The Wester Problem Set

Timothy Daly

October 22, 2026

Abstract

I'm starting to construct a document based on the Wester problem set (ref Wester, Michael J. (ed) "Computer Algebra Systems" Wiley 1999 ISBN 0-471-98353-5) that shows axiom in it's worst light. I've attached the proto-pamphlet. The idea is to focus on upgrading axiom to enhance the algebra. Eventually it is intended to be a "Can't Do" document of the kinds of algebra axiom still cannot do. A future effort wishlist.

Contents

1 Boolean Logic and Quantifier Elimination

```
-- Simplify logical expressions => false
true and false


   (1)  false
                                                               Type: Boolean
                                        Time: 0.03 (OT) + 0.02 (GC) = 0.05 sec
-- => true
x or (not x)


   Argument number 1 to "or" must be a Boolean.
x : Boolean

                                                                  Type: Void
                                                                  Time: 0 sec
x or (not x)


   x is declared as being in Boolean but has not been given a value.
-- => x or y
y : Boolean

                                                                  Type: Void
                                                                  Time: 0 sec
x or y or (x and y)


   x is declared as being in Boolean but has not been given a value.
-- => x
xor(xor(x, y), y)


   x is declared as being in Boolean but has not been given a value.
-- => [not (w and x)] or (y and z)
w : Boolean

                                                                  Type: Void
                                                                  Time: 0 sec
z : Boolean

                                                                  Type: Void
                                                                  Time: 0 sec
implies(w and x, y and z)
```

```
   w is declared as being in Boolean but has not been given a value.
-- => (x and y) or [not (x or y)]
--x iff y
-- => false
x and 1 > 2


   x is declared as being in Boolean but has not been given a value.
)clear properties w x y z

-- Quantifier elimination: See Richard Liska and Stanly Steinberg, ``Using
-- Computer Algebra to Test Stability'', draft of September 25, 1995, and
-- Hoon Hong, Richard Liska and Stanly Steinberg, ``Testing Stability by
-- Quantifier Elimination'', _Journal of Symbolic Computation_, Volume 24,
-- 1997, 161--187.
-- => (a > 0 and b > 0 and c > 0) or (a < 0 and b < 0 and c < 0)
--    [Hong, Liska and Steinberg, p. 169]
--forAll y in C {implies(a*y**2 + b*y + c = 0, real(y) < 0)}
-- => v > 1   [Liska and Steinberg, p. 24]
--thereExists w in R suchThat _
--{v > 0 and w > 0 and -5*v**2 - 13*v + v*w - w > 0}
-- => a^2 <= 1/2   [Hoon, Liska and Steinberg, p. 174]
--forAll c in R _
--{implies(-1 <= c <= 1, a**2*(-c**4 - 2*c**3 + 2*c + 1) + c**2 + 2*c + 1 <= 4)}
-- => v > 0 and w > |W|   [Liska and Steinberg, p. 22]
--forAll y in C _
--{implies(v > 0 and y**4 + 4*v*w*y**3 + 2*(2*v**2*w**2 + w**2 + W**2)*y**2 _
--   + 4*v*w*(w**2 - W**2) _
--   + (w**2 - W**2)**2 = 0, real(y) < 0)}
-- This quantifier free problem was derived from the above example by QEPCAD
-- => v > 0 and w > |W|   [Liska and Steinberg, p. 22]
v > 0 and 4*w*v > 0 and 4*w*(4*w**2*v**2 + 3*W**2 + w**2) > 0 _
   and 64*w**2*v**2*(w**2 - W**2)*(w**2*v**2 + W**2) > 0 _
   and 64*w**2*v**2*(w**2 - W**2)**3*(w**2*v**2 + W**2) > 0


   (6)  true
                                                            Type: Boolean
              Time: 0.40 (IN) + 0.05 (EV) + 0.08 (OT) + 0.03 (GC) = 0.57 sec
-- => B < 0 and a b > 0   [Liska and Steinberg, p. 49 (equation 86)]
--thereExists y in C, thereExists n in C, thereExists e in R suchThat _
--{real(y) > 0 and real(n) < 0 and y + A*%i*e - B*n = 0 and a*n + b = 0}
-- ---------- Quit ----------
)quit
```

```
real   3.1
user   1.4
sys    0.2
```

2 Set Theory

```
-- ---------- Set Theory ----------
x:= set [a, b, b, c, c, c];


                                              Type: Set OrderedVariableList [a,b,c]
                    Time: 0.03 (IN) + 0.05 (EV) + 0.08 (OT) + 0.07 (GC) = 0.23 sec
y:= set [d, c, b];


                                              Type: Set OrderedVariableList [d,c,b]
                                                           Time: 0.02 (IN) = 0.02 sec
z:= set [b, e, b];


                                                Type: Set OrderedVariableList [b,e]
                                                           Time: 0.02 (EV) = 0.02 sec
-- [x \/ y \/ z, x /\ y /\ z] => [{a, b, c, d, e}, {b}]
[union(union(x, y), z), intersect(intersect(x, y), z)]


   (4)  [{a,b,c,d,e},{b}]
                                                              Type: List Set Symbol
                                             Time: 1.20 (IN) + 0.20 (OT) = 1.40 sec
-- x \/ y \/ z - x /\ y /\ z => {a, c, d, e}
difference(%.1, %.2)


   (5)  {a,c,d,e}
                                                                    Type: Set Symbol
                                             Time: 0.02 (IN) + 0.02 (EV) = 0.03 sec
)clear properties x y z

-- Cartesian product of sets => {(a, c), (a, d), (b, c), (b, d)}
Cartesian(A, B) == _
   set reduce(append, [[[A.i, B.j] for j in 1..#B] for i in 1..#A])
```

```
                                                                  Type: Void
                                                                 Time: 0 sec
Cartesian([a, b], [c, d])

   Compiling function Cartesian with type (List OrderedVariableList [a,
      b],List OrderedVariableList [c,d]) -> Set List Symbol

   (7)  {[a,c],[a,d],[b,c],[b,d]}
                                                       Type: Set List Symbol
                                      Time: 0.12 (IN) + 0.08 (OT) = 0.20 sec
-- ---------- Quit ----------
)quit


real   8.4
user   2.7
sys    0.3
```

