

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Reduce (see chapter 4.6.2)

- factorization is transformation of a polynomial into a product of polynomials

  Factor[**a^2 - b^2**]

```
    (a - b) (a + b)
```

Factor[**a^2 + b^2**]

```
     2    2
    a  + b
```

Factor[**a^2 + b^2, GaussianIntegers -¿ True**]

```
    (-I a + b) (I a + b)
```

fa := (x^2 z + y^4 z^2 + 5) * (x y^3 + z^2) * (-x^3 y + z^2 + 3) * (x^3 y^4 + z^2)
fa = Expand[fa]

```
        4   7       7   8       6   7       9   8            3   2
    15 x   y  - 5 x   y  + 3 x   y  z - x   y  z + 15 x y  z  +

         3   4   2       4   4   2       6   5   2       4   7   2
     15 x   y   z  - 5 x   y   z  - 5 x   y   z  + 5 x   y   z  +

        4   11   2     7   12   2      3   3   3      5   4   3
     3 x   y    z  - x   y    z  + 3 x   y   z  + 3 x   y   z  -

      6   4   3    8   5   3    6   7   3        4       3    4
     x   y   z  - x   y   z  + x   y   z  + 15 z  - 5 x   y z  +
```

```
      3  4       3  4  4        7  4      3  8  4
5 x y  z  + 5 x  y  z  + 3 x y  z  + 3 x  y  z  -

 4  8  4    6  9  4    4  11  4      2  5    5    5
x  y  z  - x  y  z  + x  y   z  + 3 x  z  - x  y z  +

 3  3  5    5  4  5      6      4  6    3  5  6
x  y  z  + x  y  z  + 5 z  + 3 y  z  - x  y  z  +

   7  6    3  8  6    2  7    4  8
x y  z  + x  y  z  + x  z  + y  z
```

Factor[**fa**]

```
      3      2      3    2    3  4    2
(3 - x  y + z ) (x y  + z ) (x  y  + z )

       2      4  2
 (5 + x  z + y  z )
```

Decomposition

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Reduce (see chapter 4.6.2)

- finding of polynomial substitution which transforms a polynomial into another polynomial in further polynomials

  Decompose[**x^6 + 9x^5 + 52x^4 + 177x^3 + 435x^2 + 630x + 593**]

  ```
                        2    3
  {593 + 210 x + 25 x  + x , x (3 + x)}
  ```

  Decompose[**x^4 + 2x^3 y + 3x^2 y^2 + 2x y^3 + y^4 + 2x^2 y + 2x y^2 + 2y^3 + 5x^2 + 5 x y + 6y^2 + 5y + 9, x**]

  ```
               2                    2        2      3
  {9 + 5 x + x  + 5 y + 2 x y + 6 y  + 2 x y  + 2 y  +

     4
    y , x (x + y)}
  ```

Grobner bases

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Reduce (see chapter 4.6.2)

polys := 45p + 35s - 165b - 36, 35p + 40z + 25t - 27s, 15w + 25p s + 30z - 18t - 165b^2, -9w + 15p t + 20z s, w p + 2z t - 11b^3, 99w - 11s b + 3b^2, b^2 + 33/50b + 2673/10000

vars := w,p,z,t,s,b

GroebnerBasis[**polys, vars**]

```
                                 2
{2673 + 6600 b + 10000 b , -9 - 500 b + 200 s,

 81 - 1850 b + 750 t, 10287 + 24500 b + 18000 z,

 -1377 - 3100 b + 1800 p, 3969 + 9500 b + 60000 w}
```

4.5.3 Rational functions

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.3) Derive (see chapter 4.2.3) Macsyma (see chapter 4.3.3) Maple (see chapter 4.4.3) Reduce (see chapter 4.6.3)

- integration

  Integrate[**(3a b^2 - 5a^2 b)/(a^4 - 2), a**]

```
             a
-5 b ArcTan[----]
             1/4
            2
----------------- -
        1/4
     2 2

            3/4  2             3/4
  (10 b - 3 2    b ) Log[-2 + 2    a]
  ---------------------------------- -
                      1/4
                   8 2

            3/4  2            3/4
 (-10 b - 3 2    b ) Log[2 + 2    a]
 ---------------------------------- -
                     1/4
                  8 2

    2                  2
 3 b  Log[2 + Sqrt[2] a ]
 ------------------------
       4 Sqrt[2]
```

- verification by derivation

  D[%,a]

```
                                          2
         -5 b                       3 a b
----------------------- - ------------------ -
                   2                      2
                  a       2 (2 + Sqrt[2] a )
2 Sqrt[2] (1 + -------)
                Sqrt[2]

              3/4  2                  3/4  2
```

```
   -10 b - 3 2    b          10 b - 3 2    b
 --------------------- - ----------------------
                3/4                        3/4
4 Sqrt[2] (2 + 2    a)   4 Sqrt[2] (-2 + 2    a)
```

Simplify[%]

```
a b (-5 a + 3 b)
----------------
          4
    -2 + a
```

- partial fraction decomposition

  Apart[**(10x^2- 11x- 6)/(x^3- x^2- 2x)**]

```
  2       3     5
------ + - + -----
-2 + x   x   1 + x
```

4.5.4 Solving equations

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Reduce (see chapter 4.6.4)

Linear systems

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Reduce (see chapter 4.6.4)

Solve[**2 x1 + x2 + 3 x3 - 9 == 0, x1 - 2 x2 + x3 + 2 == 0, 3 x1 + 2 x2 + 2 x3 - 7 == 0, x1, x2, x3**]

```
{{x1 -> -1, x2 -> 2, x3 -> 3}}
```

Nonlinear equations

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Reduce (see chapter 4.6.4)

- using decompositioni for solving high degree polynomials

  Solve[**x^8 - 8x^7 + 34x^6 - 92x^5 + 175x^4 - 236x^3 + 226x^2 - 140x + 46 == 0**]

```
       4 - Sqrt[16 - 8 (5 - Sqrt[-3 - 4 Sqrt[3]])]
{{x -> -------------------------------------------},
                           4

       4 + Sqrt[16 - 8 (5 - Sqrt[-3 - 4 Sqrt[3]])]
 {x -> -------------------------------------------},
                           4

       4 - Sqrt[16 - 8 (5 + Sqrt[-3 - 4 Sqrt[3]])]
```

```
  {x -> -----------------------------------------------},
                              4

         4 + Sqrt[16 - 8 (5 + Sqrt[-3 - 4 Sqrt[3]])]
  {x -> -----------------------------------------------},
                              4

         4 - Sqrt[16 - 8 (5 - Sqrt[-3 + 4 Sqrt[3]])]
  {x -> -----------------------------------------------},
                              4

         4 + Sqrt[16 - 8 (5 - Sqrt[-3 + 4 Sqrt[3]])]
  {x -> -----------------------------------------------},
                              4

         4 - Sqrt[16 - 8 (5 + Sqrt[-3 + 4 Sqrt[3]])]
  {x -> -----------------------------------------------},
                              4

         4 + Sqrt[16 - 8 (5 + Sqrt[-3 + 4 Sqrt[3]])]
  {x -> -----------------------------------------------}}
                              4
```

- multiple use of inversion functions

  Solve[**Log**[**ArcCos**[**ArcSin**[**x^(2/3)-b**]- **1**]]+ **2 == 0, x**]

```
                 3      2              -2
 {{x -> -Sqrt[b  + 3 b  Sin[1 + Cos[E  ]] +

                         -2  2                -2  3
       3 b Sin[1 + Cos[E  ]]  + Sin[1 + Cos[E  ]] ]},

                3      2              -2
  {x -> Sqrt[b  + 3 b  Sin[1 + Cos[E  ]] +

                        -2  2                -2  3
      3 b Sin[1 + Cos[E  ]]  + Sin[1 + Cos[E  ]] ]}}
```

Simplify[%]

```
                              1   2  3
 {{x -> -Sqrt[(b + Sin[2 Cos[----] ]) ]},
                                2
                              2 E

                             1   2  3
  {x -> Sqrt[(b + Sin[2 Cos[----] ]) ]}}
                               2
                             2 E
```

Nonlinear systems

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter **??**) Macsyma (see chapter **??**) Maple (see chapter 4.4.4) Reduce (see chapter 4.6.4)

- polynomial systems by the use of (see chapter2.7)

  Solve[alpha c1 -beta c1^2 -gamma c1 c2 +epsilon c3 == 0, -gamma c1 c2 + (epsilon+theta) c3 - eta c2 == 0, gamma c1 c2 + eta c2 - (epsilon+theta) c3 == 0 , c3,c2,c1]

```
{{c3 -> (c1 (-alpha + beta c1 -

             alpha c1 epsilon gamma
          ---------------------------- +
          epsilon eta - c1 gamma theta

                    2
             beta c1  epsilon gamma
          ---------------------------- -
          epsilon eta - c1 gamma theta

              alpha c1 gamma theta
          ---------------------------- +
          epsilon eta - c1 gamma theta

                     2
              beta c1  gamma theta
          ----------------------------)) / epsilon,
          epsilon eta - c1 gamma theta

  c2 -> (c1 (alpha epsilon - beta c1 epsilon +

          alpha theta - beta c1 theta)) /

    (-(epsilon eta) + c1 gamma theta)}}
```

Simplify[%]

```
          c1 (-alpha + beta c1) (eta + c1 gamma)
{{c3 -> ---------------------------------------,
               epsilon eta - c1 gamma theta

          c1 (alpha - beta c1) (epsilon + theta)
  c2 -> ---------------------------------------}}
             -(epsilon eta) + c1 gamma theta
```

- c1 is an arbitrary complex number

4.5.5 Analytical operations

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

Limits

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

Limit[**Sin**[x]/**x, x -¿ 0**]

```
1
```

Limit[**(3 Sin**[**Pi x**] **- Sin**[**3 Pi x**]**)/x^3, x -¿ 0**]

```
     3
4 Pi
```

Limit[**(2x + 5)/ (3x - 2), x -¿ Infinity**]

```
2
-
3
```

Taylor series

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

Series[**E^x, x,0,4**]

```
         2    3    4
        x    x    x        5
1 + x + -- + -- + -- + O[x]
        2    6    24
```

Series[**E^(x+y), x,0,2, y,0,2**]

```
         2                      2
        y        3             y        3
1 + y + -- + O[y]  + (1 + y + -- + O[y] ) x +
        2                      2

           2
  1   y   y         3   2       3
 (- + - + -- + O[y] ) x  + O[x]
  2   2   4
```

%^2

```
                 2       3                      2       3
1 + 2 y + 2 y  + O[y]  + (2 + 4 y + 4 y  + O[y] ) x +

                   2       3   2       3
 (2 + 4 y + 4 y  + O[y] ) x  + O[x]
```

Summation and Products

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

- modul Algebra'SymbolicSum' dealing with sumation

  Needs["Algebra'SymbolicSum'"]

  Sum[m^2 x^m, m, 1, n]

```
                 n         n    2  n    1 + n
(x (-1 - x + x  + 2 n x  + n  x  + x      -

             1 + n      2  1 + n    2  2 + n              3
     2 n x      - 2 n  x      + n  x     )) /   (-1 + x)
```

  Sum[Cos[(2m-1) Pi/ (2n+1)], m,1,r]

```
          Pi             Pi       Pi r
-Cos[-------] + Cos[------- - -------]
      1 + 2 n         1 + 2 n   1 + 2 n

           Pi           Pi       Pi r
    Csc[-------] Sin[------- + -------]
         1 + 2 n      1 + 2 n   1 + 2 n
```

  Product[Exp[Sin[m x]], m, 1, n]

```
 Csc[x/2] Sin[(n x)/2] Sin[((1 + n) x)/2]
E
```

  Sum[m 2^m/ Factorial[m+2], m,1,n]

```
    n      n
-6 2  - 2 2  n + Gamma[4 + n]
-----------------------------
        Gamma[4 + n]
```

Integration

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

Integrate[x^2 (a+b x)^p, x]

```
                             3
         p                2 a
(a + b x)  (-------------------------- -
              3
             b  (1 + p) (2 + p) (3 + p)
```

```
               2
           2 a  p x
----------------------------- +
(3 + p) (b + b p) (2 b + b p)

             2                3
      a p x              b x
------------------- + ---------)
(3 + p) (2 b + b p)   3 b + b p
```

Simplify[%]

```
          1 + p     2                            2  2
((a + b x)      (2 a  - 2 a b x - 2 a b p x + 2 b  x  +

        2    2    2  2  2        3                 2    3
     3 b  p x  + b  p  x )) / (b  (6 + 11 p + 6 p  + p ))
```