3 Numbers

```
-- ---------- Numbers ----------
-- Let's begin by playing with numbers: infinite precision integers
-- => 30414 0932017133 7804361260 8166064768 8443776415 6896051200 0000000000
factorial(50)


   (1)  30414093201713378043612608166064768844377641568960512000000000000
                                                       Type: PositiveInteger
                         Time: 0.03 (EV) + 0.02 (OT) + 0.03 (GC) = 0.08 sec
-- => 2^47 3^22 5^12 7^8 11^4 13^3 17^2 19^2 23^2 29 31 37 41 43 47
factor(%)


         47 22 12 8  4  3  2  2  2
   (2)  2  3  5  7 11 13 17 19 23 29 31 37 41 43 47
                                                      Type: Factored Integer
                                      Time: 0.03 (EV) + 0.05 (OT) = 0.08 sec
-- Double factorial => 10!! = 10*8*6*4*2 = 3840, 9!! = 9*7*5*3*1 = 945
--[10!!, 9!!]
-- ABC base 16 => 2748 base 10
abc
```

```
   (3)  abc
                                                            Type: Variable abc
                                                                   Time: 0 sec
-- 123 base 10 => 234 base 7
radix(123, 7)


   (4)  234
                                                        Type: RadixExpansion 7
                Time: 0.02 (IN) + 0.02 (EV) + 0.03 (OT) + 0.02 (GC) = 0.08 sec
-- 677 base 8 => 1BF base 16
radix(447, 16)


   (5)  1BF
                                                       Type: RadixExpansion 16
                                        Time: 0.02 (EV) + 0.03 (OT) = 0.05 sec
-- [log base 8](32768) => 5
log(32768)/log(8)


        log(32768)
   (6)  ----------
          log(8)
                                                      Type: Expression Integer
                Time: 0.03 (IN) + 0.33 (EV) + 0.12 (OT) + 0.12 (GC) = 0.60 sec
-- 5^(-1) mod 7 => 3;   5^(-1) mod 6 => 5
(5 :: PrimeField 7)**(-1)


   (7)  3
                                                            Type: PrimeField 7
                                        Time: 0.03 (IN) + 0.03 (OT) = 0.07 sec
recip(5 :: PrimeField(7))


   (8)  3
                                                 Type: Union(PrimeField 7,...)
                                                    Time: 0.02 (OT) = 0.02 sec
invmod(5, 7)


   (9)  3
                                                         Type: PositiveInteger
                                                                   Time: 0 sec
(5 :: IntegerMod 6)**(-1)
```

```
   Compiling function G82136 with type Integer -> Boolean
   There are 20 exposed and 17 unexposed library operations named **
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op **
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named **
      with argument type(s)
                                IntegerMod 6
                                   Integer

recip(5 :: IntegerMod 6)


   (10)  5
                                                Type: Union(IntegerMod 6,...)
                                                   Time: 0.02 (EV) = 0.02 sec
invmod(5, 6)


   (11)  5
                                                        Type: PositiveInteger
                                                                  Time: 0 sec
-- Greatest common divisor => 74
reduce(gcd, [1776, 1554, 5698])


   (12)  74
                                                        Type: PositiveInteger
                                                   Time: 0.03 (OT) = 0.03 sec
-- Infinite precision rational numbers => 4861/2520
1/2 + 1/3 + 1/4 + 1/5 + 1/6 + 1/7 + 1/8 + 1/9 + 1/10


         4861
   (13)  ----
         2520
                                                       Type: Fraction Integer
                                                   Time: 0.05 (IN) = 0.05 sec
-- Complete decimal expansion of a rational number => 0.142857 ...
decimal(1/7)
```

```
           ------
   (14)  0.142857
                                                       Type: DecimalExpansion
                                       Time: 0.02 (IN) + 0.02 (EV) = 0.03 sec
-- Multiply two complete decimal expansions and produce an exact result => 2
decimal(7/11) * decimal(22/7)


   (15)  2
                                                       Type: DecimalExpansion
                                                   Time: 0.02 (IN) = 0.02 sec
-- This number should immediately simplify to 3^(1/3)
10/7 * (1 + 29/1000)**(1/3)


         3+-+
   (16)  \|3
                                                        Type: AlgebraicNumber
                           Time: 0.20 (IN) + 0.02 (EV) + 0.02 (OT) = 0.23 sec
-- Simplify an expression with nested square roots => 1 + sqrt(3)
sqrt(2*sqrt(3) + 4)


          +---------+
          |  +-+
   (17)  \|2\|3  + 4
                                                        Type: AlgebraicNumber
                           Time: 0.03 (IN) + 0.02 (EV) + 0.05 (OT) = 0.10 sec
simplify(%)


          +---------+
          |  +-+
   (18)  \|2\|3  + 4
                                                     Type: Expression Integer
                                       Time: 0.10 (EV) + 0.02 (OT) = 0.12 sec
-- Try a more complicated example (from the Putnam exam) => 3 + sqrt(2)
sqrt(14 + 3*sqrt(3 + 2*sqrt(5 - 12*sqrt(3 - 2*sqrt(2)))))


          +---------------------------------------+
          |  +------------------------------+
          |  |  +----------------------+
          |  |  |     +-----------+
          |  |  |     |    +-+
```

```
   (19)  \|3\|2\|- 12\|- 2\|2  + 3  + 5  + 3  + 14
                                                        Type: AlgebraicNumber
                                       Time: 0.05 (IN) + 0.03 (OT) = 0.08 sec
-- See D.J. Jeffrey and A.D. Rich, ``The nesting habits of radicals'', draft of
-- 1998 => sqrt(2) + sqrt(3) + sqrt(5)
sqrt(10 + 2*sqrt(6) + 2*sqrt(10) + 2*sqrt(15))


          +----------------------------+
          |  +--+     +--+     +-+
   (20)  \|2\|15  + 2\|10  + 2\|6  + 10
                                                        Type: AlgebraicNumber
                           Time: 0.05 (IN) + 0.02 (EV) + 0.02 (OT) = 0.08 sec
-- Rationalize the denominator => 5 + 2 sqrt(6)
(sqrt(3) + sqrt(2))/(sqrt(3) - sqrt(2))


           +-+ +-+
   (21)  2\|2 \|3  + 5
                                                        Type: AlgebraicNumber
                           Time: 0.03 (IN) + 0.22 (EV) + 0.02 (OT) = 0.27 sec
-- A factorization of 3 in the integers extended by sqrt(-5)
sqrt(-2 + sqrt(-5)) * sqrt(-2 - sqrt(-5))


          +------------+ +----------+
          |   +---+      | +---+
   (22)  \|- \|- 5  - 2 \|\|- 5  - 2
                                                        Type: AlgebraicNumber
                           Time: 0.02 (IN) + 0.02 (EV) + 0.02 (OT) = 0.05 sec
-- => 3 + sqrt(7)   [Jeffrey and Rich]
(90 + 34*sqrt(7))^(1/3)


          +-----------+
         3|   +-+
   (23)  \|34\|7  + 90
                                                        Type: AlgebraicNumber
                                       Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
-- This is a nontrivial way of writing 12 !
((135 + 78*sqrt(3))**(2/3) + 3)*sqrt(3)/(135 + 78*sqrt(3))**(1/3)


                         +------------+2         +------------+
               +-+      3|   +-+              +-+3|   +-+
   (24)  (- 15\|3  + 26)\|78\|3  + 135   + \|3 \|78\|3  + 135
```

```
                                                         Type: AlgebraicNumber
                           Time: 0.05 (IN) + 0.07 (EV) + 0.02 (OT) = 0.13 sec
% :: Float


   (25)  11.9999999999 99999999998
                                                                   Type: Float
                                        Time: 0.17 (IN) + 0.03 (OT) = 0.20 sec
-- See David Jeffrey, ``Current Problems in Computer Algebra Systems'', talk
-- => 1 + sqrt(2)
(49 + 21*sqrt(2))^(1/5)


          +-----------+
         5|   +-+
   (26)  \|21\|2  + 49
                                                         Type: AlgebraicNumber
                                        Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
-- A nasty example generated by Axiom => [log(sqrt(2) + 1) + sqrt(2)]/3
q:= ((6 - 4*sqrt(2))*log(3 - 2*sqrt(2)) + (3 - 2*sqrt(2))*log(17 - 12*sqrt(2)) _
     + 32 - 24*sqrt(2)) / (48*sqrt(2) - 72)


   (27)
        +-+             +-+             +-+              +-+            +-+
   (- 4\|2  + 6)log(- 2\|2  + 3) + (- 2\|2  + 3)log(- 12\|2  + 17) - 24\|2  + 32
   -----------------------------------------------------------------------------
                                       +-+
                                    48\|2  - 72
                                                       Type: Expression Integer
                           Time: 0.43 (IN) + 0.05 (EV) + 0.07 (OT) = 0.55 sec
q :: Float


   Cannot convert from type Expression Integer to Float for value
        +-+             +-+             +-+              +-+            +-+
   (- 4\|2  + 6)log(- 2\|2  + 3) + (- 2\|2  + 3)log(- 12\|2  + 17) - 24\|2  + 32
   -----------------------------------------------------------------------------
                                       +-+
                                    48\|2  - 72

q :: Complex Float


   (28)  0.7651957164 6421269157
                                                            Type: Complex Float
```

```
                                      Time: 0.25 (IN) + 0.03 (GC) = 0.28 sec
(log(sqrt(2) + 1) + sqrt(2))/3 :: Float


   (29)  0.7651957164 6421269135
                                                      Type: Expression Float
                                      Time: 0.35 (IN) + 0.05 (OT) = 0.40 sec
)clear properties q

-- Cardinal numbers => infinity
2*Aleph(0) - 3


   (30)  Aleph(0)
                                             Type: Union(CardinalNumber,...)
                                                Time: 0.13 (IN) = 0.13 sec
-- 2^aleph_0 => aleph_1
2**Aleph(0)


   >> Error detected within library code:
   Transfinite exponentiation only implemented under GCH

initial (31) ->
real   12.4
user   5.8
sys    0.4
```

4 Numerical Analysis

```
-- ---------- Numerical Analysis ----------
-- This number should immediately simplify to 0.0
0.0/sqrt(2)


   (1)  0.0
                                                      Type: Expression Float
                Time: 0.40 (IN) + 0.20 (EV) + 0.22 (OT) + 0.18 (GC) = 1.0 sec
-- This number normally produces an underflow => 3.29683e-434295
exp(-1000000.0)


   (2)  0.3296831478 088558579 E -434294
                                                                 Type: Float
```

```
                                      Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
-- Arbitrary precision floating point numbers
digits(50);



                                                       Type: PositiveInteger
                                                  Time: 0.02 (IN) = 0.02 sec
-- This number is nearly an integer:
-- 26253741 2640768743.9999999999 9925007259 7198185688 ...
exp(sqrt(163.)*%pi)


   (4)  26253741 2640768743.9999999999 9925007259 7198185688 9
                                                                 Type: Float
                        Time: 0.20 (IN) + 0.03 (OT) + 0.03 (GC) = 0.27 sec
digits(20);



                                                       Type: PositiveInteger
                                                                 Time: 0 sec
-- => [-2, -1]
[floor(-5/3), ceiling(-5/3)]


   (6)  [- 2,- 1]
                                                          Type: List Integer
                                      Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
-- Generate a cubic natural spline s from x = [1, 2, 4, 5] and y = [1, 4, 2, 3]
-- and then compute s(3) => 27/8
[[1, 2, 4, 5], [1, 4, 2, 3]]


   (7)  [[1,2,4,5],[1,4,2,3]]
                                             Type: List List PositiveInteger
                                                  Time: 0.02 (OT) = 0.02 sec
-- Translation
a:= operator('a);



                                                         Type: BasicOperator
                                      Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
p:= sum(a(i)*x**i, i = 1..n)


         n
        --+        i
```

```
   (9)  >     a(i)x
        --+
        i= 1
                                                     Type: Expression Integer
               Time: 0.77 (IN) + 0.68 (EV) + 0.15 (OT) + 0.07 (GC) = 1.67 sec
-- Convert into FORTRAN syntax
outputAsFortran('p = p)


   >> Fortran translation error:
   No corresponding Fortran structure for:

    n
   --+        i
   >     a(i)x
   --+
   i= 1
                                                                   Type: Void
                           Time: 0.27 (IN) + 0.03 (EV) + 0.05 (OT) = 0.35 sec
-- Convert into C syntax
-- Horner's rule---this is important for numerical algorithms
-- => (a[1] + (a[2] + (a[3] + (a[4] + a[5] x) x) x) x) x
p:= sum(a(i)*x**i, i = 1..5)


              5        4        3        2
   (11)  a(5)x  + a(4)x  + a(3)x  + a(2)x  + a(1)x
                                                     Type: Expression Integer
                           Time: 0.07 (IN) + 0.22 (EV) + 0.03 (OT) = 0.32 sec
factor(p)


              5        4        3        2
   (12)  a(5)x  + a(4)x  + a(3)x  + a(2)x  + a(1)x
                                            Type: Factored Expression Integer
                                       Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
p:= p :: MPOLY([x], Expression Integer)


              5        4        3        2
   (13)  a(5)x  + a(4)x  + a(3)x  + a(2)x  + a(1)x
                         Type: MultivariatePolynomial([x],Expression Integer)
                           Time: 0.07 (IN) + 0.03 (OT) + 0.02 (GC) = 0.12 sec
p:= factor(p)

   WARNING (genufact): No known algorithm to factor
```

```
      4   a(4)  3   a(3)  2   a(2)     a(1)
     ?  + ---- ?  + ---- ?  + ---- ? + ----, trying square-free.
          a(5)      a(5)      a(5)     a(5)

                4   a(4)  3   a(3)  2   a(2)     a(1)
   (14)  a(5)x(x  + ---- x  + ---- x  + ---- x + ----)
                    a(5)      a(5)      a(5)     a(5)
                Type: Factored MultivariatePolynomial([x],Expression Integer)
               Time: 0.08 (IN) + 0.32 (EV) + 0.02 (OT) + 0.05 (GC) = 0.47 sec
-- Convert the result into FORTRAN syntax
-- => p = (a(1) + (a(2) + (a(3) + (a(4) + a(5)*x)*x)*x)*x)*x
)set fortran ints2floats off

outputAsFortran('p = p)

      p=a(5)*x*(x**4+(a(4)/a(5))*x**3+(a(3)/a(5))*x*x+(a(2)/a(5))*x+a(1)
     &/a(5))
                                                                   Type: Void
                           Time: 0.48 (IN) + 0.03 (EV) + 0.12 (OT) = 0.63 sec
)clear properties a p

-- Convert the result into C syntax
-- => p = (a[1] + (a[2] + (a[3] + (a[4] + a[5]*x)*x)*x)*x)*x ;
-- Count the number of (floating point) operations needed to compute an
-- expression => {[+, n - 1], [*, (n^2 - n)/2], [f, (n^2 + n)/2]}
f:= operator('f);


                                                          Type: BasicOperator
                                                   Time: 0.03 (IN) = 0.03 sec
sum(product(f(i, k), i = 1..k), k = 1..n)


          n      k
         --+   ++-++
   (17)  >      | |   f(i,i)
         --+    | |
         k= 1  i= 1
                                                     Type: Expression Integer
                           Time: 0.13 (IN) + 0.10 (EV) + 0.05 (OT) = 0.28 sec
-- Interval analysis (interval polynomial example):
-- ([-4, 2] x + [1, 3])^2 => [-8, 16] x^2 + [-24, 12] x + [1, 9]
-- Discretize a PDE: for example, forward differencing time (explicit Euler)
-- and central differencing x on the heat equation =>
-- (f[i, j+1] - f[i, j])/dt = (f[i+1, j] - 2 f[i, j] + f[i-1, j])/dx^2
D(f(x, t), t) = D(f(x, t), x, 2)
```

```
   (18)  f  (x,t)= f    (x,t)
          ,2        ,1,1
                                                Type: Equation Expression Integer
                           Time: 0.08 (IN) + 0.07 (EV) + 0.03 (OT) = 0.18 sec
)clear properties f

-- ---------- Quit ----------
)quit


real   7.1
user   6.2
sys    0.4
```

5 Statistics

```
initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Statistics ----------
-- Compute the mean of a list of numbers => 9
mean(lst) == reduce(+, lst)/#lst

                                                                       Type: Void
                                                                    Time: 0 sec
mean([3, 7, 11, 5, 19])

   Compiling function mean with type List PositiveInteger -> Fraction
      Integer

   (2)  9
                                                          Type: Fraction Integer
                           Time: 0.08 (IN) + 0.02 (OT) + 0.02 (GC) = 0.12 sec
-- Compute the median of a list of numbers => 7
[3, 7, 11, 5, 19]
```

```
   (3)  [3,7,11,5,19]
                                                    Type: List PositiveInteger
                                                    Time: 0.02 (OT) = 0.02 sec
-- Compute the first quartile (25% quantile) of a list of numbers => 2 or 5/2
xx:= [1, 2, 3, 4, 5, 6, 7, 8];


                                                    Type: List PositiveInteger
                                                                 Time: 0 sec
--quartiles(xx, 1)
--quantile(xx, 1/4)
)clear properties xx

-- Compute the mode (the most frequent item) of  a list of numbers => 7
[3, 7, 11, 7, 3, 5, 7]


   (5)  [3,7,11,7,3,5,7]
                                                    Type: List PositiveInteger
                                                    Time: 0.02 (IN) = 0.02 sec
-- Compute the unbiased sample standard deviation of a list of numbers
-- => sqrt(5/2)
[1, 2, 3, 4, 5]


   (6)  [1,2,3,4,5]
                                                    Type: List PositiveInteger
                                                                 Time: 0 sec
-- Discrete distributions---what is the probability of finding exactly 12
-- switches that work from a group of 15 where the probability of a single one
-- working is 75%?  (Need to use the probability density function [PDF] of the
-- binomial distribution.) => 0.22520
--PDF(BinomialDistribution(15, .75), 12)
-- Replace ‘exactly’ by ‘up through’ in the above.  (Need to use the cumulative
-- probability density function [CDF] of the binomial distribution.) => 0.76391
--CDF(BinomialDistribution(15, .75), 12)
-- Continuous distributions---if a radiation emission can be modeled by a
-- normal distribution with a mean of 4.35 mrem and a standard deviation of
-- 0.59 mrem, what is the probability of being exposed to anywhere from 4 to 5
-- mrem? => .5867
--CDF(Normal(4.35, 0.59), 5) - CDF(Normal(4.35, 0.59), 4)
-- Hypothesis testing---how good of a guess is 5 for the mean of xx? */
xx:= [1, -2, 3, -4, 5, -6, 7, -8, 9, 10];


                                                            Type: List Integer
```

```
                                                                    Time: 0 sec
-- Using Student's T distribution (preferred) => 0.057567
--students_t_distrib((sample_mean(xx) - 5)/(sample_standard_deviation(xx) /
--                                          sqrt(length(xx))), length(xx) - 1)
--% :: Float
-- Using the normal distribution (as an alternative) => 0.040583
--standard_normal_distrib((sample_mean(xx) - 5)/(sample_standard_deviation(xx) /
--                                               sqrt(length(xx))))
--% :: Float
)clear properties xx

-- Chi-square test---what is the expectation that row characteristics are
-- independent of column characteristics for a two dimensional array of data?
-- => 0.469859   (chi2 = 1153/252)
x:= matrix([[41, 27, 22], [79, 53, 78]]);


                                                         Type: Matrix Integer
                         Time: 0.02 (IN) + 0.02 (EV) + 0.02 (OT) = 0.05 sec
m:= nrows(x);


                                                         Type: PositiveInteger
                                                                    Time: 0 sec
n:= ncols(x);


                                                         Type: PositiveInteger
                                                                    Time: 0 sec
rowSum:= [reduce(+, row(x, i)) for i in 1..m];


                                                            Type: List Integer
                                                    Time: 0.03 (IN) = 0.03 sec
colSum:= [reduce(+, column(x, j)) for j in 1..n];


                                                            Type: List Integer
                                                    Time: 0.02 (IN) = 0.02 sec
matSum:= reduce(+, rowSum);


                                                         Type: PositiveInteger
                                                    Time: 0.02 (IN) = 0.02 sec
e : ARRAY2 Fraction Integer := new(m, n, 0);
```

```
                                   Type: TwoDimensionalArray Fraction Integer
                                                   Time: 0.02 (IN) = 0.02 sec
for i in 1..m repeat for j in 1..n repeat e(i, j):= rowSum(i)*colSum(j)/matSum;

                                                                   Type: Void
                                       Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
for i in 1..m repeat for j in 1..n repeat _
   e(i, j):= (x(i, j) - e(i, j))**2/e(i, j);

                                                                   Type: Void
               Time: 0.03 (IN) + 0.02 (EV) + 0.03 (OT) + 0.02 (GC) = 0.10 sec
-- chi2:= sum(sum((x(i, j) - e(i, j))**2/e(i, j), i = 1..m), j = 1..n)
chi2:= reduce(+, [reduce(+, row(e, i)) for i in 1..m])


         1153
   (17)  ----
          252
                                                       Type: Fraction Integer
                                                   Time: 0.03 (IN) = 0.03 sec
--ChiSquarePValue(chi2, m*n - 1)
-- or
--1 - CDF(ChiSquareDistribution(m*n - 1), chi2)
)clear properties chi2 colSum matSum rowSum e m n x

-- Linear regression (age as a function of developmental score).  See Lambert
-- H. Koopmans, _Introduction to Contemporary Statistical Methods_, Second
-- Edition, Duxbury Press, 1987, p. 459 => y' = 0.7365 x + 6.964
t:= [[3.33, 3.25, 3.92, 3.50,  4.33,  4.92,  6.08,  7.42,  8.33,  8.00,  9.25, _
       10.75], _
     [8.61, 9.40, 9.86, 9.91, 10.53, 10.61, 10.59, 13.28, 12.76, 13.44, 14.27, _
       14.13]];


                                                        Type: List List Float
                           Time: 0.10 (IN) + 0.03 (OT) + 0.02 (GC) = 0.15 sec
--fit(transpose(%), [1, x], x)
)clear properties t

-- Multiple linear regression (income as a function of age and years of
-- college) => y = -16278.7 + 960.925 x1 + 2975.66 x2
[[37, 45, 38, 42, 31], [4, 0, 5, 2, 4], [31200, 26800, 35000, 30300, 25400]];


                                           Type: List List NonNegativeInteger
```

```
                                  Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
--fit(transpose(%), [1, x1, x2], [x1, x2])
-- Multiple linear regression using the L1 or Least Absolute Deviations
-- technique rather than the Least Squares technique (minimizing the sum of the
-- absolute values of the residuals rather than the sum of the squares of the
-- residuals).  Here, the Stack-loss Data is used (percentage of ammonia lost
-- times 10 from the operation of a plant over 21 days as a function of air
-- flow to the plant, cooling water inlet temperature and acid concentration).
-- See W. N. Venables and B. D. Ripley, _Modern Applied Statistics with
-- S-plus_, Springer, 1994, p. 218.
-- => y = 0.83188 x1 + 0.57391 x2 - 0.06086 x3 - 39.68984
[[80, 80, 75, 62, 62, 62, 62, 62, 58, 58, 58, 58, 58, 58, 50, 50, 50, 50, 50, _
  56, 70], _
 [27, 27, 25, 24, 22, 23, 24, 24, 23, 18, 18, 17, 18, 19, 18, 18, 19, 19, 20, _
  20, 20], _
 [89, 88, 90, 87, 87, 87, 93, 93, 87, 80, 89, 88, 82, 93, 89, 86, 72, 79, 80, _
  82, 91], _
 [42, 37, 37, 28, 18, 18, 19, 20, 15, 14, 14, 13, 11, 12,  8,  7,  8,  8,  9, _
  15, 15]];


                                             Type: List List PositiveInteger
                                                Time: 0.02 (IN) = 0.02 sec
--fit(transpose(%), [1, x1, x2, x3], [x1, x2, x3])
-- Nonlinear regression (Weight Loss Data from an Obese Patient consisting of
-- the time in days and the weight in kilograms of a patient undergoing a
-- weight rehabilitation program).  Fit this using least squares to weight =
-- b0 + b1 2^(- days/th), starting at (b0, b1, th) = (90, 95, 120)   [Venables
-- and Ripley, p. 225] => weight = 81.37375 + 102.6842 2^(- days/141.9105)
[[  0,   4,   7,   7,  11,  18,  24,  30,  32,  43,  46,  60,  64,  70,  71, _
    71,  73,  74,  84,  88,  95, 102, 106, 109, 115, 122, 133, 137, 140, 143, _
   147, 148, 149, 150, 153, 156, 161, 164, 165, 165, 170, 176, 179, 198, 214, _
   218, 221, 225, 233, 238, 241, 246], _
 [184.35, 182.51, 180.45, 179.91, 177.91, 175.81, 173.11, 170.06, 169.31, _
  165.10, 163.11, 158.30, 155.80, 154.31, 153.86, 154.20, 152.20, 152.80, _
  150.30, 147.80, 146.10, 145.60, 142.50, 142.30, 139.40, 137.90, 133.70, _
  133.70, 133.30, 131.20, 133.00, 132.20, 130.80, 131.30, 129.00, 127.90, _
  126.90, 127.70, 129.50, 128.40, 125.40, 124.90, 124.90, 118.20, 118.20, _
  115.30, 115.70, 116.00, 115.50, 112.60, 114.00, 112.60]];


                                                    Type: List List Float
                                  Time: 0.30 (IN) + 0.03 (OT) = 0.33 sec
-- ---------- Quit ----------
)quit
```

```
real   15.0
user   2.4
sys    0.3
```

6 Combinatorial Theory

```
-- ---------- Combinatorial Theory ----------
-- Pochhammer symbol (a)_n = a (a + 1) ... (a + n - 1) => a (a + 1) (a + 2)
--pochhammer(a, 3)
-- Binomial coefficient => n (n - 1) (n - 2)/6
binomial(n, 3)


        n
   (1)  ( )
        3
                                                       Type: Expression Integer
               Time: 0.27 (IN) + 0.12 (EV) + 0.20 (OT) + 0.12 (GC) = 0.70 sec
factorFraction(normalize(%) :: Fraction Polynomial Integer)


        (n - 2)(n - 1)n
   (2)  ---------------
               6
                                     Type: Fraction Factored Polynomial Integer
               Time: 0.12 (IN) + 0.38 (EV) + 0.25 (OT) + 0.07 (GC) = 0.82 sec
-- 2^n n! (2 n - 1)!! => (2 n)!
--2**n * factorial(n) * (2*n - 1)!!
-- 2^n n! product(2 k - 1, k = 1..n) => (2 n)!
2**n * factorial(n) * product(2*k - 1, k = 1..n)


             n
           n++-++
   (3)  n!2  | |   2k - 1
             | |
            k= 1
                                                       Type: Expression Integer
                           Time: 0.30 (IN) + 0.03 (EV) + 0.07 (OT) = 0.40 sec
-- => (2 n)!/[2^(2 n) (n!)^2]   or   (2 n - 1)!!/[2^n n!]
Gamma(n + 1/2)/(sqrt(%pi) * factorial(n))
```

```
         _ 2n + 1
        | (------)
              2
   (4)  ----------
            +---+
         n!\|%pi
                                                     Type: Expression Integer
               Time: 0.85 (IN) + 0.02 (EV) + 0.07 (OT) + 0.02 (GC) = 0.95 sec
-- Partitions of an integer => {1+1+1+1, 1+1+2, 1+3, 2+2, 4} (5 in all)
4


   (5)  4
                                                        Type: PositiveInteger
                                                                  Time: 0 sec
-- Stirling numbers of the first kind: S_1(5, 2) => -50
stirling1(5, 2)


   (6)  - 50
                                                                Type: Integer
                                                   Time: 0.02 (EV) = 0.02 sec
-- Euler's totient function => 576
eulerPhi(1776)


   (7)  576
                                                        Type: PositiveInteger
                                                   Time: 0.02 (IN) = 0.02 sec
-- ---------- Quit ----------
)quit


real   9.9
user   3.6
sys    0.3
```