Integrate[**x^2 Log**[**x^2 + a^2**], **x**]

```
                     3        x
  2        3   2 a  ArcTan[-]    3      2    2
2 a  x   2 x                 a    x  Log[a  + x ]
------ - ---- - -------------- + ---------------
  3       9           3                 3
```

Simplify[%]

```
  2        3      3        x       3      2    2
6 a  x - 2 x  - 6 a  ArcTan[-] + 3 x  Log[a  + x ]
                             a
--------------------------------------------------
                        9
```

Simplify[**Integrate**[**x d^x Sin**[**x**], **x**]]

```
  x                                                 2
(d  (-(x Cos[x]) + 2 Cos[x] Log[d] - x Cos[x] Log[d]  +

                                           2
     Sin[x] + x Log[d] Sin[x] - Log[d]  Sin[x] +

             3                          2 2
     x Log[d]  Sin[x])) / (1 + Log[d] )
```

Simplify[**Integrate**[**x Sqrt**[**a + b x**]**^p**, **x**]]

```
           1 + p/2
2 (a + b x)        (-2 a + 2 b x + b p x)
-----------------------------------------
```

```
            2
           b  (2 + p) (4 + p)
```

Simplify[**Integrate**[**2x Exp**[x^2] **Log**[x]+ **Exp**[x^2]/x + (**Log**[x]-2)/(**Log**[x]^2+x)^2 + ((2/x) **Log**[x]+ (1/x)+1)/ (**Log**[x]^2+x), x]]

```
      2
     x             Log[x]                      2
    E   Log[x] - ----------- + Log[x + Log[x] ]
                           2
                 x + Log[x]
```

Ordinary differential equations

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter **??**) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Reduce (see chapter 4.6.5)

DSolve[y'[x] + y[x] **Sin**[x]/**Cos**[x] - 1/**Cos**[x]==0, y[x],x]

```
{{y[x] -> C[1] Cos[x] + Sin[x]}}
```

- Bernoulli equation

  Simplify[**DSolve**[x (1-x^2) y'[x] + (2x^2-1) y[x]- x^3 y^3 == 0, y[x], x]]

```
                   3              2
{{y[x] -> x (y   + Sqrt[1 - x ] C[1])}}
```

Simplify[**DSolve**[y"[x] + 4y'[x] + 4y[x] - x **Exp**[x] == 0, y[x], x]]

```
                 x     x
             -2 E     E  x    C[1]    x C[2]
{{y[x] -> ----- + ---- + ---- + ------}}
              27      9       2 x     2 x
                             E       E
```

4.5.6 Matrices

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.6) Derive (see chapter 4.2.6) Macsyma (see chapter 4.3.6) Maple (see chapter 4.4.6) Reduce (see chapter 4.6.6)

xx := a11,a12, a21,a22
yy := y1,y2
xx // MatrixForm

```
a11   a12

a21   a22
```

Det[xx]

```
-(a12 a21) + a11 a22
```

zz = Inverse[xx] . yy // MatrixForm

```
      a22 y1                  a12 y2
-------------------- - --------------------
-(a12 a21) + a11 a22   -(a12 a21) + a11 a22

        a21 y1                  a11 y2
-(--------------------) + --------------------
  -(a12 a21) + a11 a22    -(a12 a21) + a11 a22
```

Simplify[%] // MatrixForm

```
  a22 y1 - a12 y2
--------------------
-(a12 a21) + a11 a22

 -(a21 y1) + a11 y2
--------------------
-(a12 a21) + a11 a22
```

MatrixPower[xx, -2] // Simplify // MatrixForm

```
                 2
    a12 a21 + a22             a12 (a11 + a22)
-----------------------   -(--------------------)
                      2                        2
(-(a12 a21) + a11 a22)      (a12 a21 - a11 a22)

                                  2
    a21 (a11 + a22)            a11  + a12 a21
-(--------------------)   -----------------------
                     2                          2
  (a12 a21 - a11 a22)     (-(a12 a21) + a11 a22)
```

Eigenvalues[2,-1,1, 0,1,1, -1,1,1]

```
{1, 1, 2}
```

4.5.7 Graphics

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Reduce (see chapter 4.6.8)

2D Graphics

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Reduce (see chapter 4.6.8)

- function of one parameter
- including labels
- graph of several functions (Bessel functions $J(n, x), n = 0, 2, 5$)

3D Graphics

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Reduce (see chapter 4.6.8)

- graph of a function with 2 parameters
- graph of another function with 2 parameters

Parametric plots

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Reduce (see chapter 4.6.8)

- parametricaly given surface
- combination of graphical objects

Contour maps

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Reduce (see chapter 4.6.8)

- map of a function with 2 parameters
- map of another function with 2 parameters
- map of further function with 2 parameters
- map of transformation of cartesian coordinate grid by a complex function

Polytopes

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter **??**) Maple (see chapter 4.4.7) Reduce (see chapter **??**)

- graphical presentation of polytopes

4.5.8 Graphical presentation of formulas

In Mathematica

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.9) Maple (see chapter 4.4.8) Reduce (see chapter 4.6.9)

4.6 Reduce

- **inputs**

  `outputs`

4.6.1 Number domains

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1)

Big integers

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1)

- integers of arbitrary size

  23^12;

  ```
  21914624432020321
  ```

  factorial 60;

  ```
  8320987112741390144276341183223364380754172606361245952449277696409600
  000000000000
  ```

  bi:=23^4*37*59*101;

  ```
  bi := 61700183203
  ```

- factorization of integers

  on ifactor;

  factorize bi;

  ```
  {23,23,23,23,37,59,101}
  ```

  bia:=23*11^6;

  ```
  bia := 40745903
  ```

- integer greatest common divisor

  gcd(bi,bia);

  ```
  23
  ```

Rational numbers

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1)

- exact calculation with rational numbers

  1234567890/98765432;

```
 617283945
-----------
 49382716
```

  rn:=1/2+2/15-64/47;

```
        - 1027
rn := ---------
        1410
```

Complex numbers

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1)

- exact calculation with complex numbers

  cn:=(2+3*i)*(15-6*i)+2/(2-4*i);

```
       63*i - 115
cn := ------------
        2*i - 1
```

- basic number domain changed to that of complex numbers

  on complex;

  cn:=cn;

```
       241 + 167*i
cn := -------------
            5
```

Algebraic numbers

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter **??**) Mathematica (see chapter **??**)

- module arnum dealing with algebraic numbers

  load_package arnum;

  defpoly sqrt22-2;**

  1/(sqrt2+1);

```
sqrt2 - 1
```

(x2+2*sqrt2*x+2)/(x+sqrt2);**

```
x + sqrt2
```

on gcd;
(x3+(sqrt2-2)*x**2-(2*sqrt2+3)*x-3*sqrt2)/(x**2-2);**

```
  2
 x  - 2*x - 3
--------------
  x - sqrt2
```

off gcd;
sqrt(x2-2*sqrt2*x*y+2*y**2);**

```
x - sqrt2*y
```

- multiple algebraic extensions

off arnum;
defpoly sqrt52-5,cbrt3**3-3;**

```
*** Defining the polynomial for a primitive element:

  6        4       3        2
a1  - 15*a1  - 6*a1  + 75*a1  - 90*a1 - 116
```

cbrt33;**

```
3
```

sqrt52;**

```
5
```

cbrt3;

```
     120     5      27     4     2000    3     1170    2     6676
 - (------*a1  + ------*a1  - ------*a1  - ------*a1  + ------*a1
     8243          8243         8243         8243         8243

        6825
    - ------)
        8243
```

sqrt(x2+2*(sqrt5-cbrt3)*x+5-2*sqrt5*cbrt3+cbrt3**2);**

```
            240     5      54     4    4000    3    2340    2    21595
      x + (------*a1  + ------*a1  - ------*a1  - ------*a1  + -------*a1
            8243         8243         8243         8243         8243

               13650
            - -------)
               8243
```

Big floating point numbers

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.1) Derive (see chapter 4.2.1) Macsyma (see chapter 4.3.1) Maple (see chapter 4.4.1) Mathematica (see chapter 4.5.1)

rn:= - 1027/1410;

```
          - 1027
  rn := ---------
            1410
```

cn:=(167*i + 241)/5;

```
         167*i + 241
  cn := -------------
              5
```

- computation with floating point numbers

  on rounded;

  rn;

  ```
    - 0.728368794326
  ```

  cn;

  ```
  33.4*i + 48.2
  ```

  pi;

  ```
  3.14159265359
  ```

  cos pi;

  ```
    - 1
  ```

  sin 1;

  ```
  0.841470984808
  ```

- computation with an arbitrary number of digits

precision 50;

```
12
```

pi;

```
3.1415926535897932384626433832795028841971693993751
```

cos pi;

```
 - 1
```

- should be cos(pi/6) = sqrt(3)/2

cos(pi/6);

```
0.86602540378443864676372317075293618347140262690519
```

ws2;**

```
0.75
```

precision 10;

```
50
```

- by default, an underflow is converted to zero

exp(-100000.12);**

```
0
```

- it is possible to get very small numbers

on roundbf;
exp(-100000.12);**

```
1.184683941e-4342953505
```

off roundbf;

- complex functions

on complex;

```
*** Domain mode rounded changed to complex-rounded
```

tan(1.0 + 1.0*i);

```
0.2717525853 + 1.083923327*i
```

log(1.0 + 1.0*i);

```
0.3465735903 + 0.7853981634*i
```

4.6.2 Polynomials

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2)

Basic operations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2)

- by default, parentheses are expanded

  pol:=(a+b+c)^4;

```
          4      3        3        2  2        2          2  2        3
pol := a  + 4*a *b + 4*a *c + 6*a *b  + 12*a *b*c + 6*a *c  + 4*a*b

              2             2        3    4      3        2  2
        + 12*a*b *c + 12*a*b*c  + 4*a*c  + b  + 4*b *c + 6*b *c

             3    4
        + 4*b*c  + c
```

- differentiation

  dpol:=df(pol,a);

```
              3      2        2          2                   2    3
dpol := 4*(a  + 3*a *b + 3*a *c + 3*a*b  + 6*a*b*c + 3*a*c  + b

               2          2    3
            + 3*b *c + 3*b*c  + c )
```

df(pol,a,b,2);

```
24*(a + b + c)
```

- integration

  int(dpol,a);

```
     3      2         2          2                    2      3       2
a*(a  + 4*a *b + 4*a *c + 6*a*b  + 12*a*b*c + 6*a*c  + 4*b  + 12*b *c

           2      3
    + 12*b*c  + 4*c )
```

- verification

  ws-pol;

```
    4      3         2  2         3    4
 - b  - 4*b *c - 6*b *c  - 4*b*c  - c
```

- polynomial greatest common divisor

  on gcd,ezgcd;

  (a^2-b^2)/(a^2-2*a*b+b^2);

```
 a + b
-------
 a - b
```

off gcd;

g := 34*x19-91*x+70*x**7-25*x**16+20*x**3-86;**

```
         19        16        7       3
g := 34*x   - 25*x   + 70*x  + 20*x  - 91*x - 86
```

f1:=g * (64*x34-21*x**47-126*x**8-46*x**5-16*x**60-81);**

```
              79         76          67         66         63          61
f1 :=  - 544*x   + 400*x   - 1120*x   - 714*x   + 205*x   + 1456*x

              60          54          53          50          48
       + 1376*x   - 1470*x   + 2176*x   - 2020*x   + 1911*x

              47          41          37          35          34
       + 1806*x   + 4480*x   + 1280*x   - 5824*x   - 5504*x

              27          24          21          19          16
       - 4284*x   + 1586*x   + 1150*x   - 2754*x   + 2025*x

              15          12          11           9          8
       - 8820*x   - 3220*x   - 2520*x   + 11466*x  + 9916*x

              7          6          5          3
       - 5670*x  + 4186*x  + 3956*x  - 1620*x  + 7371*x + 6966
```

f2:=g * (72*x60-25*x**25-19*x**23-22*x**39-83*x**52+54*x**10+81);**

```
                79          76          71          68          67          63
f2 := 2448*x   - 1800*x   - 2822*x   + 2075*x   + 5040*x   + 1440*x

              61          60          59         58          55
       - 6552*x   - 6192*x   - 5810*x   - 748*x   - 1110*x

              53          52          46         44          42
       + 7553*x   + 7138*x   - 1540*x   - 850*x   - 1086*x

             41          40          39          32          30
       + 625*x   + 2002*x   + 2367*x   - 1750*x   - 1330*x

              29         28         26          25          24
       + 1836*x   - 500*x   + 545*x   + 2150*x   + 1729*x

              23          19          17          16          13
       + 1634*x   + 2754*x   + 3780*x   - 2025*x   + 1080*x

              11          10          7          3
       - 4914*x   - 4644*x   + 5670*x  + 1620*x  - 7371*x - 6966
```

gcd(f1,f2);

```
    19       16       7       3
34*x   - 25*x   + 70*x  + 20*x  - 91*x - 86
```