7 Number Theory

```
-- ---------- Number Theory ----------
-- Display the largest 6-digit prime and the smallest 7-digit prime
-- => [999983, 1000003]
[prevPrime(1000000), nextPrime(1000000)]
```

```
   (1)  [999983,1000003]
                                                   Type: List PositiveInteger
                           Time: 0.02 (IN) + 0.08 (OT) + 0.02 (GC) = 0.12 sec
-- Primitive root => 19
191


   (2)  191
                                                        Type: PositiveInteger
                                                   Time: 0.02 (OT) = 0.02 sec
-- (a + b)^p mod p => a^p + b^p for p prime and a, b in Z_p   [Chris Hurlburt]
-- See Thomas W. Hungerford, _Algebra_, Springer-Verlag, 1974, p. 121 for a
-- more general simplification: (a +- b)^(p^n) => a^(p^n) +- b^(p^n)
(a + b)**p :: PrimeField(p)


   Cannot convert the first argument of PrimeField p to the type
      PositiveInteger.
-- Congruence equations.  See Harold M. Stark, _An Introduction to Number
-- Theory_, The MIT press, 1984.
-- 9 x = 15 mod 21 => x = 4 mod 7   or   {4, 11, 18} mod 21   [Stark, p. 68]
solve(9*x = 15 :: IntegerMod(21), x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                      Equation Polynomial IntegerMod 21
                                 Variable x

-- 7 x = 22 mod 39 => x = 5 mod 13   or   31 mod 39   [Stark, p. 69]
solve(7*x = 22 :: IntegerMod(39), x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
```

```
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                    Equation Polynomial IntegerMod 39
                              Variable x

-- x^2 + x + 4 = 0 mod 8 => x = {3, 4} mod 8   [Stark, p. 97]
solve(x**2 + x + 4 = 0 :: IntegerMod(8), x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                    Equation Polynomial IntegerMod 8
                              Variable x

-- x^3 + 2 x^2 + 5 x + 6 = 0 mod 11 => x = 3 mod 11   [Stark, p. 97]
solve(x**3 + 2*x**2 + 5*x + 6 = 0 :: PrimeField(11), x)


   (3)  [x= 3]
                      Type: List Equation Fraction Polynomial PrimeField 11
                         Time: 0.72 (IN) + 0.18 (EV) + 0.18 (OT) = 1.08 sec
-- {x = 7 mod 9, x = 13 mod 23, x = 1 mod 2} => x = 151 mod 414   [Stark,
-- p. 76]
chineseRemainder([7, 13, 1], [9, 23, 2])


   (4)  151
                                                       Type: PositiveInteger
               Time: 0.02 (IN) + 0.02 (EV) + 0.03 (OT) + 0.03 (GC) = 0.10 sec
-- {5 x + 4 y = 6 mod 7, 3 x - 2 y = 6 mod 7} => x = 1 mod 7, y = 2 mod 7
-- [Stark, p. 76]
solve([5*x + 4*y = 6 :: PrimeField(7), 3*x - 2*y = 6 :: PrimeField(7)], [x, y])


   (5)  [[x= 1,y= 2]]
                  Type: List List Equation Fraction Polynomial PrimeField 7
               Time: 1.33 (IN) + 0.08 (EV) + 0.27 (OT) + 0.02 (GC) = 1.70 sec
```

```
-- 2 x + 3 y = 1 mod 5 =>
-- (x, y) = {(0, 2), (1, 3), (2, 4), (3, 0), (4, 1)} mod 5
solve(2*x + 3*y = 1 :: PrimeField(5), [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                      Equation Polynomial PrimeField 5
                       List OrderedVariableList [x,y]

-- 2 x + 3 y = 1 mod 6 =>   [Stark, p. 76]
-- (x, y) = {(2, 1), (2, 3), (2, 5), (5, 1), (5, 3), (5, 5)} mod 6
solve(2*x + 3*y = 1 :: IntegerMod(6), [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                      Equation Polynomial IntegerMod 6
                       List OrderedVariableList [x,y]

-- Diophantine equations => x = 2, y = 5 (Wallis)   [Stark, p. 147]
solve(x**4 + 9 = y**2, [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
```

```
      with argument type(s)
                         Equation Polynomial Integer
                       List OrderedVariableList [x,y]

-- => x = 11, y = 5 (Fermat)   [Stark, p. 147]
solve(x**2 + 4 = y**3, [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                         Equation Polynomial Integer
                       List OrderedVariableList [x,y]

-- => (x, y, t, z, w) = (3, 4, 5, 12, 13), (7, 24, 25, 312, 313), ...
--    [Stark, p. 154]
system:= [x**2 + y**2 = t**2, t**2 + z**2 = w**2]


          2    2   2  2    2   2
   (6)  [y  + x = t ,z  + t = w ]
                                       Type: List Equation Polynomial Integer
                           Time: 0.02 (IN) + 0.02 (EV) + 0.03 (OT) = 0.07 sec
solve(system, [x, y, t, z, w])


   >> Error detected within library code:
   system does not have a finite number of solutions

initial (7) ->
real   58.4
user   14.4
sys    0.5

-------------------------------------------------------------------------------

Fri Jun 13 01:16:45 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
```

```
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Number Theory ----------
-- Rational approximation of sqrt(3) with an error tolerance of 1/500 => 26/15
rationalApproximation(sqrt(3.), 3)


        26
   (1)  --
        15
                                                       Type: Fraction Integer
               Time: 0.02 (IN) + 0.05 (EV) + 0.07 (OT) + 0.07 (GC) = 0.20 sec
-- Continued fractions => 3 + 1/(7 + 1/(15 + 1/(1 + 1/(292 + ...
continuedFraction(3.1415926535)


   (2)
           1 |     1  |     1 |      1  |     1 |     1 |     1 |     1 |
     3 + +---+ + +----+ + +---+ + +-----+ + +---+ + +---+ + +---+ + +---+
         | 7     | 15     | 1     | 292     | 1     | 1     | 6     | 2
   +
       1  |     1 |
     +----+ + +---+ + ...
     | 13     | 3
                                              Type: ContinuedFraction Integer
                           Time: 0.03 (EV) + 0.05 (OT) + 0.02 (GC) = 0.10 sec
-- => 4 + 1/(1 + 1/(3 + 1/(1 + 1/(8 + 1/(1 + 1/(3 + 1/(1 + 1/(8 + ...
--    [Stark, p. 340]
continuedFraction(sqrt(23))

   There are 2 exposed and 3 unexposed library operations named
```

```
      continuedFraction having 1 argument(s) but none was determined to
      be applicable. Use HyperDoc Browse, or issue
                        )display op continuedFraction
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      continuedFraction with argument type(s)
                               AlgebraicNumber

continuedFraction(sqrt(23) :: Float)


   (3)
           1 |     1 |     1 |     1 |     1 |     1 |     1 |     1 |     1 |
     4 + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+
         | 1     | 3     | 1     | 8     | 1     | 3     | 1     | 8     | 1
   +
       1 |
     +---+ + ...
     | 3
                                              Type: ContinuedFraction Integer
               Time: 0.13 (IN) + 0.03 (EV) + 0.03 (OT) + 0.03 (GC) = 0.23 sec
-- => 1 + 1/(1 + 1/(1 + 1/(1 + ...   See Oskar Perron, _Die Lehre von den
--    Kettenbr\"uchen_, Chelsea Publishing Company, 1950, p. 52.
continuedFraction((1 + sqrt(5))/2 :: Float)


   (4)
           1 |     1 |     1 |     1 |     1 |     1 |     1 |     1 |     1 |
     1 + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+ + +---+
         | 1     | 1     | 1     | 1     | 1     | 1     | 1     | 1     | 1
   +
       1 |
     +---+ + ...
     | 1
                                              Type: ContinuedFraction Integer
                           Time: 0.35 (IN) + 0.02 (EV) + 0.05 (OT) = 0.42 sec
-- => 1/(2 x + 1/(6 x + 1/(10 x + 1/(14 x + ...   [Perron, p. 353]
continuedFraction((exp(1/x) - 1)/(exp(1/x) + 1))

   There are 2 exposed and 3 unexposed library operations named
      continuedFraction having 1 argument(s) but none was determined to
      be applicable. Use HyperDoc Browse, or issue
                        )display op continuedFraction
```

```
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      continuedFraction with argument type(s)
                             Expression Integer

-- => 1/(2 x + 1/(2 x + 1/(2 x + ...   (Re x > 0)   From Liyang Xu, ``Method
--    Derived from Continued Fraction Approximations'', draft.
continuedFraction(sqrt(x**2 + 1) - x)

   There are 2 exposed and 3 unexposed library operations named
      continuedFraction having 1 argument(s) but none was determined to
      be applicable. Use HyperDoc Browse, or issue
                       )display op continuedFraction
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      continuedFraction with argument type(s)
                             Expression Integer

-- ---------- Quit ----------
)quit


real   9.3
user   3.0
sys    0.4
```

8 Algebra

```
-- ---------- Algebra ----------
-- One would think that the simplification 2 2^n => 2^(n + 1) would happen
-- automatically or at least easily ...
2*2**n


          n
   (1)  2 2
                                                   Type: Expression Integer
```

```
               Time: 0.18 (IN) + 0.10 (EV) + 0.17 (OT) + 0.12 (GC) = 0.57 sec
-- And how about 4 2^n => 2^(n + 2)?   [Richard Fateman]
4*2**n



           n
   (2)  4 2
                                                     Type: Expression Integer
                                                   Time: 0.05 (IN) = 0.05 sec
-- (-1)^[n (n + 1)] => 1 for integer n
(-1)**(n*(n + 1))



              2
             n  + n
   (3)  (- 1)
                                                     Type: Expression Integer
                                       Time: 0.07 (IN) + 0.02 (EV) = 0.08 sec
-- Also easy => 2 (3 x - 5)
factor(6*x - 10)



   (4)  2(3x - 5)
                                            Type: Factored Polynomial Integer
               Time: 0.08 (IN) + 0.15 (EV) + 0.03 (OT) + 0.05 (GC) = 0.32 sec
-- Univariate gcd: gcd(p1, p2) => 1, gcd(p1 q, p2 q) => q   [Richard Liska]
p1:= 64*x**34 - 21*x**47 - 126*x**8 - 46*x**5 - 16*x**60 - 81;




                                                     Type: Polynomial Integer
                                       Time: 0.02 (IN) + 0.05 (OT) = 0.07 sec
p2:= 72*x**60 - 25*x**25 - 19*x**23 - 22*x**39 - 83*x**52 + 54*x**10 + 81;



                                                     Type: Polynomial Integer
                                                   Time: 0.07 (IN) = 0.07 sec
q:= 34*x**19 - 25*x**16 + 70*x**7 + 20*x**3 - 91*x - 86;



                                                     Type: Polynomial Integer
                                       Time: 0.02 (IN) + 0.05 (OT) = 0.07 sec
gcd(p1, p2)



   (8)  1
                                                     Type: Polynomial Integer
```

```
                                                   Time: 0.07 (EV) = 0.07 sec
gcd(expand(p1*q), expand(p2*q)) - q


   (9)  0
                                                     Type: Polynomial Integer
             Time: 31.85 (IN) + 0.08 (EV) + 0.02 (OT) + 4.53 (GC) = 36.48 sec
-- resultant(p1 q, p2 q) => 0
resultant(expand(p1*q), expand(p2*q), x)


   (10)  0
                                                     Type: Polynomial Integer
                         Time: 31.66 (IN) + 1.42 (EV) + 4.85 (GC) = 37.93 sec
-- How about factorization? => p1 * p2
factor(expand(p1 * p2))


   (11)
   -
             120        112        107        99        94        86       85
        1152x    - 1328x    + 1512x    - 2095x   - 4608x   + 4850x   - 400x
      +
              83        73       72       70        68        65        60
        - 304x   + 1408x   - 525x   + 465x   + 9072x   + 3312x   - 3330x
      +
             59        57        52        47        44        39        34
        1600x   - 1468x   - 6723x   - 1071x   - 4468x   - 1782x   - 5184x
      +
               33        31        30       28        25        23        18
        - 3150x   - 2394x   - 1150x   - 874x   - 2025x   - 1539x   + 6804x
      +
             15        10         8        5
        2484x   + 4374x   + 10206x  + 3726x  + 6561
                                            Type: Factored Polynomial Integer
        Time: 182.21 (IN) + 184.93 (EV) + 0.07 (OT) + 60.45 (GC) = 427.65 sec
)clear properties p1 p2 q

-- Multivariate gcd: gcd(p1, p2) => 1, gcd(p1 q, p2 q) => q
p1:= 24*x*y**19*z**8 - 47*x**17*y**5*z**8 + 6*x**15*y**9*z**2 - 3*x**22 + 5;


                                                     Type: Polynomial Integer
                                       Time: 0.08 (IN) + 0.02 (OT) = 0.10 sec
p2:= 34*x**5*y**8*z**13 + 20*x**7*y**7*z**7 + 12*x**9*y**16*z**4 + 80*y**14*z;
```

```
                                                     Type: Polynomial Integer
                                       Time: 0.08 (IN) + 0.03 (OT) = 0.12 sec
q:= 11*x**12*y**7*z**13 - 23*x**2*y**8*z**10 + 47*x**17*y**5*z**8;


                                                     Type: Polynomial Integer
                                       Time: 0.05 (IN) + 0.03 (OT) = 0.08 sec
gcd(p1, p2)


   (15)  1
                                                     Type: Polynomial Integer
                                                   Time: 0.02 (EV) = 0.02 sec
gcd(expand(p1*q), expand(p2*q)) - q


   (16)  0
                                                     Type: Polynomial Integer
                          Time: 6.62 (IN) + 8.73 (EV) + 1.32 (GC) = 16.67 sec
-- How about factorization? => p1 * p2
factor(expand(p1 * p2))


   (17)
       7        19      17 5  8     15 9 2     22
     2y z((24x y   - 47x  y )z  + 6x  y z  - 3x   + 5)
  *
         5   12      7 6     9 9 3      7
     (17x y z   + 10x z  + 6x y z  + 40y )
                                            Type: Factored Polynomial Integer
             Time: 9.30 (IN) + 10.72 (EV) + 0.02 (OT) + 1.53 (GC) = 21.57 sec
)clear properties p1 p2 q

-- => x^n for n > 0   [Chris Hurlburt]
gcd(2*x**(n + 4) - x**(n + 2), 4*x**(n + 1) + 3*x**n)


   (18)  1
                                                     Type: Expression Integer
                                       Time: 0.07 (IN) + 0.05 (OT) = 0.12 sec
-- Resultants.  If the resultant of two polynomials is zero, this implies they
-- have a common factor.  See Keith O. Geddes, Stephen R. Czapor and George
-- Labahn, _Algorithms for Computer Algebra_, Kluwer Academic Publishers, 1992,
-- p. 286 => 0
resultant(3*x**4 + 3*x**3 + x**2 - x - 2, x**3 - 3*x**2 + x + 5, x)
```

```
   (19)  0
                                                     Type: Polynomial Integer
                                                   Time: 0.03 (IN) = 0.03 sec
-- Numbers are nice, but symbols allow for variability---try some high school
-- algebra: rational simplification => (x - 2)/(x + 2)
(x**2 - 4)/(x**2 + 4*x + 4)


         x - 2
   (20)  -----
         x + 2
                                            Type: Fraction Polynomial Integer
                                       Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
-- This example requires more sophistication => e^(x/2) - 1
[(%e**x - 1)/(%e**(x/2) + 1), (exp(x) - 1)/(exp(x/2) + 1)]


            x       x
          %e  - 1 %e  - 1
   (21)  [-------,-------]
            x       x
            -       -
            2       2
          %e  + 1 %e  + 1
                                                Type: List Expression Integer
               Time: 1.02 (IN) + 0.20 (EV) + 0.15 (OT) + 0.35 (GC) = 1.72 sec
map(normalize, %)


            x       x
            -       -
            2       2
   (22)  [%e  - 1,%e  - 1]
                                                Type: List Expression Integer
                           Time: 0.05 (IN) + 0.45 (EV) + 0.02 (OT) = 0.52 sec
-- Expand and factor polynomials
(x + 1)**20


   (23)
      20      19       18        17        16         15         14         13
     x   + 20x   + 190x   + 1140x   + 4845x   + 15504x   + 38760x   + 77520x
   +
            12          11          10          9          8         7         6
```

```
     125970x   + 167960x   + 184756x   + 167960x  + 125970x  + 77520x  + 38760x
   +
           5        4        3       2
     15504x  + 4845x  + 1140x  + 190x  + 20x + 1
                                                     Type: Polynomial Integer
                                       Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
D(%, x)


   (24)
        19         18          17           16          15           14           13
     20x   + 380x   + 3420x   + 19380x   + 77520x   + 232560x   + 542640x
   +
             12           11           10          9           8           7
     1007760x   + 1511640x   + 1847560x   + 1847560x  + 1511640x  + 1007760x
   +
            6          5         4         3        2
     542640x  + 232560x  + 77520x  + 19380x  + 3420x  + 380x + 20
                                                     Type: Polynomial Integer
                                       Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
factor(%)


                  19
   (25)  20(x + 1)
                                            Type: Factored Polynomial Integer
                                                   Time: 0.02 (EV) = 0.02 sec
-- Completely factor this polynomial, then try to multiply it back together!
radicalSolve(x**3 + x**2 - 7 = 0, x)


   (26)
                       +--------------+2                  +--------------+
                       |   +----+                         |   +----+
                       |   |1295                          |   |1295
                       |9  |---- + 187                    |9  |---- + 187
           +---+       |  \|  3               +---+       |  \|  3
       (- \|- 3  + 1) 3|--------------  + (- \|- 3  - 1) 3|-------------- - 2
                      \|       2                         \|       2
   [x= ----------------------------------------------------------------------,
                                          +--------------+
                                          |   +----+
                                          |   |1295
                                          |9  |---- + 187
                              +---+       |  \|  3
                           (3\|- 3  + 3) 3|--------------
```

```
                                         \|       2
                       +--------------+2                  +--------------+
                       |   +----+                         |   +----+
                       |   |1295                          |   |1295
                       |9  |---- + 187                    |9  |---- + 187
           +---+       |  \|  3               +---+       |  \|  3
       (- \|- 3  - 1) 3|--------------  + (- \|- 3  + 1) 3|-------------- + 2
                      \|       2                         \|       2
    x= ----------------------------------------------------------------------,
                                          +--------------+
                                          |   +----+
                                          |   |1295
                                          |9  |---- + 187
                              +---+       |  \|  3
                           (3\|- 3  - 3) 3|--------------
                                         \|       2
        +--------------+2    +--------------+
        |   +----+           |   +----+
        |   |1295            |   |1295
        |9  |---- + 187      |9  |---- + 187
        |  \|  3             |  \|  3
       3|--------------   - 3|--------------  + 1
       \|       2           \|       2
    x= ------------------------------------------]
                      +--------------+
                      |   +----+
                      |   |1295
                      |9  |---- + 187
                      |  \|  3
                   3 3|--------------
                     \|       2
                                       Type: List Equation Expression Integer
                           Time: 0.13 (IN) + 0.12 (EV) + 0.13 (OT) = 0.38 sec
reduce(*, map(e +-> lhs(e) - rhs(e), %))


          3    2
   (27)  x  + x  - 7
                                                     Type: Expression Integer
                                       Time: 0.03 (IN) + 0.07 (EV) = 0.10 sec
x**100 - 1


          100
   (28)  x      - 1
                                                     Type: Polynomial Integer
```

```
                                                   Time: 0.02 (OT) = 0.02 sec
factor(%)


   (29)
                     2       4    3    2           4    3    2
     (x - 1)(x + 1)(x  + 1)(x  - x  + x  - x + 1)(x  + x  + x  + x + 1)
  *
       8    6    4    2       20    15    10    5       20    15    10    5
     (x  - x  + x  - x  + 1)(x   - x   + x   - x  + 1)(x   + x   + x   + x  + 1)
  *
       40    30    20    10
     (x   - x   + x   - x   + 1)
                                            Type: Factored Polynomial Integer
                                       Time: 0.05 (EV) + 0.03 (OT) = 0.08 sec
-- Factorization over the complex rationals
-- => (2 x + 3 i) (2 x - 3 i) (x + 1 + 4 i) (x + 1 - 4 i)
factor(4*x**4 + 8*x**3 + 77*x**2 + 18*x + 153, [rootOf(i**2 + 1)])


                           3i      3i
   (30)  4(x - 4i + 1)(x - --)(x + --)(x + 4i + 1)
                            2       2
                                    Type: Factored Polynomial AlgebraicNumber
                           Time: 0.17 (IN) + 1.65 (EV) + 0.03 (OT) = 1.85 sec
-- Algebraic extensions
sqrt2:= rootOf(sqrt2**2 - 2);



                                                        Type: AlgebraicNumber
                                                   Time: 0.02 (OT) = 0.02 sec
-- => sqrt2 + 1
1/(sqrt2 - 1)


   (32)  sqrt2 + 1
                                                        Type: AlgebraicNumber
                           Time: 0.02 (IN) + 0.03 (EV) + 0.02 (OT) = 0.07 sec
-- => (x^2 - 2 x - 3)/(x - sqrt2) = (x + 1) (x - 3)/(x - sqrt2)
--    [Richard Liska]
(x**3 + (sqrt2 - 2)*x**2 - (2*sqrt2 + 3)*x - 3*sqrt2)/(x**2 - 2)


          2
         x  - 2x - 3
   (33)  -----------
```

```
          x - sqrt2
                                            Type: Fraction Polynomial AlgebraicNumber
                                  Time: 0.40 (IN) + 0.20 (EV) + 0.12 (OT) = 0.72 sec
numer(%)/ratDenom(denom(%))


            2
         - x  + 2x + 3
   (34)  -------------
           sqrt2 - x
                                                              Type: Expression Integer
                                            Time: 0.27 (IN) + 0.08 (OT) = 0.35 sec
)clear properties sqrt2

-- Multiple algebraic extensions
sqrt3:= rootOf(sqrt3**2 - 3);



                                                                 Type: AlgebraicNumber
                                            Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
cbrt2:= rootOf(cbrt2**3 - 2);



                                                                 Type: AlgebraicNumber
                                                        Time: 0.02 (IN) = 0.02 sec
-- => 2 cbrt2 + 8 sqrt3 + 18 cbrt2^2 + 12 cbrt2 sqrt3 + 9
(cbrt2 + sqrt3)**4


                                      2
   (37)  (12cbrt2 + 8)sqrt3 + 18cbrt2  + 2cbrt2 + 9
                                                                 Type: AlgebraicNumber
                                            Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
)clear properties sqrt3 cbrt2

-- Factor polynomials over finite fields and field extensions
p:= x**4 - 3*x**2 + 1


          4     2
   (38)  x  - 3x  + 1
                                                              Type: Polynomial Integer
                                                        Time: 0.02 (IN) = 0.02 sec
factor(p)
```

```
           2           2
   (39)  (x  - x - 1)(x  + x - 1)
                                            Type: Factored Polynomial Integer
                                                   Time: 0.03 (EV) = 0.03 sec
-- => (x - 2)^2 (x + 2)^2  mod  5
factor(p :: Polynomial(PrimeField(5)))


                2       2
   (40)  (x + 2) (x + 3)
                                       Type: Factored Polynomial PrimeField 5
               Time: 0.12 (IN) + 0.12 (EV) + 0.02 (OT) + 0.05 (GC) = 0.30 sec
expand(%)


          4     2
   (41)  x  + 2x  + 1
                                                Type: Polynomial PrimeField 5
                                                   Time: 0.02 (IN) = 0.02 sec
-- => (x^2 + x + 1) (x^9 - x^8 + x^6 - x^5 + x^3 - x^2 + 1)  mod  65537
--    [Paul Zimmermann]
factor(x**11 + x + 1 :: Polynomial(PrimeField(65537)))


           2           9         8    6         5    3         2
   (42)  (x  + x + 1)(x  + 65536x  + x  + 65536x  + x  + 65536x  + 1)
                                   Type: Factored Polynomial PrimeField 65537
                           Time: 0.13 (IN) + 0.13 (EV) + 0.03 (OT) = 0.30 sec
-- => (x - phi) (x + phi) (x - phi + 1) (x + phi - 1)
--    where phi^2 - phi - 1 = 0 or phi = (1 +- sqrt(5))/2
phi:= rootOf(phi**2 - phi - 1);


                                                        Type: AlgebraicNumber
                                                   Time: 0.03 (IN) = 0.03 sec
factor(p, [phi])


   (44)  (x - phi)(x - phi + 1)(x + phi - 1)(x + phi)
                                    Type: Factored Polynomial AlgebraicNumber
                                       Time: 0.80 (EV) + 0.02 (OT) = 0.82 sec
)clear properties phi p

expand((x - 2*y**2 + 3*z**3)**20);
```

```
                                                      Type: Polynomial Integer
               Time: 0.17 (IN) + 0.55 (EV) + 0.03 (OT) + 0.42 (GC) = 1.17 sec
factor(%)


            3     2     20
   (46)  (3z  - 2y  + x)
                                             Type: Factored Polynomial Integer
                                                    Time: 0.13 (EV) = 0.13 sec
expand((sin(x) - 2*cos(y)**2 + 3*tan(z)**3)**20);


                                                      Type: Expression Integer
                           Time: 0.07 (IN) + 1.30 (EV) + 0.07 (OT) = 1.43 sec
factor(%)


   (48)
                     60                                         2       57
     3486784401tan(z)   + (23245229340sin(x) - 46490458680cos(y) )tan(z)
   +
                         2                     2                           4
       (73609892910sin(x)  - 294439571640cos(y) sin(x) + 294439571640cos(y) )
    *
             54
       tan(z)
   +
                           3                     2      2
         147219785820sin(x)  - 883318714920cos(y) sin(x)
       +
                            4                            6
         1766637429840cos(y) sin(x) - 1177758286560cos(y)
    *
             51
       tan(z)
   +
                           4                      2      3
         208561363245sin(x)  - 1668490905960cos(y) sin(x)
       +
                            4      2                      6
         5005472717880cos(y) sin(x)  - 6673963623840cos(y) sin(x)
       +
                            8
         3336981811920cos(y)
    *
             48
```

```
    tan(z)
+
                        5                      2      4