Factorization

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter 4.2.2) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2)

- factorization is the transformation of a polynomial into a product of polynomials

  factorize(a^2-b^2);

  ```
  {a - b,a + b}
  ```

  on complex;

  factorize(a^2+b^2);

  ```
  {a - i*b,a + i*b}
  ```

  off complex;

  fa:=(x2*z+y**4*z**2+5)* (x*y**3+z**2)* (-x**3*y+z**2+3)* (x**3*y**4+z**2);**

  ```
            9   8       8   5   3     7   12   2       7   8     6   9   4
  fa :=  - x *y *z - x *y *z  - x *y   *z  - 5*x *y  - x *y *z

            6   7   3       6   7         6   5   2     6   4   3     5   4   5
         + x *y *z  + 3*x *y *z - 5*x *y *z  - x *y *z  + x *y *z
  ```

```
      5   4   3     5     5     4  11  4       4  11  2     4  8  4
 + 3*x *y *z  - x *y*z  + x *y   *z  + 3*x *y   *z  - x *y *z

      4   7   2        4   7       4  4  2     3  8  6       3  8  4
 + 5*x *y *z  + 15*x *y   - 5*x *y *z  + x *y *z  + 3*x *y *z

    3   5   6       3   4   4        3   4  2     3   3  5       3   3  3
 - x *y *z  + 5*x *y *z  + 15*x *y *z  + x *y *z  + 3*x *y *z

      3     4     2  7       2   5      7  6          7   4
 - 5*x *y*z  + x *z  + 3*x *z  + x*y *z  + 3*x*y *z

        3   4          3   2     4   8      4   6       6       4
 + 5*x*y *z  + 15*x*y *z  + y *z  + 3*y *z  + 5*z  + 15*z
```

factorize fa;

```
    3    2
{x*y  + z ,

  2      4  2
 x *z + y *z  + 5,

    3      2
  - x *y + z  + 3,

  3  4    2
 x *y  + z }
```

Decomposition

In Reduce

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2)

- decomposing a polynomial into simpler polynomials which when composed, produce the original polynomial

  decompose(x6+9x**5+52x**4+177x**3+435x**2+630x+593);**

```
  3       2                   2
{u  + 25*u  + 210*u + 593,u=x  + 3*x}
```

decompose(x4+2x**3*y + 3x**2*y**2 + 2x*y**3 + y**4 + 2x**2*y +2x*y**2 + 2y**3 + 5 x**2 + 5*x*y + 6*y**2 + 5y + 9);**

```
  2              2          2
{u  + 5*u + 9,u=x  + x*y + y  + y}
```

Simplification with side relations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter **??**) Maple (see chapter **??**) Mathematica (see chapter **??**)

- load the module compact for polynomial compactification: simplification of a polynomial in which the variables are restricted by side relations

  load_package compact;

  com:=(1-(sin x)2)**5*(1-(cos x)**2)**5*((sin x)**2+(cos x)**2)**5;**

```
                  20        10             20        8              20        6
com := sin(x)  *cos(x)   - 5*sin(x)  *cos(x)  + 10*sin(x)  *cos(x)

                20        4             20        2          20
        - 10*sin(x)  *cos(x)  + 5*sin(x)  *cos(x)  - sin(x)

               18        12             18        10
        + 5*sin(x)  *cos(x)   - 30*sin(x)  *cos(x)

                18        8              18        6
        + 75*sin(x)  *cos(x)  - 100*sin(x)  *cos(x)

                18        4             18        2            18
        + 75*sin(x)  *cos(x)  - 30*sin(x)  *cos(x)  + 5*sin(x)

                16        14             16        12
        + 10*sin(x)  *cos(x)   - 75*sin(x)  *cos(x)

                 16        10              16        8
        + 235*sin(x)  *cos(x)   - 400*sin(x)  *cos(x)

                 16        6              16        4
        + 400*sin(x)  *cos(x)  - 235*sin(x)  *cos(x)

                16        2             16             14        16
        + 75*sin(x)  *cos(x)  - 10*sin(x)   + 10*sin(x)  *cos(x)

                 14        14              14        12
        - 100*sin(x)  *cos(x)   + 400*sin(x)  *cos(x)

                 14        10               14        8
        - 860*sin(x)  *cos(x)   + 1100*sin(x)  *cos(x)

                 14        6              14        4
        - 860*sin(x)  *cos(x)  + 400*sin(x)  *cos(x)

                 14        2             14             12        18
        - 100*sin(x)  *cos(x)  + 10*sin(x)   + 5*sin(x)  *cos(x)

                12        16              12        14
        - 75*sin(x)  *cos(x)   + 400*sin(x)  *cos(x)

                  12        12               12        10
        - 1100*sin(x)  *cos(x)   + 1780*sin(x)  *cos(x)

                  12        8               12        6
        - 1780*sin(x)  *cos(x)  + 1100*sin(x)  *cos(x)

                 12        4             12        2            12
        - 400*sin(x)  *cos(x)  + 75*sin(x)  *cos(x)  - 5*sin(x)
```

$$+ \sin(x)^{10} \cos(x)^{20} - 30 \sin(x)^{10} \cos(x)^{18}$$

$$+ 235 \sin(x)^{10} \cos(x)^{16} - 860 \sin(x)^{10} \cos(x)^{14}$$

$$+ 1780 \sin(x)^{10} \cos(x)^{12} - 2252 \sin(x)^{10} \cos(x)^{10}$$

$$+ 1780 \sin(x)^{10} \cos(x)^{8} - 860 \sin(x)^{10} \cos(x)^{6}$$

$$+ 235 \sin(x)^{10} \cos(x)^{4} - 30 \sin(x)^{10} \cos(x)^{2} + \sin(x)^{10}$$

$$- 5 \sin(x)^{8} \cos(x)^{20} + 75 \sin(x)^{8} \cos(x)^{18}$$

$$- 400 \sin(x)^{8} \cos(x)^{16} + 1100 \sin(x)^{8} \cos(x)^{14}$$

$$- 1780 \sin(x)^{8} \cos(x)^{12} + 1780 \sin(x)^{8} \cos(x)^{10}$$

$$- 1100 \sin(x)^{8} \cos(x)^{8} + 400 \sin(x)^{8} \cos(x)^{6}$$

$$- 75 \sin(x)^{8} \cos(x)^{4} + 5 \sin(x)^{8} \cos(x)^{2}$$

$$+ 10 \sin(x)^{6} \cos(x)^{20} - 100 \sin(x)^{6} \cos(x)^{18}$$

$$+ 400 \sin(x)^{6} \cos(x)^{16} - 860 \sin(x)^{6} \cos(x)^{14}$$

$$+ 1100 \sin(x)^{6} \cos(x)^{12} - 860 \sin(x)^{6} \cos(x)^{10}$$

$$+ 400 \sin(x)^{6} \cos(x)^{8} - 100 \sin(x)^{6} \cos(x)^{6}$$

$$+ 10 \sin(x)^{6} \cos(x)^{4} - 10 \sin(x)^{4} \cos(x)^{20}$$

$$+ 75 \sin(x)^{4} \cos(x)^{18} - 235 \sin(x)^{4} \cos(x)^{16}$$

$$+ 400 \sin(x)^{4} \cos(x)^{14} - 400 \sin(x)^{4} \cos(x)^{12}$$

$$+ 235 \sin(x)^{4} \cos(x)^{10} - 75 \sin(x)^{4} \cos(x)^{8}$$

$$+ 10 \sin(x)^{4} \cos(x)^{6} + 5 \sin(x)^{2} \cos(x)^{20}$$

```
              2       18              2       16
- 30*sin(x) *cos(x)   + 75*sin(x) *cos(x)

               2       14              2       12
- 100*sin(x) *cos(x)   + 75*sin(x) *cos(x)

              2       10             2       8         20
- 30*sin(x) *cos(x)   + 5*sin(x) *cos(x)  - cos(x)

         18             16             14            12
+ 5*cos(x)   - 10*cos(x)   + 10*cos(x)   - 5*cos(x)

       10
+ cos(x)
```

compact(com,cos x^2+sin x^2=1);

```
      10        10
sin(x)  *cos(x)
```

Grobner bases

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.2) Derive (see chapter ??) Macsyma (see chapter 4.3.2) Maple (see chapter 4.4.2) Mathematica (see chapter 4.5.2)

- load the module groebner for dealing with Grobner bases

  load_package groebner;

  polys := 45*p + 35*s - 165*b - 36, 35*p + 40*z + 25*t - 27*s, 15*w + 25*p*s + 30*z - 18*t - 165*b2, - 9*w + 15*p*t + 20*z*s, w*p + 2*z*t - 11*b**3, 99*w - 11*s*b + 3*b**2, b**2 + 33/50*b + 2673/10000$**

  vars := w,p,z,t,s,b$

  groebner(polys,vars);

  ```
  {60000*w + 9500*b + 3969,

   1800*p - 3100*b - 1377,

   18000*z + 24500*b + 10287,

   750*t - 1850*b + 81,

   200*s - 500*b - 9,

          2
   10000*b  + 6600*b + 2673}
  ```

- solving a system of polynomial equations by Grobner bases

  groesolve(polys,vars);

```
     - 148*sqrt(11)*i - 461
{{t=-------------------------,
              500

    190*sqrt(11)*i - 139
  w=----------------------,
            10000

    490*sqrt(11)*i - 367
  z=----------------------,
             3000

     - 62*sqrt(11)*i + 59
  p=-----------------------,
              300

    3*( - 5*sqrt(11)*i - 13)
  s=--------------------------,
               50

    3*( - 4*sqrt(11)*i - 11)
  b=--------------------------},
              100

    148*sqrt(11)*i - 461
 {t=----------------------,
             500

     - 190*sqrt(11)*i - 139
  w=-------------------------,
              10000

     - 490*sqrt(11)*i - 367
  z=-------------------------,
              3000

    62*sqrt(11)*i + 59
  p=--------------------,
            300

    3*(5*sqrt(11)*i - 13)
  s=-----------------------,
              50

    3*(4*sqrt(11)*i - 11)
  b=-----------------------}}
              100
```

- (see chapter4.6.4)

4.6.3 Rational functions

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.3) Derive (see chapter 4.2.3) Macsyma (see chapter 4.3.3) Maple (see chapter 4.4.3) Mathematica (see chapter 4.5.3)

- integration

int((3*a*b^2-5*a^2*b)/(a^4-2),a);

```
                     1/4        a          1/4      1/4
(sqrt(2)*b*( - 10*2    *atan(------) + 5*2   *log(2    + a)
                                1/4
                               2

                    1/4         1/4                 1/4
             - 5*2    *log( - 2    + a) + 3*log(2    + a)*b

                           1/4                           2
             + 3*log( - 2    + a)*b - 3*log(sqrt(2) + a )*b))/8
```

- verification by differentiation

  df(ws,a);

```
 a*b*( - 5*a + 3*b)
--------------------
        4
       a  - 2
```

- partial fraction decomposition

  pf((10x^2-11x-6)/(x^3-x^2-2x),x);

```
    5       2     3
{-------,-------,---}
  x + 1   x - 2   x
```

4.6.4 Solving equations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4)

Linear systems

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4)

solve(2*x1+x2+3*x3-9,x1-2*x2+x3+2,3*x1+2*x2+2*x3-7, x1,x2,x3);

```
{{x1=-1,x2=2,x3=3}}
```

Nonlinear equations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter 4.2.4) Macsyma (see chapter 4.3.4) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4)

- using decomposition to solve high degree polynomials

  solve(x8-8*x**7+34*x**6-92*x**5+175*x**4-236*x**3+226*x**2-140*x+46);**

```
Unknown: x

   sqrt( - sqrt( - 4*sqrt(3) - 3) - 3)*sqrt(2) + 2
{x=------------------------------------------------,
                        2

    - sqrt( - sqrt( - 4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=---------------------------------------------------,
                         2

   sqrt( - sqrt(4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=---------------------------------------------,
                       2

    - sqrt( - sqrt(4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=------------------------------------------------,
                         2

   sqrt(sqrt( - 4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=---------------------------------------------,
                       2

    - sqrt(sqrt( - 4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=------------------------------------------------,
                         2

   sqrt(sqrt(4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=------------------------------------------,
                      2

    - sqrt(sqrt(4*sqrt(3) - 3) - 3)*sqrt(2) + 2
 x=---------------------------------------------}
                       2
```