      222465454128sin(x)  - 2224654541280cos(y) sin(x)
    +
                         4      3                       6      2
      8898618165120cos(y) sin(x)  - 17797236330240cos(y) sin(x)
    +
                          8                            10
      17797236330240cos(y) sin(x) - 7118894532096cos(y)
 *
          45
    tan(z)
+
                        6                      2      5
      185387878440sin(x)  - 2224654541280cos(y) sin(x)
    +
                          4      4                       6      3
      11123272706400cos(y) sin(x)  - 29662060550400cos(y) sin(x)
    +
                          8      2                       10
      44493090825600cos(y) sin(x)  - 35594472660480cos(y)  sin(x)
    +
                          12
      11864824220160cos(y)
 *
          42
    tan(z)
+
                        7                      2      6
      123591918960sin(x)  - 1730286865440cos(y) sin(x)
    +
                          4      5                       6      4
      10381721192640cos(y) sin(x)  - 34605737308800cos(y) sin(x)
    +
                          8      3                       10      2
      69211474617600cos(y) sin(x)  - 83053769541120cos(y)  sin(x)
    +
                          12                             14
      55369179694080cos(y)  sin(x) - 15819765626880cos(y)
 *
          39
    tan(z)
+
                       8                      2      7
      66945622770sin(x)  - 1071129964320cos(y) sin(x)
```

```
    +
                         4      6                       6      5
      7497909750240cos(y) sin(x)  - 29991639000960cos(y) sin(x)
    +
                          8      4                        10       3
      74979097502400cos(y) sin(x)  - 119966556003840cos(y)  sin(x)
    +
                           12       2                       14
      119966556003840cos(y)  sin(x)  - 68552317716480cos(y)  sin(x)
    +
                          16
      17138079429120cos(y)
 *
          36
    tan(z)
+
                       9                     2      8
      29753610120sin(x)  - 535564982160cos(y) sin(x)
    +
                         4      7                       6      6
      4284519857280cos(y) sin(x)  - 19994426000640cos(y) sin(x)
    +
                          8      5                        10       4
      59983278001920cos(y) sin(x)  - 119966556003840cos(y)  sin(x)
    +
                           12       3                        14       2
      159955408005120cos(y)  sin(x)  - 137104635432960cos(y)  sin(x)
    +
                          16                              18
      68552317716480cos(y)  sin(x) - 15233848381440cos(y)
 *
          33
    tan(z)
+
                       10                     2      9
      10909657044sin(x)   - 218193140880cos(y) sin(x)
    +
                         4      8                       6      7
      1963738267920cos(y) sin(x)  - 10473270762240cos(y) sin(x)
    +
                          8      6                       10       5
      36656447667840cos(y) sin(x)  - 87975474402816cos(y)  sin(x)
    +
                           12       4                        14       3
      146625790671360cos(y)  sin(x)  - 167572332195840cos(y)  sin(x)
    +
```

```
                           16      2                       18
      125679249146880cos(y)  sin(x)  - 55857444065280cos(y)  sin(x)
    +
                          20
      11171488813056cos(y)
 *
          30
    tan(z)
+
                      11                    2      10
      3305956680sin(x)   - 72731046960cos(y) sin(x)
    +
                        4      9                      6      8
      727310469600cos(y) sin(x)  - 4363862817600cos(y) sin(x)
    +
                          8      7                       10      6
      17455451270400cos(y) sin(x)  - 48875263557120cos(y)  sin(x)
    +
                          12      5                        14      4
      97750527114240cos(y)  sin(x)  - 139643610163200cos(y)  sin(x)
    +
                           16      3                       18      2
      139643610163200cos(y)  sin(x)  - 93095740108800cos(y)  sin(x)
    +
                          20                            22
      37238296043520cos(y)  sin(x) - 6770599280640cos(y)
 *
          27
    tan(z)
+
                     12                    2      11
      826489170sin(x)   - 19835740080cos(y) sin(x)
    +
                        4      10                      6      9
      218193140880cos(y) sin(x)   - 1454620939200cos(y) sin(x)
    +
                         8      8                       10      7
      6545794226400cos(y) sin(x)  - 20946541524480cos(y)  sin(x)
    +
                          12      6                       14      5
      48875263557120cos(y)  sin(x)  - 83786166097920cos(y)  sin(x)
    +
                           16      4                       18      3
      104732707622400cos(y)  sin(x)  - 93095740108800cos(y)  sin(x)
    +
                          20      2                       22
```

```
      55857444065280cos(y)  sin(x)  - 20311797841920cos(y)  sin(x)
    +
                         24
      3385299640320cos(y)
 *
          24
    tan(z)
+
                     13                   2      12
      169536240sin(x)   - 4407942240cos(y) sin(x)
    +
                       4      11                     6      10
      52895306880cos(y) sin(x)   - 387898917120cos(y) sin(x)
    +
                         8      9                      10      8
      1939494585600cos(y) sin(x)  - 6982180508160cos(y)  sin(x)
    +
                          12      7                       14      6
      18619148021760cos(y)  sin(x)  - 37238296043520cos(y)  sin(x)
    +
                          16      5                       18      4
      55857444065280cos(y)  sin(x)  - 62063826739200cos(y)  sin(x)
    +
                          20      3                       22      2
      49651061391360cos(y)  sin(x)  - 27082397122560cos(y)  sin(x)
    +
                         24                            26
      9027465707520cos(y)  sin(x) - 1388840878080cos(y)
 *
          21
    tan(z)
+
                    14                  2      13
      28256040sin(x)   - 791169120cos(y) sin(x)
    +
                       4      12                    6      11
      10285198560cos(y) sin(x)   - 82281588480cos(y) sin(x)
    +
                        8      10                      10      9
      452548736640cos(y) sin(x)   - 1810194946560cos(y)  sin(x)
    +
                         12      8                       14      7
      5430584839680cos(y)  sin(x)  - 12412765347840cos(y)  sin(x)
    +
                          16      6                       18      5
      21722339358720cos(y)  sin(x)  - 28963119144960cos(y)  sin(x)
```

```
    +
                          20      4                       22      3
      28963119144960cos(y)  sin(x)  - 21064086650880cos(y)  sin(x)
    +
                          24      2                      26
      10532043325440cos(y)  sin(x)  - 3240628715520cos(y)  sin(x)
    +
                        28
      462946959360cos(y)
 *
          18
    tan(z)
+
                   15                  2      14                   4      13
      3767472sin(x)   - 113024160cos(y) sin(x)   + 1582338240cos(y) sin(x)
    +
                         6      12                    8      11
      - 13713598080cos(y) sin(x)   + 82281588480cos(y) sin(x)
    +
                          10      10                      12      9
      - 362038989312cos(y)  sin(x)   + 1206796631040cos(y)  sin(x)
    +
                           14      8                       16      7
      - 3103191336960cos(y)  sin(x)   + 6206382673920cos(y)  sin(x)
    +
                           18      6                        20      5
      - 9654373048320cos(y)  sin(x)   + 11585247657984cos(y)  sin(x)
    +
                            22      4                       24      3
      - 10532043325440cos(y)  sin(x)   + 7021362216960cos(y)  sin(x)
    +
                           26      2                      28
      - 3240628715520cos(y)  sin(x)   + 925893918720cos(y)  sin(x)
    +
                          30
      - 123452522496cos(y)
 *
          15
    tan(z)
+
                  16                 2      15                  4      14
      392445sin(x)   - 12558240cos(y) sin(x)   + 188373600cos(y) sin(x)
    +
                        6      13                    8      12
      - 1758153600cos(y) sin(x)   + 11427998400cos(y) sin(x)
    +
```

```
                         10      11                     12      10
      - 54854392320cos(y)  sin(x)   + 201132771840cos(y)  sin(x)
    +
                          14      9                      16      8
      - 574665062400cos(y)  sin(x)  + 1292996390400cos(y)  sin(x)
    +
                           18      7                      20      6
      - 2298660249600cos(y)  sin(x)  + 3218124349440cos(y)  sin(x)
    +
                           22      5                      24      4
      - 3510681108480cos(y)  sin(x)  + 2925567590400cos(y)  sin(x)
    +
                           26      3                     28      2
      - 1800349286400cos(y)  sin(x)  + 771578265600cos(y)  sin(x)
    +
                          30                          32
      - 205754204160cos(y)  sin(x) + 25719275520cos(y)
 *
          12
    tan(z)
+
                 17                2      16                 4      15
      30780sin(x)   - 1046520cos(y) sin(x)   + 16744320cos(y) sin(x)
    +
                       6      14                   8      13
      - 167443200cos(y) sin(x)   + 1172102400cos(y) sin(x)
    +
                        10      12                   12      11
      - 6094932480cos(y)  sin(x)   + 24379729920cos(y)  sin(x)
    +
                         14      10                    16      9
      - 76622008320cos(y)  sin(x)   + 191555020800cos(y)  sin(x)
    +
                          18      8                    20      7
      - 383110041600cos(y)  sin(x)  + 612976066560cos(y)  sin(x)
    +
                          22      6                    24      5
      - 780151357440cos(y)  sin(x)  + 780151357440cos(y)  sin(x)
    +
                          26      4                    28      3
      - 600116428800cos(y)  sin(x)  + 342923673600cos(y)  sin(x)
    +
                          30      2                   32
      - 137169469440cos(y)  sin(x)  + 34292367360cos(y)  sin(x)
    +
                         34
```

```
      - 4034396160cos(y)
 *
          9
    tan(z)
+
                18              2      17                4      16
      1710sin(x)   - 61560cos(y) sin(x)   + 1046520cos(y) sin(x)
    +
                      6      15                 8      14
      - 11162880cos(y) sin(x)   + 83721600cos(y) sin(x)
    +
                       10      13                   12      12
      - 468840960cos(y)  sin(x)   + 2031644160cos(y)  sin(x)
    +
                        14      11                    16      10
      - 6965637120cos(y)  sin(x)   + 19155502080cos(y)  sin(x)
    +
                         18      9                    20      8
      - 42567782400cos(y)  sin(x)  + 76622008320cos(y)  sin(x)
    +
                          22      7                     24      6
      - 111450193920cos(y)  sin(x)  + 130025226240cos(y)  sin(x)
    +
                          26      5                    28      4
      - 120023285760cos(y)  sin(x)  + 85730918400cos(y)  sin(x)
    +
                         30      3                    32      2
      - 45723156480cos(y)  sin(x)  + 17146183680cos(y)  sin(x)
    +
                        34                        36
      - 4034396160cos(y)  sin(x) + 448266240cos(y)
 *
          6
    tan(z)
+
              19             2      18              4      17
      60sin(x)   - 2280cos(y) sin(x)   + 41040cos(y) sin(x)
    +
                    6      16                8      15
      - 465120cos(y) sin(x)   + 3720960cos(y) sin(x)
    +
                      10      14                  12      13
      - 22325760cos(y)  sin(x)   + 104186880cos(y)  sin(x)
    +
                       14      12                   16      11
      - 386979840cos(y)  sin(x)   + 1160939520cos(y)  sin(x)
```

```
    +
                        18      10                   20      9
      - 2837852160cos(y)  sin(x)   + 5675704320cos(y)  sin(x)
    +
                        22      8                    24      7
      - 9287516160cos(y)  sin(x) + 12383354880cos(y)  sin(x)
    +
                         26      6                    28      5
      - 13335920640cos(y)  sin(x)  + 11430789120cos(y)  sin(x)
    +
                        30      4                   32      3
      - 7620526080cos(y)  sin(x) + 3810263040cos(y)  sin(x)
    +
                      34      2                  36                       38
    - 1344798720cos(y)  sin(x) + 298844160cos(y)  sin(x) - 31457280cos(y)
 *
          3
    tan(z)
+
        20           2      19            4      18             6      17
  sin(x)   - 40cos(y) sin(x)   + 760cos(y) sin(x)   - 9120cos(y) sin(x)
+
             8      16               10      15                12      14
  77520cos(y) sin(x)   - 496128cos(y)  sin(x)   + 2480640cos(y)  sin(x)
+
                 14      13                 16      12
  - 9922560cos(y)  sin(x)   + 32248320cos(y)  sin(x)
+
                  18      11                  20      10
  - 85995520cos(y)  sin(x)   + 189190144cos(y)  sin(x)
+
                   22      9                  24      8
  - 343982080cos(y)  sin(x) + 515973120cos(y)  sin(x)
+
                   26      7                  28      6
  - 635043840cos(y)  sin(x) + 635043840cos(y)  sin(x)
+
                   30      5                  32      4
  - 508035072cos(y)  sin(x) + 317521920cos(y)  sin(x)
+
                   34      3                 36      2
  - 149422080cos(y)  sin(x) + 49807360cos(y)  sin(x)
+
                  38                      40
  - 10485760cos(y)  sin(x) + 1048576cos(y)
                                         Type: Factored Expression Integer
```

```
                                       Time: 0.02 (EV) + 0.88 (OT) = 0.90 sec
-- expand[(1 - c^2)^5 (1 - s^2)^5 (c^2 + s^2)^10] => c^10 s^10 when
-- c^2 + s^2 = 1   [modification of a problem due to Richard Liska]
expand((1 - c**2)**5 * (1 - s**2)**5 * (c**2 + s**2)**10);


                                                     Type: Polynomial Integer
                           Time: 0.25 (IN) + 0.02 (EV) + 0.02 (OT) = 0.28 sec
groebner([%, c**2 + s**2 - 1])


           2    2      20     18      16      14     12    10
   (50)  [s  + c  - 1,c   - 5c   + 10c   - 10c   + 5c   - c  ]
                                                Type: List Polynomial Integer
                           Time: 0.02 (IN) + 0.17 (EV) + 0.02 (OT) = 0.20 sec
map(factor, %)


           2    2            5 10       5
   (51)  [s  + c  - 1,(c - 1) c  (c + 1) ]
                                       Type: List Factored Polynomial Integer
                           Time: 0.03 (IN) + 0.02 (EV) + 0.02 (OT) = 0.07 sec
-- => (x + y) (x - y)  mod  3
factor(4*x**2 - 21*x*y + 20*y**2 :: Polynomial(PrimeField(3)))

   There are 20 exposed and 17 unexposed library operations named **
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op **
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named **
      with argument type(s)
                                 Variable y
                           Polynomial PrimeField 3

-- => 1/4 (x + y) (2 x +  y [-1 + i sqrt(3)]) (2 x + y [-1 - i sqrt(3)])
factor(x**3 + y**3, [rootOf(isqrt3**2 + 3)])


              - isqrt3 - 1              isqrt3 - 1
   (52)  (y + ------------ x)(y + x)(y + ---------- x)
                    2                         2
                                    Type: Factored Polynomial AlgebraicNumber
```

```
               Time: 0.05 (IN) + 2.33 (EV) + 0.07 (OT) + 0.43 (GC) = 2.88 sec
-- Partial fraction decomposition => 3/(x + 2) - 2/(x + 1) + 2/(x + 1)^2
(x**2 + 2*x + 3)/(x**3 + 4*x**2 + 5*x + 2)


             2
            x  + 2x + 3
   (53)  -----------------
          3     2
         x  + 4x  + 5x + 2
                                            Type: Fraction Polynomial Integer
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
fullPartialFraction( _
   % :: Fraction UnivariatePolynomial(x, Fraction Integer))


             2         2        3
   (54)  - ----- + -------- + -----
           x + 1          2   x + 2
                   (x + 1)
Type: FullPartialFractionExpansion(Fraction Integer,UnivariatePolynomial(x,Fraction Integer)
                           Time: 0.13 (IN) + 0.20 (EV) + 0.05 (OT) = 0.38 sec
-- Noncommutative algebra: note that (A B C)^(-1) = C^(-1) B^(-1) A^(-1)
-- => A B C A C B - C^(-1) B^(-1) C B
A : SquareMatrix(2, Integer);

                                                                   Type: Void
                                                                  Time: 0 sec
B : SquareMatrix(2, Integer);

                                                                   Type: Void
                                                                  Time: 0 sec
C : SquareMatrix(2, Integer);

                                                                   Type: Void
                                                                  Time: 0 sec
(A*B*C - (A*B*C)**(-1)) * A*C*B


   A is declared as being in SquareMatrix(2,Integer) but has not been
      given a value.
-- Jacobi's identity: [A, B, C] + [B, C, A] + [C, A, B] = 0 where [A, B, C] =
-- [A, [B, C]] and [A, B] = A B - B A is the commutator of A and B
comm2(A, B) == A * B - B * A;

                                                                   Type: Void
```

```
                                                                   Time: 0 sec
comm3(A, B, C) == comm2(A, comm2(B, C));

                                                                    Type: Void
                                                                   Time: 0 sec
comm2(A, B)


   A is declared as being in SquareMatrix(2,Integer) but has not been
      given a value.
comm3(A, B, C) + comm3(B, C, A) + comm3(C, A, B)


   A is declared as being in SquareMatrix(2,Integer) but has not been
      given a value.
)clear properties A B C comm2 comm3

-- ---------- Quit ----------
)quit


real   1461.7
user   485.5
sys    0.6
```