- multiple use of inversion functions
  solve(log(acos(asin(x(2/3)-b)-1))+2,x);**

```
             1             3/2
{x=(sin(cos(----) + 1) + b)    ,
              2
             e

                1             3/2
 x= - (sin(cos(----) + 1) + b)    }
                 2
                e
```

Nonlinear systems

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.4) Derive (see chapter **??**) Macsyma (see chapter **??**) Maple (see chapter 4.4.4) Mathematica (see chapter 4.5.4)

- solve polynomial systems by the use of (see chapter2.7)

solve((alpha * c1 - beta * c12 - gamma*c1*c2 + epsilon*c3, -gamma*c1*c2 + (epsilon+theta)*c3 -eta *c2, gamma*c1*c2 + eta*c2 - (epsilon+theta) * c3), (c3,c2,c1));**

```
{{c1=arbcomplex(12),

  c2=(c1*( - c1*beta*epsilon - c1*beta*theta + alpha*epsilon

           + alpha*theta))/(c1*gamma*theta - epsilon*eta),

  c3=(c1*

              2
      ( - c1 *beta*gamma + c1*alpha*gamma - c1*beta*eta + alpha*eta)

      )/(c1*gamma*theta - epsilon*eta)}}
```

- the value of the operator arbcomplex is an arbitrary complex number
- (see chapter4.6.2)

4.6.5 Analytical operations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

Limits

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

limit(sin(x)/x,x,0);

```
1
```

limit((3*sin(pi*x) - sin(3*pi*x))/x^3,x,0);

```
    3
4*pi
```

limit((2x+5)/(3x-2),x,infinity);

```
 2
---
 3
```

Taylor series

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

- load the module taylor for computing Taylor series

  load_package taylor;

  taylor (ex, x, 0, 4);**

```
            1  2    1  3    1   4      5
1 + x + ---*x  + ---*x  + ----*x  + O(x )
         2        6        24
```

taylor (e(x+y), x, 0, 2, y, 0, 2);**

```
            1  2                          3  3
1 + y + ---*y  + x + y*x + (4 terms) + O(x ,y )
         2
```

taylorcombine (ws2);**

```
                2                             3  3
1 + 2*y + 2*y  + 2*x + 4*y*x + (4 terms) + O(x ,y )
```

Summation and Products

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

sum(m2*x**m,m,1,n);**

```
     n  2  2      n  2      n  2     n         n      n      n
(x*(x *n *x  - 2*x *n *x + x *n  - 2*x *n*x + 2*x *n + x *x + x  - x

         3      2
 - 1))/(x  - 3*x  + 3*x - 1)
```

sum(cos((2*m-1)*pi/(2*n+1)),m,1,r);

```
      2*r*pi
  sin(---------)
      2*n + 1
------------------
         pi
 2*sin(---------)
       2*n + 1
```

prod(e(sin(m*x)),m,1,n);**

```
      cos(x/2)/(2*sin(x/2))
     e
----------------------------------
   cos((2*n*x + x)/2)/(2*sin(x/2))
  e
```

for all n,m such that fixp m let factorial(n+m)=if m ¿ 0 then factorial(n+m-1)*(n+m) else factorial(n+m+1)/(n+m+1);

sum(m*2m/factorial(m+2),m,1,n);**

```
       n                 2
  - 2*2  + factorial(n)*n  + 3*factorial(n)*n + 2*factorial(n)
----------------------------------------------------------------
                                 2
                  factorial(n)*(n  + 3*n + 2)
```

Integration

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter 4.2.5) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

int(x2*(a+b*x)**p,x);**

```
          p     3      2            2  2  2      2    2    3  2  3
((a + b*x) *(2*a  - 2*a *b*p*x + a*b *p *x  + a*b *p*x  + b *p *x

                   3    3      3  3     3   3      2
              + 3*b *p*x  + 2*b *x ))/(b *(p  + 6*p  + 11*p + 6))
```

int(x2*log(x**2+a**2),x);**

```
            x    3          2    2   3      2        3
  - 6*atan(---)*a  + 3*log(a  + x )*x  + 6*a *x - 2*x
            a
-------------------------------------------------------
                           9
```

int(x*dx*sin x,x);**

```
  x               3                  2
(d *(sin(x)*log(d) *x - sin(x)*log(d)  + sin(x)*log(d)*x + sin(x)

                     2                                         4
      - cos(x)*log(d) *x + 2*cos(x)*log(d) - cos(x)*x))/(log(d)

              2
    + 2*log(d)  + 1)
```

int(x*sqrt(a+b*x)p,x);**

```
            p/2        2              2    2      2  2
 2*(a + b*x)   *( - 2*a  + a*b*p*x + b *p*x  + 2*b *x )
--------------------------------------------------------
                    2   2
                   b *(p  + 6*p + 8)
```

int(2*x*e(x**2)*log(x)+e**(x**2)/x+(log(x)-2)/(log(x)**2+x)**2+ ((2/x)*log(x)+(1/x)+1)/(log(x)**2+x),x**

```
   2             2
  x        3    x                      2            2
(e  *log(x)  + e  *log(x)*x + log(log(x)  + x)*log(x)
```

```
                 2                          2
  + log(log(x)  + x)*x - log(x))/(log(x)  + x)
```

Ordinary differential equations

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter **??**) Macsyma (see chapter 4.3.5) Maple (see chapter 4.4.5) Mathematica (see chapter 4.5.5)

- load the module odesolve for the analytical solution of ordinary differential equations

  load_package odesolve;

  depend y,x;

  odesolve(df(y,x) + y * sin x/cos x - 1/cos x,y,x);

  ```
  {y=arbconst(1)*cos(x) + sin(x)}
  ```

- Bernoulli equation

  odesolve(x*(1-x2)*df(y,x) + (2*x**2 -1)*y - x**3*y**3,y,x);**

  ```
                             5
     1    5*arbconst(2) + 2*x
   {----=----------------------}
     2          4      2
    y        5*x  - 5*x
  ```

  solve(ws,y);

  ```
                2
        sqrt(x  - 1)*sqrt(5)*x
   {y=---------------------------,
                               5
       sqrt(5*arbconst(2) + 2*x )

                2
       - sqrt(x  - 1)*sqrt(5)*x
    y=---------------------------}
                               5
       sqrt(5*arbconst(2) + 2*x )
  ```

  odesolve(df(y,x,2)+4*df(y,x)+4*y-x*exp(x),y,x);

  ```
         3*x         3*x
      3*e   *x - 2*e    + 27*arbconst(3)*x + 27*arbconst(3)
   {y=----------------------------------------------------------}
                                  2*x
                              27*e
  ```

Substitutions - pattern matching

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.5) Derive (see chapter ??) Macsyma (see chapter 4.3.5) Maple (see chapter ??) Mathematica (see chapter ??)

factor cos,sin;

(a1*cos(wt) + a3*cos(3*wt) + b1*sin(wt) + b3*sin(3*wt))3 where cos(˜x)*cos(˜y) =¿ (cos(x+y)+cos(x-y))/2, cos(˜x)*sin(˜y) =¿ (sin(x+y)-sin(x-y))/2, sin(˜x)*sin(˜y) =¿ (cos(x-y)-cos(x+y))/2, cos(˜x)**2 =¿ (1+cos(2*x))/2, sin(˜x)**2 =¿ (1-cos(2*x))/2;**

```
                  2       2
(cos(9*wt)*a3*(a3  - 3*b3 )

                      2         2
 + 3*cos(7*wt)*(a1*a3  - a1*b3  - 2*a3*b1*b3) + 3*cos(5*wt)*

    2           2                    2        2
 (a1 *a3 + a1*a3  - 2*a1*b1*b3 - a1*b3  - a3*b1  + 2*a3*b1*b3) +

             3       2             2       3          2          2
 cos(3*wt)*(a1  + 6*a1 *a3 - 3*a1*b1  + 3*a3  + 6*a3*b1  + 3*a3*b3 )

 + 3*cos(wt)*

    3     2             2        2                       2        2
 (a1  + a1 *a3 + 2*a1*a3  + a1*b1  + 2*a1*b1*b3 + 2*a1*b3  - a3*b1 )

                       2     2
  + sin(9*wt)*b3*(3*a3  - b3 )

                                2           2
  + 3*sin(7*wt)*(2*a1*a3*b3 + a3 *b1 - b1*b3 ) + 3*sin(5*wt)*

    2                                2        2           2
 (a1 *b3 + 2*a1*a3*b1 + 2*a1*a3*b3 - a3 *b1 - b1 *b3 + b1*b3 ) +

               2          2          2        3       2          3
 sin(3*wt)*(3*a1 *b1 + 6*a1 *b3 + 3*a3 *b3 - b1  + 6*b1 *b3 + 3*b3 )

 + 3*sin(wt)*

    2        2                      2        3     2            2
 (a1 *b1 + a1 *b3 - 2*a1*a3*b1 + 2*a3 *b1 + b1  - b1 *b3 + 2*b1*b3 )

 )/4
```

operator integrate;

linear integrate;

let integrate(˜x˜p,x) =¿ x**(p+1)/(p+1) when df(p,x)=0, integrate(˜x,x) =¿ x**2/2, integrate(1,˜x) =¿ x$**

integrate(aˆ2*b+aˆb+3*a-5,a);

```
       b      2  2      2
 a*(6*a  + 2*a *b  + 2*a *b + 9*a*b + 9*a - 30*b - 30)
-----------------------------------------------------
                      6*(b + 1)
```

integrate(a^(a+1),a);

```
             a
integrate(a *a,a)
```

4.6.6 Matrices

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.6) Derive (see chapter 4.2.6) Macsyma (see chapter 4.3.6) Maple (see chapter 4.4.6) Mathematica (see chapter 4.5.6)

matrix xx,yy;

let xx= mat((a11,a12),(a21,a22)), yy= mat((y1),(y2));

xx;

```
[a11  a12]
[        ]
[a21  a22]
```

det xx;

```
a11*a22 - a12*a21
```

zz:= xx(-1)*yy;**

```
      [  - a12*y2 + a22*y1 ]
      [--------------------]
      [ a11*a22 - a12*a21  ]
zz := [                    ]
      [  a11*y2 - a21*y1   ]
      [------------------- ]
      [ a11*a22 - a12*a21  ]
```

1/xx2;**

```
                             2
                  a12*a21 + a22
mat((------------------------------------------,
        2    2                           2    2
     a11 *a22  - 2*a11*a12*a21*a22 + a12 *a21

                 - a12*(a11 + a22)
    ------------------------------------------),
        2    2                           2    2
     a11 *a22  - 2*a11*a12*a21*a22 + a12 *a21

                 - a21*(a11 + a22)
    (------------------------------------------,
        2    2                           2    2
     a11 *a22  - 2*a11*a12*a21*a22 + a12 *a21
```

```
                     2
                  a11  + a12*a21
     -------------------------------------------))
          2    2                          2    2
      a11 *a22  - 2*a11*a12*a21*a22 + a12 *a21
```

```
mateigen(mat((2,-1,1),(0,1,1),(-1,1,1)),et);
```

```
    {{et - 1,2,

      [arbcomplex(13)]
      [              ]
      [arbcomplex(13)]
      [              ]
      [      0       ]

      },

     {et - 2,1,

      [      0       ]
      [              ]
      [arbcomplex(14)]
      [              ]
      [arbcomplex(14)]

      }}
```

4.6.7 Code generation

In Reduce

For comparison with other CAS choose from: Axiom (see chapter ??) Derive (see chapter ??) Macsyma (see chapter 4.3.7) Maple (see chapter ??) Mathematica (see chapter ??)

- load the module gentran for the generation of numerical programs

```
load_package gentran;

gentranout gentst$

matrix m(3,3)$

m(1,1) := 18*cos(q3)*cos(q2)*m30*p**2 - 9*sin(q3)**2*p**2*m30 - sin(q3)**2*j30y + sin(q3)**2*j30z
+ p**2*m10 + 18*p**2*m30 + j10y + j30y$

m(2,1) := m(1,2) := 9*cos(q3)*cos(q2)*m30*p**2 - sin(q3)**2*j30y + sin(q3)**2*j30z - 9*sin(q3)**2*m30*p*
+ j30y + 9*m30*p**2$

m(3,1) := m(1,3) := -9*sin(q3)*sin(q2)*m30*p**2$

m(2,2) := -sin(q3)**2*j30y + sin(q3)**2*j30z - 9*sin(q3)**2* m30*p**2 + j30y + 9*m30*p**2$

m(3,2) := m(2,3) := 0$

m(3,3) := 9*m30*p**2 + j30x$

gentranlang!* := 'fortran$

fortlinelen!* := 72$

gentran literal "c", cr!*, "c", tab!*, "*** compute values for matrix m ***", cr!*, "c", cr!*$

for j:=1:3 do for k:=j:3 do gentran m(j,k) ::=: m(j,k)$
```

gentran literal "c", cr!*, "c", tab!*, "* compute values for inverse matrix ***",cr!*, "c", cr!*$**

share var$

for j:=1:3 do for k:=j:3 do if m(j,k) neq 0 then ¡¡ var := tempvar nil; markvar var; m(j,k) := var; m(k,j) := var; gentran eval(var) := m(eval(j),eval(k)) ¿¿$

- matrix m contains

m := m;

```
        [t0  t1  t2]
        [          ]
   m := [t1  t3  0 ]
        [          ]
        [t2  0   t4]
```

matrix mxinv(3,3)$

mxinv := m(-1)$**

for j:=1:3 do for k:=j:3 do gentran mxinv(j,k) ::=: mxinv(j,k)$

gentran for j:=1:3 do for k:=j+1:3 do ¡¡ m(k,j) := m(j,k); mxinv(k,j) := mxinv(j,k) ¿¿$

gentranshut gentst;

Generated FORTRAN program

Generated program

```
c
c      *** compute values for matrix m ***
c
      m(1,1)=-(9.0*sin(real(q3))**2*p**2*m30)-(sin(real(q3))**2*j30y)+
     . sin(real(q3))**2*j30z+18.0*cos(real(q3))*cos(real(q2))*p**2*m30+
     . 18.0*p**2*m30+p**2*m10+j30y+j10y
      m(1,2)=-(9.0*sin(real(q3))**2*p**2*m30)-(sin(real(q3))**2*j30y)+
     . sin(real(q3))**2*j30z+9.0*cos(real(q3))*cos(real(q2))*p**2*m30+
     . 9.0*p**2*m30+j30y
      m(1,3)=-(9.0*sin(real(q3))*sin(real(q2))*p**2*m30)
      m(2,2)=-(9.0*sin(real(q3))**2*p**2*m30)-(sin(real(q3))**2*j30y)+
     . sin(real(q3))**2*j30z+9.0*p**2*m30+j30y
      m(2,3)=0.0
      m(3,3)=9.0*p**2*m30+j30x
c
c      *** compute values for inverse matrix ***
c
      t0=m(1,1)
      t1=m(1,2)
      t2=m(1,3)
      t3=m(2,2)
      t4=m(3,3)
      mxinv(1,1)=-(t3*t4)/(t1**2*t4+t2**2*t3-(t0*t3*t4))
      mxinv(1,2)=(t1*t4)/(t1**2*t4+t2**2*t3-(t0*t3*t4))
      mxinv(1,3)=(t2*t3)/(t1**2*t4+t2**2*t3-(t0*t3*t4))
      mxinv(2,2)=(t2**2-(t0*t4))/(t1**2*t4+t2**2*t3-(t0*t3*t4))
      mxinv(2,3)=-(t1*t2)/(t1**2*t4+t2**2*t3-(t0*t3*t4))
      mxinv(3,3)=(t1**2-(t0*t3))/(t1**2*t4+t2**2*t3-(t0*t3*t4))
```

```
      do 25001 j=1,3
          do 25002 k=j+1,3
              m(k,j)=m(j,k)
              mxinv(k,j)=mxinv(j,k)
25002     continue
25001 continue
```

4.6.8 Graphics

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7)

2D Graphics

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7)

- function of one parameter

  on demo,rounded,numval;

  plot(sin(e^x),x=(0 .. pi), xlabel="Time", ylabel="Signal");

- graph of several functions (Bessel functions $J(n,x), n = 0, 2, 5$)

  load_package specfn;

  plot(besselj(0,x),besselj(2,x),besselj(5,x),x=(0 .. 10), title="Bessel Functions BesselJ(n,x)", ylabel="Value");

3D Graphics

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7)

- graph of a function with 2 parameters

  plot_xmesh:=plot_ymesh:=35;

  plot(sin(pi*sin x+y),x=(-3 .. 3),y=(-3 .. 3),hidden3d,view="30,40");

- graph of another function with 2 parameters

  plot_xmesh:=plot_ymesh:=50;

  plot(tan(x*y), x=(-2*pi/3 .. 2*pi/3), y=(-2*pi/3 .. 2*pi/3), z=(-5 .. 5), hidden3d,view="30,30");

Parametric plots

In Reduce

For comparison with other CAS choose from: Axiom (see chapter 4.1.7) Derive (see chapter 4.2.7) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7)

- surface defined parametrically

  dd:=pi/10;

  w:=for u:=0 step dd until 2*pi collect for v:=-pi/2 step dd until pi/2 collect sin v, sin(2*v)*sin u, sin(2*v)*cos u$

  plot w;

Contour maps

In Reduce

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.8) Maple (see chapter 4.4.7) Mathematica (see chapter 4.5.7)

- contour map of a function with 2 parameters

  plot_xmesh:=plot_ymesh:=40;

  plot(sin(x+cos y), x=(0 .. 3*pi/2), y=(0 .. 3*pi/2), contour,nosurface,view="0,0");

- contour map of another function with 2 parameters

  plot_xmesh:=plot_ymesh:=80;

  plot(tan(x*y), x=(-pi .. pi), y=(-pi .. pi), z=(-3 .. 3), contour,nosurface,view="0,0");

- contour map of yet another function with 2 parameters

  plot_xmesh:=plot_ymesh:=30;

  plot(e^(-sqrt(x^2+y^2))*cos(atan(x/y)), x=(-1 .. 1), y=(-1 .. 1), contour,nosurface,view="0,0");

4.6.9 Graphical presentation of formulas

In Reduce

For comparison with other CAS choose from: Axiom (see chapter **??**) Derive (see chapter **??**) Macsyma (see chapter 4.3.9) Maple (see chapter 4.4.8) Mathematica (see chapter 4.5.8)

Chapter 5

Applications of computer algebra

5.1 Classical application areas

- celestial mechanics
  - calculation of orbits, gravitational fields
  - Fourier, Poisson series

$$\sum_i \left[P_i \sin\left(\sum_j \alpha_{ij}\Phi_j\right) + Q_i \cos\left(\sum_j \beta_{ij}\Phi_j\right)\right]$$

  - Delaunay's (1867) calculation of the orbit of the Moon took him 20 years (including physical effects like non-symmentry of the earth and influence of the Sun); recalculation on a small computer (1980) took 20 hours of CPU time; the check found only a few mistakes in his hand calculations in some of the high order terms
  - systems TRIGMAN (1970), CAMAL (1975)
- general theory of relativity
  - calculations with various metrics
  - systems CAMAL (1975), SHEEP (1977), general purpose systems
- quantum electro-dynamics
- high energy physics
  - interaction of particles, Dirac matrices, Feynman diagrams, calculation of integrals
  - systems REDUCE (1968), SCHOONSHIP (1971)

5.2 Other application areas

- physics
  - plasma physics, physics of fluids
  - electron optics, non-linear optics
  - molecular physics
  - electronics
  - mechanics
- mathematics

  - number theory
  - group theory
  - computational geometry
  - numerical analysis

- chemistry
- biology
- robotics
- economy

5.3 Case study 1. - Perturbation methods

5.3.1 Celestial mechanics - nonlinear algebraic equations

- from J.P. Fitch, International Conference on Computer Algebra and its Applications in Theoretical Physics, p. 262-275, Dubna, (1985).
- solution of a series in a small parameter

$$y^2 = 1 - \varepsilon$$

- zeroth order solution $y_0 = \pm 1$, n-th order solution y_n

$$y = y_n + \eta + O(\varepsilon^{n+2})$$
$$\eta = \varepsilon^{n+1}\vartheta$$
$$(y_n + \eta + O(\varepsilon^{n+2}))^2 = 1 - \varepsilon$$
$$y_n^2 + \eta^2 + 2\eta y_n + O(\varepsilon^{n+2}) = 1 - \varepsilon$$
$$y_n^2 + 2\eta y_0 + O(\varepsilon^{n+2}) = 1 - \varepsilon$$
$$\eta = \frac{1}{2}(1 - \varepsilon - y_n^2) + O(\varepsilon^{n+2})$$

- resulting iteration scheme

$$\begin{aligned} y_0 &= 1 \\ y_{n+1} &= y_n + \frac{1}{2}(1 - \varepsilon - y_n^2) \end{aligned}$$

- another example: the Kepler equation

$$E = u + \varepsilon \sin E$$

- zeroth order solution $E_0 = u, n$-th order solution E_n

$$E = E_n + \eta + O(\varepsilon^{n+2})$$
$$E_n + \eta = u + \varepsilon \sin(E_n + \eta) + O(\varepsilon^{n+2})$$
$$E_n + \eta = u + \varepsilon(\sin E_n \cos \eta + \cos E_n \sin \eta) + O(\varepsilon^{n+2})$$
$$E_n + \eta = u + \varepsilon \sin E_n + O(\varepsilon^{n+2})$$
$$E_n = u + A_n$$

$$A_{n+1} = \varepsilon \sin(u + A_n)$$

$$A_{n+1} = \varepsilon(\sin u \cos A_n + \cos u \sin A_n)$$

$$\begin{aligned} A_0 &= 0 \\ A_{n+1} &= \varepsilon \sin u \left(1 - \frac{A_n^2}{2!} + \frac{A_n^4}{4!} - \ldots\right)_n \\ &\quad +\varepsilon \cos u \left(A_n - \frac{A_n^3}{3!} + \frac{A_n^5}{5!} - \ldots\right)_n \end{aligned}$$

- final solution in the form of Fourier series

$$A_n = \sum_{j=0}^{n} P_j(\varepsilon) \sin(ju)$$

- Fourier series are used by specialized computer algebra systems such as CAMAL and TRIGMAN

5.3.2 Mechanics - nonlinear ordinary differential equations

- from D.M. Klimov, V.M. Rudenko, Computer Algebra Methods in Mechanics, Nauka, (1989).
- nonlinear periodic movement, nonlinear ordinary differential equation

$$U'' + \omega_0^2 U = \varepsilon f(U', U, \varepsilon)$$

- new time variable $\tau = \omega t$, ω is an unknown frequency

$$\omega^2 U'' + \omega_0^2 U = \varepsilon f(\omega U', U, \varepsilon)$$

- Poincare perturbation method—expansion into a power series

$$\begin{aligned} U &= \sum_{i=0}^{\infty} U_i(\tau)\varepsilon^i \\ \omega &= \sum_{i=0}^{\infty} \omega_i \varepsilon^i \end{aligned}$$

- example: the van der Pol equation

$$U'' + U - \varepsilon\, U'(1 - U^2) = 0$$
$$\omega^2 U'' + U - \varepsilon\omega U'(1 - U^2) = 0$$

- after substituting power series in

$$\begin{aligned} \varepsilon^0 : &\quad \omega_0^2 U_0'' + U_0 = 0 \\ \varepsilon^1 : &\quad \omega_0^2 U_1'' + U_1 = -2\omega_0\omega_1 U_0'' + \omega_0 U_0'(1 - U_0^2) \\ \varepsilon^2 : &\quad \omega_0^2 U_2'' + U_2 = -(2\omega_0\omega_2 + \omega_1^2)U_0'' - 2\omega_0\omega_1 U_1'' \\ &\quad +\omega_1 U_0'(1 - U_0^2) - 2\omega_0 U_0' U_1 U_0 + \omega_0 U_1'(1 - U_0^2) \\ &\vdots \end{aligned}$$

- initial condition
- zeroth order solution
- first order solution

$$U_1'' + U_1 = 2\omega_1 A_0 \cos\,\tau - A_0 \sin\,\tau(1 - A_0^2 \cos\,\tau)$$

- after Fourier expansion

$$\begin{aligned} U_1'' + U_1 &= 2\omega_1 A_0 \cos\,\tau + A_0 \sin\,\tau \left(\frac{A_0^2}{4} - 1\right) \\ &+ \frac{A_0^3}{4} \sin(3\tau) \end{aligned}$$

- no secular terms in the solution require zero coefficients of sinand coson the right hand side

$$2\omega_1 A_0 = 0\;, \quad A_0 \left(\frac{A_0^2}{4} - 1\right) = 0$$

- solution

$$\begin{aligned} U_0 &= 2\cos\,\tau, \omega_0 = 1 \\ U_1 &= A_1 \cos\,\tau + B_1 \sin\,\tau - \frac{1}{4}\sin(3\tau)\;,\;\; \omega_1 = 0 \end{aligned}$$