9 Trigonometry

```
-- ---------- Trigonometry ----------
-- => - [(sqrt(5) + 1) sqrt(2)]/[(sqrt(5) - 1) sqrt(sqrt(5) + 5)]
--    = - sqrt[1 + 2/sqrt(5)]
-- From B. F. Caviness, Robert P. Gilbert, Wolfram Koepf, Roman Shtokhamer and
-- David W. Wood, _An Introduction to Applied Symbolic Computation using
-- MACSYMA_, University of Delaware, draft of December 14, 1993, section 2.3.3.
tan(7*%pi/10)


           7%pi
   (1)  tan(----)
            10
                                                      Type: Expression Integer
                Time: 0.48 (IN) + 0.18 (EV) + 0.14 (OT) + 0.20 (GC) = 1.0 sec
-- => - cos 3
sqrt((1 + cos(6))/2)
```

```
         +----------+
         |cos(6) + 1
   (2)   |----------
        \|     2
                                                     Type: Expression Integer
                                       Time: 0.18 (IN) + 0.03 (EV) = 0.22 sec
simplify(normalize(%))


         +-------+
         |      2
   (3)  \|cos(3)
                                                     Type: Expression Integer
               Time: 0.10 (IN) + 0.20 (EV) + 0.02 (OT) + 0.02 (GC) = 0.33 sec
-- cos(n pi) + sin((4 n - 1)/2 pi) => (-1)^n - 1 for integer n
cos(n*%pi) + sin((4*n - 1)/2 * %pi)


            (4n - 1)%pi
   (4)  sin(-----------) + cos(n %pi)
                 2
                                                     Type: Expression Integer
                           Time: 1.15 (IN) + 0.02 (EV) + 0.17 (OT) = 1.33 sec
-- cos(cos(n pi) pi) + sin(cos(n pi) pi/2) => -1 + (-1)^n for integer n
cos(cos(n*%pi)*%pi) + sin(cos(n*%pi)*%pi/2)


            %pi cos(n %pi)
   (5)  sin(--------------) + cos(%pi cos(n %pi))
                   2
                                                     Type: Expression Integer
                           Time: 0.12 (IN) + 0.02 (EV) + 0.07 (OT) = 0.20 sec
-- sin([n^5/5 + n^4/2 + n^3/3 - n/30] pi) => 0 for integer n
-- [Paul Zimmermann]
sin((n**5/5 + n**4/2 + n**3/3 - n/30) * %pi)


               5      4      3
            (6n  + 15n  + 10n  - n)%pi
   (6)  sin(--------------------------)
                        30
                                                     Type: Expression Integer
                                       Time: 0.37 (IN) + 0.05 (OT) = 0.42 sec
-- | cos x |, | sin x | => - cos x, - sin x  for  - 3 pi < x < - 5/2 pi
```

```
--assume(-3*%pi < x, x < -5/2*%pi)
[abs(cos(x)), abs(sin(x))]


   (7)  [abs(cos(x)),abs(sin(x))]
                                                  Type: List Expression Integer
                           Time: 0.05 (IN) + 0.03 (EV) + 0.02 (OT) = 0.10 sec
--forget(-3*%pi < x, x < -5/2*%pi)
-- Trigonometric manipulations---these are typically difficult for students
r:= cos(3*x)/cos(x)


        cos(3x)
   (8)  -------
         cos(x)
                                                       Type: Expression Integer
                                       Time: 0.03 (IN) + 0.02 (EV) = 0.05 sec
-- => cos(x)^2 - 3 sin(x)^2 or similar
real(complexNormalize(r))


                  2          2
   (9)  - 2sin(x)  + 2cos(x)  - 1
                                                       Type: Expression Integer
               Time: 0.07 (IN) + 1.45 (EV) + 0.03 (OT) + 0.02 (GC) = 1.57 sec
-- => 2 cos(2 x) - 1
real(normalize(simplify(complexNormalize(r))))


   (10)  2cos(2x) - 1
                                                       Type: Expression Integer
                                       Time: 0.02 (IN) + 1.10 (EV) = 1.12 sec
-- Use rewrite rules => cos(x)^2 - 3 sin(x)^2
sincosAngles:= rule ( _
   cos((n | integer?(n)) * x) == _
      cos((n - 1)*x) * cos(x) - sin((n - 1)*x) * sin(x); _
   sin((n | integer?(n)) * x) == _
      sin((n - 1)*x) * cos(x) + cos((n - 1)*x) * sin(x) )


   (11)
   {cos(n x) == - sin(x)sin((n - 1)x) + cos(x)cos((n - 1)x),
    sin(n x) == cos(x)sin((n - 1)x) + cos((n - 1)x)sin(x)}
                            Type: Ruleset(Integer,Integer,Expression Integer)
               Time: 0.25 (IN) + 0.15 (EV) + 0.07 (OT) + 0.02 (GC) = 0.48 sec
sincosAngles r
```

```
                 2         2
   (12)  - 3sin(x)  + cos(x)
                                                     Type: Expression Integer
              Time: 0.02 (IN) + 0.20 (EV) + 0.02 (OT) + 0.02 (GC) = 0.25 sec
)clear properties r

-- Here is a tricky way of writing 0/0
expr:= (tan(x)**2 + 1 - sec(x)**2)/(sin(x)**2 + cos(x)**2 - 1)


              2         2
         tan(x)  - sec(x)  + 1
   (13)  ---------------------
              2         2
         sin(x)  + cos(x)  - 1
                                                     Type: Expression Integer
                          Time: 0.08 (IN) + 0.02 (EV) + 0.02 (OT) = 0.12 sec
-- Let's try simplifying this expression!
simplify(expr)


            1
   (14)  -------
              2
         cos(x)
                                                     Type: Expression Integer
                                                   Time: 0.07 (EV) = 0.07 sec
normalize(expr)


   >> Error detected within library code:
   catdef: division by zero

initial (15) ->
real   8.9
user   8.2
sys    0.4

--------------------------------------------------------------------------------

Thu Apr 17 07:23:53 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
```

```
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Trigonometry ----------
expr:= (tan(x)**2 + 1 - sec(x)**2)/(sin(x)**2 + cos(x)**2 - 1)


              2         2
        tan(x)  - sec(x)  + 1
   (1)  ---------------------
              2         2
        sin(x)  + cos(x)  - 1
                                                     Type: Expression Integer
               Time: 0.32 (IN) + 0.40 (EV) + 0.27 (OT) + 0.20 (GC) = 1.18 sec
-- Let's try simplifying this expression!
complexNormalize(expr)


   >> Error detected within library code:
   catdef: division by zero

initial (2) ->
real   5.5
user   2.0
sys    0.3

-------------------------------------------------------------------------------

Thu Apr 17 07:25:51 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
```

```
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Trigonometry ----------
expr:= (tan(x)**2 + 1 - sec(x)**2)/(sin(x)**2 + cos(x)**2 - 1)


              2         2
        tan(x)  - sec(x)  + 1
   (1)  ---------------------
              2         2
        sin(x)  + cos(x)  - 1
                                                     Type: Expression Integer
               Time: 0.30 (IN) + 0.45 (EV) + 0.23 (OT) + 0.20 (GC) = 1.18 sec
-- What is its limit at zero?
limit(expr, x = 0)


   (2)  0
                        Type: Union(OrderedCompletion Expression Integer,...)
               Time: 0.35 (IN) + 8.02 (EV) + 0.05 (OT) + 0.53 (GC) = 8.95 sec
-- What is the derivative?
dexpr:= D(expr, x)


               3             2
        2tan(x)  + (- 2sec(x)  + 2)tan(x)
   (3)  ---------------------------------
                    2         2
              sin(x)  + cos(x)  - 1
                                                     Type: Expression Integer
                           Time: 0.02 (IN) + 0.13 (EV) + 0.02 (OT) = 0.17 sec
```

```
simplify(dexpr)


        2sin(x)
   (4)  -------
              3
        cos(x)
                                                       Type: Expression Integer
                           Time: 0.03 (IN) + 0.12 (EV) + 0.03 (OT) = 0.18 sec
normalize(dexpr)


   >> Error detected within library code:
   catdef: division by zero

initial (5) ->
real   32.0
user   10.7
sys    0.4

--------------------------------------------------------------------------------

Thu Apr 17 07:28:07 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Trigonometry ----------
expr:= (tan(x)**2 + 1 - sec(x)**2)/(sin(x)**2 + cos(x)**2 - 1)
```

```
              2         2
        tan(x)  - sec(x)  + 1
   (1)  ---------------------
              2         2
        sin(x)  + cos(x)  - 1
                                                       Type: Expression Integer
              Time: 0.37 (IN) + 0.40 (EV) + 0.22 (OT) + 0.20 (GC) = 1.18 sec
-- What is the derivative?
dexpr:= D(expr, x)


               3             2
        2tan(x)  + (- 2sec(x)  + 2)tan(x)
   (2)  ---------------------------------
                    2         2
              sin(x)  + cos(x)  - 1
                                                       Type: Expression Integer
                          Time: 0.15 (EV) + 0.02 (OT) + 0.02 (GC) = 0.18 sec
complexNormalize(dexpr)


   >> Error detected within library code:
   catdef: division by zero

initial (3) ->
real   5.6
user   2.1
sys    0.3
```