- using the initial condition
- second order equation

$$\begin{aligned} U_2'' + U_2 &= 2A_1 \sin\,\tau + (4\omega_2 + \frac{1}{4}) \cos\,\tau - \frac{2}{3}\cos(3\tau) \\ &+ 3A_1 \sin(3\tau) + \frac{5}{4}\cos(5\tau) \\ \omega_2 &= -\frac{1}{16}\;, \quad A_1 = 0 \end{aligned}$$

- first order solution

$$U_1 = \frac{3}{4}\sin\,\tau - \frac{1}{4}\sin(3\tau)\;, \quad \omega_1 = 0$$

- second order solution

$$U_2 = A_2 \cos\,\tau + B_2 \sin\,\tau + \frac{3}{16}\cos(3\tau) - \frac{5}{96}\cos(5\tau)$$

- etc. for higher orders

5.3.3 Quantum mechanics - eigenvalue problem

- from T.C. Scott, R.A. Moore, M.B. Monagan, G.J. Fee, E.R. Vrscay, J. of Comp. Phys., Vol. 87, No. 2, p.366-395, (1990).
- Rayleigh-Schrodinger perturbation theory
- Hamiltonian, solvable part $H_0$and perturbative part H_1

$$H = H_0 + \lambda H_1$$

- eigenvalues (energies) and eigenfunctions are expanded into power series in λ

$$\begin{aligned} E &= \sum_{p=0}^{\infty} \lambda^p E_p \\ \Phi &= \sum_{p=0}^{\infty} \lambda^p \Phi^p \end{aligned}$$

- eigenvalue problem

$$H\Phi = E\Phi$$

- hierarchy of equations

$$\begin{aligned} \lambda^0 : &\quad H_0\Phi^0 = E_0\Phi^0 \\ \lambda^1 : &\quad H_0\Phi^1 + H_1\Phi^0 = E_0\Phi^1 + E_1\Phi^0 \\ \lambda^2 : &\quad H_0\Phi^2 + H_1\Phi^1 = E_0\Phi^2 + E_1\Phi^1 + E_2\Phi^0 \\ \vdots & \\ \lambda^j : &\quad H_0\Phi^j + H_1\Phi^{j-1} = \sum_{i=0}^{j} E_i\Phi^{j-i} \end{aligned}$$

- normalizations

$$\begin{aligned} &\quad < \Phi|\Phi >= 1 \\ \lambda^0 : &\quad < \Phi^0|\Phi^0 >= 1 \\ \lambda^1 : &\quad < \Phi^0|\Phi^1 > + < \Phi^1|\Phi^0 >= 0 \\ \vdots & \\ \lambda^j : &\quad \sum_{i=0}^{j} < \Phi^i|\Phi^{j-i} >= 1 \end{aligned}$$

- separable problems, second order ordinary differential equations (ODEs), inhomogenous ODE

$$\left[\frac{d^2}{d\,t^2} + P(t)\frac{d}{d\,t} + Q(t)\right] y(t) = f(t)$$

- one solution $y_1(t)$of the homogeneous equation and another linearly independent solution $y_2(t)$in which the degree decreases

$$\begin{aligned} y_2(t) &= y_1(t)\int \frac{W(t)}{y_1^2} d\,t \\ W(t) &= W(y_1, y_2) = \exp\left(-\int P(t) d\,t\right) \end{aligned}$$

- particular solution of the inhomogenous equation, variation of parameters

$$
\begin{aligned}
y_p &= u(t)y_1(t) + v(t)y_2(t) \\
u(t) &= -\int \frac{y_2(t)f(t)}{W} d\,t \;, \quad v(t) = \int \frac{y_1(t)f(t)}{W} d\,t
\end{aligned}
$$

- the general solution of the inhomogenous equation is

$$
y = u_p + C_1 y_1 + C_2 y_2
$$

- one particle Dirac equation

$$
(c\vec{\alpha} \cdot \vec{p} + \beta m_0 c^2 + IV)\Phi = E\Phi
$$

- Hamiltonian

$$
\begin{aligned}
H &= H_0 + \lambda H_1 \\
H_0 &= c\vec{\alpha} \cdot \vec{p} + \beta m_0 c^2 + \frac{1}{2}(I + \beta)V \\
H_1 &= \frac{1}{2\alpha^2}(I - \beta)V \;, \quad \lambda = \alpha^2
\end{aligned}
$$

- for hydrogen-like atoms

$$
E_0 = m_0 c^2 - Z^2 \frac{R_{\mathcal{H}}}{n^2} \left(1 + \frac{Z^2 \alpha^2}{4n^2} \right)^{-1}
$$

- two particle Dirac equation
- necessary algebraic operations
  - indefinite integrals
  - definite integrals
  - pattern matching, zero recognition
  - manipulation and simplification of large sums

5.4 Case study 2. - General theory of relativity

5.4.1 Basic notions

- Riemann geometry, metric tensor g_{ij}, covariant components g_{ij}

$$
d\,s^2 = \sum_{i,j} g_{ij} d\,x^i d\,x^j
$$

- contravariant components $g^{ij}$of metric tensor

$$
\sum_i g_{ij} g^{ik} = \delta_{jk}
$$

- Christoffel symbols of the first kind

$$
\Gamma_{ikl} = \frac{1}{2} \left(\frac{\partial\, g_{ik}}{\partial\, x^l} + \frac{\partial\, g_{il}}{\partial\, x^k} - \frac{\partial\, g_{il}}{\partial\, x^k} \right)
$$

- Christoffel symbols of the second kind

$$\Gamma^i_{kl} = \sum_j g^{ij}\Gamma_{jkl}$$

- Riemann curvature tensor

$$R^i_{jkl} = \frac{\partial\,\Gamma^i_{jl}}{\partial\,x^k} - \frac{\partial\,\Gamma^i_{jk}}{\partial\,x^l} + \sum_m \Gamma^i_{mk}\Gamma^m_{jl} - \sum_m \Gamma^i_{ml}\Gamma^m_{jk}$$

- Ricci tensor

$$R_{ij} = \sum_k R^k_{jki}$$

- for a general metric in 4D, there are 9 990 diagonal and 13 280 off-diagonal terms
- Ricci scalar

$$R = \sum_{i,j} g^{ij} R_{ij}$$

- Einstein vacuum equations

$$R_{ij} = \Lambda g_{ij}$$

- system of second order nonlinear partial differential equations with cosmological constant Λ

5.4.2 Examples

- diagonal metrics, e.g., spherically symmetric metrics

$$\begin{aligned} d\,s^2 &= -\exp(p(r))d\,r^2 - r^2 d\,\Theta^2 \\ &\quad -r^2\sin^2\Theta\;d\,\Theta^2 + \exp(q(r))d\,t^2 \end{aligned}$$

- non-diagonal metrics, e.g., Bondi metrics

$$\begin{aligned} d\,s^2 &= \frac{V\exp(2\beta)}{r} - U^2r^2\exp(2\gamma)d\,u^2 \\ &\quad +2\exp(2\beta)d\,u\;d\,r + 2Ur^2\exp(2\gamma)d\,u\;d\,\Theta \\ &\quad -r^2(\exp(2\gamma)d\,\Theta^2 + \exp(-2\gamma)\sin^2\Theta\;d\,\Phi) \end{aligned}$$

- verification of metric tensor solution g, B. Nielsen and H. Pedersen, SIGSAM Bull.,Vol. 22, 1, Issue 83, p.7-11,(1988)
- coordinate system (u_1, u_2, u_3, u_4)

$$\begin{aligned} d\,s^2 &= \frac{u_4^2}{4}(\alpha\rho_1^2 + \alpha^{-1}\rho_2^2 + \beta\rho_3^2) + \beta^{-1}d\,u_4^2 \\ \rho_1 &= \cos\;u_1\;d\,u_2 + \sin\;u_1\sinh\;u_2\;d\,u_3 \\ \rho_2 &= -\sin\;u_1\;d\,u_2 + \cos\;u_1\sinh\;u_2\;d\,u_3 \\ \rho_3 &= d\,u_2 + \cosh\;u_2\;d\,u_3 \\ \alpha &= \sqrt{\frac{a^4}{u_4^4} - 1}\sqrt{\frac{b^4}{u_4^4} - 1} \\ \beta &= \frac{\sqrt{\frac{a^4}{u_4^4} - 1}}{\sqrt{\frac{b^4}{u_4^4} - 1}} \end{aligned}$$

- solution of Einstein vacuum equation with

$$R_{ij} = 0$$

5.4.3 Other problems and references

- finding a solution
- calculation of properties of any solutions discovered (symmetry)
- problem of equivalence, classification of geometries
- equations including mass (energy, momentum)
- references
  - M.A.H. MacCallum, EUROCAL'87, p.34-43, Springer-Verlag,(1989).
  - M.A.H. MacCallum, Computer Algebra in Physical Research, p. 278-287, World Scientific, (1991).

5.5 Case study 3. - Collision integrals in plasma physics

- collision integrals in plasma fluid equations

5.5.1 Basic notions

- Fokker-Planck equation for distribution function f_s

$$\frac{\partial f_s}{\partial t} + v_i \frac{\partial f_s}{\partial x_i} + \frac{F_{si}}{m_s}\frac{\partial f_s}{\partial v_i} = \left(\frac{\partial f_s}{\partial t}\right)_s$$

- Coulomb collisions with a small angle

$$\left(\frac{\partial f_s}{\partial t}\right)_s = -\frac{\partial (a_i f_s)}{\partial v_i} + \frac{1}{2}\frac{\partial^2 (b_{ij} f_s)}{\partial v_i \, \partial v_j}$$

- Rosenbluth potentials

$$\begin{aligned}
a_i(\vec{v}) &= \sum_t C_{st} \int \frac{f_t(\vec{u}) g_i}{g^3} d^3 u \\
b_{ij}(\vec{v}) &= \sum_t D_{st} \int \frac{f_t(\vec{u})(\delta_i j g^2 - g_i g_j)}{g^3} d^3 u \\
\vec{g} &= \vec{u} - \vec{v}\,, \quad g = |\vec{g}|
\end{aligned}$$

- macroscopic quantities: density N_s, velocity $\vec{u_s}$, pressure p_s, viscous pressure $\overset{\leftrightarrow}{P}_s$, heat flow $\vec{q_s}$

$$\begin{aligned}
q_{si} &= \frac{1}{2} m_s \int c_s^2 c_{si} f_s \, d^3 v \\
\vec{c}_s &= \vec{v} - \vec{u}_s\,, \quad c_s^2 = \vec{c}_s^2
\end{aligned}$$

- collision terms

$$
\begin{aligned}
R_{s1i} &= m_s \int f_s a_i \, d^3v \\
R_{s2} &= m_s \int f_s (c_{si} a_i + 1/2 b_{ii}) d^3v \\
R_{s3ij} &= m_s \int f_s (c_{si} a_j + c_{sj} a_i - 2/3 \delta_{ij} c_{si} a_i \\
& +b_{ij} - 1/3 \delta_{ij} b_{kk}) d^3v \\
R_{s4i} &= 1/2 m_s \int f_s (c_s^2 a_i + 2 c_{si} c_{sk} a_k \\
& +c_{si} b_{kk} + 2 c_{sk} b_{ik}) d^3v
\end{aligned}
$$

- common form of all collision terms

$$
R_{sn} = \sum_t \int \int f_s(\vec{v}) f_t(\vec{u}) \frac{P_{stn}(\vec{v}, \vec{u})}{|\vec{v} - \vec{u}|^3} d^3v \, d^3u
$$

- 13-th moment approximation of the distribution function

$$
\begin{aligned}
f_s(\vec{x}, \vec{v}, t) &= f_{s0}(\vec{x}, \vec{v}, t)(1 + \Phi_s(\vec{x}, \vec{v}, t)) \\
f_{s0} &= \frac{N_s}{\pi^{3/2} a_s^3} \exp\left(-\frac{c_s^2}{a_s^2}\right) \\
a_s^2 &= \frac{2kT_s}{m_s} \\
\Phi_s &= \frac{P_{sij} c_{si} c_{sj}}{p_s a_s^2} + \frac{4 q_{si} c_{si} (c_s^2 - 5/2 a_s^2)}{5 p_s a_s^4} \\
f_s f_t &= f_{s0} f_{t0} (1 + \Phi_s + \Phi_t + \Phi_s \Phi_t)
\end{aligned}
$$

5.5.2 Analytical calculation of collision integrals

- from A. Salat, Plasma Physics, Vol. 17, p. 589-607, (1975).
- common form of the integrals

$$
\int \int \exp\left(-\frac{(\vec{v} - \vec{u}_s)^2}{a_s^2} - \frac{(\vec{u} - \vec{u}_t)^2}{a_t^2}\right) \frac{P(\vec{v}, \vec{u})}{|\vec{v} - \vec{u}|^3} d^3v d^3u
$$

- transformation of variables, new form

$$
\int \int \exp\left(-\frac{C^2}{a^2} - \frac{|\vec{g} - \vec{u}|^2}{\alpha^2}\right) \frac{P(\vec{C}, \vec{g})}{g^3} d^3C d^3g
$$

- sum of 288 terms up to the 9-th degree in v, one term of the 9-th degree in $v$gives after substitutions 19 683 terms, in one part of one collision integral is 259 584 terms (one term is a product appearing in a sum)
- integration over d^3Cderived from

$$
\int \exp(-x^2) d\, x = \sqrt{\pi}
$$

- integration over d^3g; rotation and transformation into spherical coordinates $g, \theta, \phi (z = \cos\theta)$

$$\int_{-1}^{1}\int_{0}^{\infty}\int_{0}^{2\pi} P(\vec{g})d\,\phi \frac{1}{g}\exp\left(-\frac{g^2-2guz}{\alpha^2}\right)d\,g\,d\,z$$

$$g_i = gz\frac{u_i}{u} + g\sqrt{1-z^2}(e_{2i}\cos\,\phi + e_{3i}\sin\,\phi)$$

- integration over $d\phi$, products $\cos\phi, \sin\phi$, elimination of e_{2i}, e_{3i}

$$e_{2i}e_{2j} + e_{3i}e_{3j} = \delta_{ij} - \frac{u_i u_j}{u^2}$$

- integration over dg

$$\int_0^{\infty} g^n \exp\left(-\frac{g^2-2guz}{\alpha^2}\right)d\,g = \alpha\left(\frac{\alpha^2}{2z}\right)^n \frac{\partial^n}{\partial\,u^n}\left[\exp\left(\frac{u^2z^2}{\alpha^2}\right)\operatorname{erfc}\left(-\frac{uz}{\alpha}\right)\right]$$

- integration over dz

$$\begin{aligned} I_k &= \int_{-a}^{a} w^{2k+1}\exp(w^2)\operatorname{erfc}(-w)d\,w \\ w &= \frac{uz}{\alpha}, \quad a = \frac{u}{\alpha} \\ I_0 &= \exp(a^2)\operatorname{erf}(a) - a \\ I_k &= a^{2k}\exp(a^2)\operatorname{erf}(a) - \frac{a^{2k+1}}{2k+1} - kI_{k-1} \end{aligned}$$

- final result contains only elementary functions and the error function
- algebraic program in REDUCE was unable to calculate all the collision integrals and so had to be rewritten into a more efficient symbolic program
- speed up of the symbolic program compared to the algebraic program depends on the size of a collision integral

Size	small	medium	large
Speed up	3x	7x	?

5.6 Case study 4. - Numerical solving of partial differential equations

5.6.1 References

- weighted residual method, replacement of the solution by its approximation, minimization of the residual error

  R. Zippel. Symbolic/numeric techniques in modeling and simulation. In B. Donald, D. Kapur, and J. Mundy, editors, Symbolic and Numerical Computation in Artificial Intelligence. Academic Press, 1992.

- finite element methods

  P. S. Wang. FINGER: A symbolic system for automatic generation of numerical programs in finite element analysis. J. Symb. Comp., 2(3):305-316, 1986.

- finite difference methods

  M. C. Wirth. On the Automation of Computational Physics. PhD thesis, Lawrence Livermore National Laboratory, Livermore, 1980. UCRL-52996.

S. Steinberg and P. J. Roache. Symbolic manipulation and computational fluid dynamics. J. Comp. Phys., 57:251-284, 1985.

G. O. Cook Jr. ALPAL: a program to generate physics simulation codes from natural descriptions. Int. J. Mod. Phys. C, 1(1):1-51, 1990.

R. Liska and L. Drska. FIDE: A REDUCE package for automation of FInite difference method for solving pDE. In S. Watanabe and M. Nagata, editors, Proceedings of ISSAC'90, 169-176, New York, 1990. ACM Press, Addison Wesley.

- numerical code generation

B. L. Gates. A numerical code generation facility for REDUCE. In B. W. Char, editor, Proceedings of SYMSAC '86, 94-99, Waterloo, 1986. ACM.

N. Sharma and P. Wang. Symbolic derivation and automatic generation of parallel routines for finite element analysis. P. Gianni, editor, Proceedings of ISSAC'88, 33-56, Berlin, 1988. Springer-Verlag.

- IRENA-interface to the numerical library (NAG, REDUCE)

M. C. Dewar and M. G. Richardson. Reconciling symbolic and numeric computation in a practical setting. In A. Miola, editor, Proceedings of DISCO'90, 195-204, Berlin, 1990. Springer-Verlag.

5.7 Survey articles on applications

- W.S. Brown and A.C. Hearn. Applications of Symbolic Mathematical Computations. Comput. Phys. Commun., Vol. 17, 207-215, 1979.
- J. Calmet and J.A. van Hulzen. Computer algebra applications. In B. Buchberger, G.E. Collins and R. Loos, editors, Computer Algebra, Symbolic and Algebraic Computation, 245-258, Berlin, 1983. Springer-Verlag.
- A. M. Cohen, editor. The SCAFI Papers, Proceedings of the 1991 SCAFI, Seminar on Studies in Computer Algebra for Industry, CAN, CWI Amsterdam, 1991.
- H.I. Cohen and J.P. Fitch. Uses Made of Computer Algebra in Physics, J. Symbolic Computation, Vol. 11, No. 3, 291-305, 1991.
- J.P. Fitch. The Application of Symbolic Algebra in Physics - a Case of Creeping Flow. In E. W. Ng, editor, EUROSAM'79, 30-41, 1979. Springer-Verlag.
- J.P. Fitch. Applying Computer Algebra. In International Conference on Computer Algebra and its Applications in Theoretical Physics, 262-275, 1985. Dubna.
- E.W. Ng. Symbolic-Numeric Interface: A Review. In E.W. Ng, editor, EUROSAM'79, 330-345, 1979. Springer-Verlag.

Chapter 6

Another sources of study

6.1 Basic references

- B. Buchberger, G.E. Collins, and R. Loos, editors. Computer Algebra, Symbolic and Algebraic Computation. Springer- Verlag, Wien, 1983. first book on computer algebra algorithms

- J.H. Davenport, Y. Siret, and E. Tournier. Computer Algebra, Systems and Algorithms for Algebraic Computation, Academic Press, London, 1988. the best introduction to computer algebra algorithms and systems, introduction to REDUCE

- K.O. Geddes, S.R. Czapor and G. Labahn. Algorithms For Computer Algebra. Kluwer Academic Publishers, Boston, 1992. the best reference book on computer algebra algorithms

- D. Harper, C. Wooff, and D. Hodgkinson. A Guide to Computer Algebra Systems. John Wiley & Sons, Chichester, 1991. comparision of the systems Derive, MACSYMA, Maple, Mathematica, REDUCE

6.2 Other references

- algorithms

  R. Zippel. Effective Polynomial Computation. Kluwer Academic Publishers, Boston, 1993.

  M. Mignotte. Mathematics for Computer Algebra. Springer-Verlag, Berlin, 1992.

  F. Winkler. Polynomial Algorithms in Computer Algebra. Springer-Verlag, Berlin, 1996. ISBN 3-211-82759-5

- design of symbolic computing systems

  A. Miola, M. Temperini (eds.). Advances in the Design of SC Systems. Springer-Verlag, Berlin, 1996.

- Derive

  W. Ellis, and E. Lodi. Derive for the Calculus Student: A Tutorial. Brooks/Cole, Pacific Grove, 1991.

  D.C. Arney. Exploring Calculus with DERIVE. Addison-Wesley, 1992.

- Macsyma

  Richard Pavelle, editor. Applications of Computer Algebra, Kluwer Academic Publishers, 1985.

  Barbara Heller. Macsyma for Statisticians. John Wiley and Sons, 1991.

  Chiang C. Mei. Mathematical Analysis in Engineering. Cambridge University Press, 1994.

- Maple

  A. Heck. Introduction to Maple. Springer-Verlag, Berlin 1993.

  D. Redfern. Maple Handbook. Springer-Verlag, Berlin 1993.

- Mathematica

  R. Maeder. Programming in Mathematica. Addison-Wesley, Redwood City, CA, 1990.

  R.E. Crandall. Mathematica for the Sciences. Addison-Wesley, Redwood City, CA, 1991.

  R. Maeder. The Mathematica Programmer. Academic Press Professional, 1994.

- REDUCE

  F. Brackx. Computer Algebra with LISP and REDUCE. Kluwer Academic Publishers, Boston, 1991. ISBN: 0-7923-1441-7, 300 p.

  M.A.H. MacCallum, and F.J. Wright. Algebraic Computing with REDUCE, Clarendon Press, Oxford, 1991.

  G. Rayna. REDUCE Software for Algebraic Computation. Springer-Verlag, Berlin, 1987.

6.3 Journals

- Journal of Symbolic Computation , Academic Press , monthly, 2 volumes per year, since 1985, basic journal for the theory of computer algebra, editor Bob Caviness, University of Delaware
- SIGSAM Bulletin, bulletin of the special interest group SIGSAM , ACM Press, quarterly, since 1967, editor Robert Corless
- Applicable Algebra in Engineering, Communication and Computing, Springer International, quarterly, since 1990, editor J. Calmet, Karlsruhe
- Macsyma Newsletter, Macsyma Inc.
- MapleTech, The Maple Technical Newsletter, Birkhauser, Boston, biannually, since 1989, editor T. Scott, Waterloo
- Maple Roots Report, The Newsletter from Waterloo Maple Software
- MathUser, The Wolfram Research Newsletter for Mathematica Users, biannually, Wolfram Research, Inc., mathuser@wri.com
- Mathematica Journal, Addison-Wesley, quarterly, since 1990

6.4 Electronic information sources

6.4.1 General electronic information sources

- news group sci.math.symbolic, much on Mathematica and Maple
- information on the special interest group ACM SIGSAM, Association for Computing Machinery, Special Interest Group on Symbolic and Algebraic Manipulation, 1515 Broadway, New York, NY 10036

  information about ACM

  information about SIGSAM

- WWW Computer algebra servers

  SymbolicNet Symbolic Mathematical Computation Information Center, Kent State University, includes mailing list for announcements

  CAIN Europe Computer Algebra Information Network, CAN, Computer Algebra Netherlands

  RISC , Research Institute for Symbolic Computation, Linz

- overview of computer algebra systems and packages, including both commercial and public systems, can be found at CAIN or at the collection of symbolic software at the University of Berkeley

6.4.2 Electronic resourses related to particular systems

Axiom electronic information sources

- WWW site
- bibliography

Derive electronic information sources

- WWW site
- email: swh@aloha.com

Macsyma electronic information sources

- WWW site
- for information about Macsyma
  e-mail:info@macsyma.com
- for service on Macsyma
  e-mail:service@macsyma.com
- ftp site:fermat.macsyma.com for extra applications packages, patches and demos

Maple electronic information sources

- WWW site
- e-mail
  - information and sale:info@maplesoft.on.ca
  - technical support:support@maplesoft.on.ca
- Maple Share Library - library of users programs
  - anonymous ftp://ftp.maplesoft.com
    anonymous ftp://ftp.can.nl/pub/maple-ftplib/
  - e-mail:maple-netlib@can.nl ("send info" in the message body)
  - library contributions: Michael Monagan
- Maple Users Group

  for information on how to subscribe to the mailing list please send email to majordomo@daisy.uwaterloo.ca with the command "info maple-list" in the body of the message.

Mathematica electronic information sources

- WWW site
- e-mail
  - general information, sale
  - general information, sale in Europe
  - customer service
  - user registration
  - technical questions and support , send bugs in Mathematica here
  - technical questions and support in Europe

  - suggestions
  - MathUser Newsletter for users

- Mathsource

  - library of Mathematica packages, notebooks, technical reports, examples, news and information
  - e-mail:mathsource@wri.com (”help intro” in the message body)
  - ftp
  - support

REDUCE electronic information sources

- WWW site at Koln
  WWW site at ZIB Berlin

- REDUCE secretary

- REDUCE Network Library - library of user written REDUCE packages

  - WWW site
  - e-mail US:reduce-netlib@rand.org
  - e-mail Europe:elib@elib.zib-berlin.de (”help” or ”send index” in the message body)

- REDUCE Forum - discussion group

  - contributions
    subscribtion

6.5 References for CAS Comparisons

Asl94 Helmer Aslaksen, *Multiple-valued complex functions and computer algebra*, Research Report No. 631, Department of Mathematics, National University of Singapore, October 1994.

Ber96 Laurent Bernardin, “A Review of Symbolic Solvers”, *SIGSAM Bulletin*, Volume 30, Number 1, March 1996, 9–20.

Fau95 J. C. Faugère, “GB: State of GB + Tutorial, *Proceedings of the POSSO Workshop on Software*, draft version, edited by Jean-Charles Faugère, Joel Marchand and Renaud Rioboo, Paris, March 1–4, 1995, 55–60.

Grä95 Hans-Gert Gräbe, “On Factorized Gröbner Bases”, *Computer Algebra in Science and Engineering*, edited by Fleischer, Grabmeier, Hehl and Küchlin, World Scientific Singapore 1995, 77–89.

Har91 David Harper, Chris Wooff and David Hodgkinson, *A Guide to COMPUTER ALGEBRA SYSTEMS*, John Wiley & Sons, 1991.

Her94 W. Hereman, “Review of Symbolic Software for the Computation of Lie Symmetries of Differential Equations”, *Euromath Bulletin*, Volume 1, Number 2, 1994, 45–79.

Her95 Willy Hereman, “Visual data analysis: maths made easy”, *Physics World*, Volume 8, Number 4, April 1995, 49–53.

Her96 Willy Hereman, “Computer algebra: lightening the load”, *Physics World*, Volume 9, Number 3, March 1996, 47–52.

Pos96 Frank Postel and Paul Zimmermann, “A review of the ODE solvers of Axiom, Derive, Macsyma, Maple, Mathematica, MuPAD and Reduce”, submitted to the 5th Rhine Workshop on Computer Algebra to be held in Saint-Louis, France, April 1–3, 1996.

Rob93 Nicolas Robidoux, “Does Axiom Solve Systems of O.D.E.'s Like Mathematica?”, LA-UR-93-2235, Los Alamos National Laboratory, Los Alamos, New Mexico.

Sim92 Barry Simon, “Comparative CAS Reviews”, *Notices of the American Mathematical Society*, Volume 39, Number 7, September 1992, 700–710.

Sto91 David R. Stoutemyer, “Crimes and Misdemeanors in the Computer Algebra Trade”, *Notices of the American Mathematical Society*, Volume 38, Number 7, September 1991, 778–785.

Wes94 Michael Wester, “A Review of CAS Mathematical Capabilities”, *Computer Algebra Nederland Nieuwsbrief*, Number 13, December 1994, ISSN 1380-1260, 41–48 (newer version of the paper below).

Wes95 Michael Wester, “A Review of CAS Mathematical Capabilities”, *Applied Mechanics in the Americas*, Volume III, edited by Luis A. Godoy, Sergio R. Idelsohn, Patricio A. A. Laura and Dean T. Mook, American Academy of Mechanics and Asociacion Argentina de Mecanica Computacional, Santa Fe, Argentina, 1995, 450–455.

Zim95 Paul Zimmermann, “Wester's test suite in MuPAD 1.2.2”, *Computer Algebra Nederland Nieuwsbrief*, Number 14, April 1995, ISSN 1380-1260, 53–64.

6.6 Computer Algebra Conferences

6.6.1 International

ISSAC

International Society of Symbolic and Algebraic Computation (ISSAC) Conferences

ISSAC'89	1989	Portland, Oregon, USA
ISSAC'90	1990	Tokyo, Japan
ISSAC'91	1991	Bonn, Germany
ISSAC'92	1992	University of California, Berkeley, California, USA
ISSAC'93	July 6–8, 1993	Kiev, Ukraine
ISSAC'94	August 25–27, 1994	London, United Kingdom
ISSAC'95	July 10–12, 1995	Concordia University, Montreal, Canada
ISSAC'96	July 24–26, 1996	ETH, Zürich, Switzerland
ISSAC'97	July 21-23, 1997	Maui, Hawaii, USA

ISSAC is the major yearly international computer algebra conference.

IMACS-ACA

IMACS Conferences on Applications of Computer Algebra (ACA)

ACA'95	May 16–19, 1995	University of New Mexico, Albuquerque, New Mexico, USA
ACA'96	July 17–20, 1996	RISC-Linz, Hagenberg, Austria
ACA'97	July 24-26, 1997	Maui, Hawaii, USA

The primary goal of these conferences is to promote the interaction of *users* of computer algebra, in particular, scientists, engineers and educators.

6.6.2 Systems meetings

Axiom

(Thanks to Grant Keady.)

Jacques Calmet will have details.

International AXIOM Meeting	1996?	Karlsruhe, Germany

Derive

(Thanks to Bernhard Kutzler.)

Derive User Group (DUG) Meetings

1st UK	April 14, 1992	Nottingham, UK
2nd UK	September 20, 1993	Birmingham, UK
1st German	April 24, 1993	Schweinbach, Germany
2nd German	April 14, 1995	Nuernberg, Germany
1st US	November 20, 1994	Orlando, Florida, USA
2nd US	November 19, 1995	Houston, Texas, USA

Derive Conferences

1st Scandinavian DERIVE Conference	October 1–2, 1993	Kungsbacka, Sweden
1st International DERIVE Symposium	April 26–30, 1992	Krems, Austria
2nd International DERIVE Symposium	September 26–30, 1993	Krems, Austria
3rd International DERIVE Symposium	July 30–August 3, 1995	Honolulu, Hawaii, USA
1st International DERIVE Conference	July 11–15, 1994	Plymouth, UK
2nd Int. DERIVE/TI-92 Conference	July 2–6, 1996	Bonn, Germany
DERIVE Days Duesseldorf	April 19–21, 1995	Duesseldorf, Germany
DERIVE Days Leeds	April 13–15, 1996	Leeds, UK

Macsyma

(Thanks to Jeff Golden.)

MACSYMA Users' Conferences (MUCs)

MUC I	July 27–29, 1977	University of California at Berkeley, USA
	Proceedings published as NASA CP-2012	
MUC II	June 20–22, 1979	Washington, D.C., USA
MUC III	July 23–25, 1984	General Electric, Schenectady, New York, USA
Macsyma and PDEase in Undergraduate Education		
	August 13, 1996	University of Washington, Seattle, USA

Maple

(Thanks to Michael Monagan.)

The Maple retreats were held from 1982 to 1994 every year in June at Sparrow Lake in Southern Ontario, Canada. This was a quiet setting where people would relax in a pleasant away from work atmosphere. The meetings were attended by Maple developers, local Maple usrs, and typically 4 invited guests, some of whom gave a presentation. The original purpose of the meetings was to have a small group get together for brain storming sessions which would be uninterrupted over the course of 3 days. Later, as the meetings became larger, and more popular, and people from the Maple company also took part, the Maple retreats became more like a scientific workshop meething with prepared presentations from many speakers, and large group discussions. These were useful for finding out what people were doing and what needed to be done, but they did not provide a good atmosphere where people would work on the design of Maple. The Maple retreat meetings were stopped in 1994. Since then a smaller group of typically 6 to 10 people have met for 2 day and 1 day meetings to focus on design issues.

Mathematica

(Thanks to Michael Trott.)

Mathematica Conferences

1st	January 11–13, 1990	Redwood City, California, USA
2nd	January 12–15, 1991	San Francisco, California, USA
3rd	May 27–31, 1992	Boston, Massachusetts, USA
4th	September 2–4, 1992	Rotterdam, Netherlands
5th	April 21–23, 1994	Champaign, Illinois, USA
1st Australian	July 8–10, 1995	University of Tasmania, Hobart, Australia

Mathematica Developer Conferences

1st	May 6–8, 1993	Champaign, Illinois, USA
2nd	October 6–8, 1995	Champaign, Illinois, USA

6.7 Manuals

- AXIOM

  R.D. Jenks, and R.S. Sutor. AXIOM, the Scientific Computation System. Springer-Verlag, 1992.

  Axiom Release 2.0 Companion Guide, The Numerical Algorithms Group Limited, Oxford, March 1995.

  S.M. Watt, P.A. Broadbery, S.S. Dooley, P. Iglio, S.C. Morrison, J.M. Steinbach, R.S. Sutor, IBM Thomas J. Watson Research Center. AXIOM Library Compiler, User Guide, The Numerical Algorithms Group Limited, Oxford, November 1994.

- Derive

  User Manual, DERIVE, A Mathematical Assistant for Your Personal Computer, Version 2. Soft Warehouse, Inc., Honululu, 1990.

  Albert Rich, Joan Rich, Theresa Shelby and David Stoutemyer. User Manual DERIVE Version 3: A Mathematical Assistant for Your Personal Computer, Soft Warehouse, Inc., September 1994.

- Macsyma

  Macsyma Mathematics and System Reference Manual, 15th edition, Macsyma Inc., 1995

  Macsyma User's Guide: A Tutorial Introduction, Second Edition, Macsyma Inc., 1995.

- Maple

  - new manuals

    K. M. Heal, M. L. Hansen and K. M. Rickard, Maple V Learning Guide, Springer-Verlag, 1996.

    M. B. Monagan, K. O. Geddes, K. M. Heal, G. Labahn and S. M. Vorkoetter, Maple V Programming Guide, Springer-Verlag, 1996.

    Darren Redfern, The Maple Handbook: Maple V Release 4, Springer-Verlag, 1996.

  - older manuals

    B.W. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, and S.M. Watt. First Leaves: A Tutorial Introduction to Maple V. Springer-Verlag, New York, 1992.

    B.W. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, and S.M. Watt: Maple V Library Reference Manual. Springer-Verlag, New York, 1991.

    B.W. Char, K.O. Geddes, G.H. Gonnet, B.L. Leong, M.B. Monagan, and S.M. Watt: Maple V Language Reference Manual. Springer-Verlag, New York, 1991.

- Mathematica

  S. Wolfram. Mathematica, A system for Doing Mathematics by Computer. Addison-Wesley, Redwood City, CA, 1991.

  S. Wolfram. The Mathematica Book, Third Edition, Mathematica Verison 3, Wolfram Media/Cambridge University Press, 1996.

- REDUCE

  A.C. Hearn and J.P. Fitch (ed.), REDUCE User's Manual 3.6, RAND Publication CP78 (Rev. 7/95), Rand, Santa Monica, CA, 1995.

- for other computer algebra systems and packages, including also public systems, consult the overview at CAIN Netherlands or the collection of symbolic software at the University of Berkeley

6.8 Distributors addresses

- AXIOM: NAG Ltd. Wilkinson House, Jordan Hill Road, OXFORD, OX2 8DR, United Kingdom

- Derive: Soft Warehouse, Inc. 3615 Harding Avenue, Suite 505, Honolulu, Hawaii 96816-3735, U.S.A. email: swh@aloha.com

- Macsyma: Macsyma Inc., 20 Academy Street, Arlington, Massachusetts 02174-6436, U.S.A., e-mail:info-macsyma@macsyma.co,

  Scientific Software Service, Attention: Mr. Gregory Kapsias, Niddastrasse 108, D-60329 Frankfurt /a Main 1, Germany, tel: (49) 69-252255, fax: (49) 69-232464, e-mail:100144.347@compuserve.com,

- Maple: Waterloo Maple Software, 450 Phillip Street, Waterloo, Ontario, N2L 5J2, Canada, e-mail:info@maplesoft.on.ca

  Waterloo Maplo Software GmbH, Tiergartenstrasse 17, W-6900, Heidelberg, Germany

  Czech Software First CS 1, Jiri Hrebicek, Brno, tel/fax: 05-74 1248, Czech Republic

- Mathematica: Wolfram Research, Inc. 100 Trade Center Drive, Champaign, IL 61820-7237, U.S.A., e-mail:info@wri.com

  Wolfram Research Europe Ltd., Evenlode Court, Main Road, Long Hanborough, Oxon, OX8 2LA, United Kingdom, e-mail:info-euro@wri.com

  ELKAN s.r.o. V tunich 12, 120 00 Praha 2, tel. 02-23 55 473, Czech Republic

- REDUCE: REDUCE secretary, RAND, 1700 Main Street, P.O. Box 2138, Santa Monica, CA 90407-2138, U.S.A., e-mail:reduce@rand.org

  Herbert Melenk, Symbolik, Konrad Zuse Zentrum fur Informationstechnik, ZIB Berlin, Heibronner Str. 10, D-10711 Berlin-Wilmersdorf, Germany, e-mail:melenk@cs.zib-berlin.de

  Codemist Ltd. "Alta", Horsecombe Vale, Combe Down, Bath BA2 5QR, United Kingdom, e-mail:jpff@maths.bath.ac.uk

  Richard Liska, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University, Brehova 7, 115 19 Praha 1, Czech Republic, e-mail:liska@siduri.fjfi.cvut.cz

- for other computer algebra systems and packages, including also public systems, consult the overview at CAIN Netherlands or the collection of symbolic software at the University of Berkeley