10 Special Functions

```
initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Special Functions ----------
-- Bernoulli numbers: B_16 => -3617/510   [Gradshteyn and Ryzhik 9.71]
bernoulli(16)
```

```
          3617
   (1)  - ----
           510
                                                       Type: Fraction Integer
               Time: 0.02 (IN) + 0.07 (EV) + 0.03 (OT) + 0.03 (GC) = 0.15 sec
-- d/dk E(phi, k) => [E(phi, k) - F(phi, k)]/k  where  F(phi, k) and E(phi, k)
-- are elliptic integrals of the 1st and 2nd kind, respectively
-- [Gradshteyn and Ryzhik 8.123(3)]
--D(E(phi, k), k)
-- Jacobian elliptic functions: d/du dn u => -k^2 sn u cn u
-- [Gradshteyn and Ryzhik 8.158(3)]
--D(dn(u), u)
-- => -2 sqrt(pi)   [Gradshteyn and Ryzhik 8.338(3)]
Gamma(-1/2)


   (2)  - 3.5449077018110313
                                                            Type: DoubleFloat
                           Time: 0.03 (IN) + 0.03 (OT) + 0.02 (GC) = 0.08 sec
% + 2*sqrt(%pi)


   (3)  4.4408920985006262e-16
                                                 Type: Expression DoubleFloat
                            Time: 0.68 (IN) + 0.17 (OT) + 0.15 (GC) = 1.0 sec
-- psi(1/3) => - Euler's_constant - pi/2 sqrt(1/3) - 3/2 log 3  where  psi(x)
-- is the psi function [= d/dx log Gamma(x)]   [Gradshteyn and Ryzhik 8.366(6)]
digamma(1/3)


   (4)  - 3.1320337800208065
                                                            Type: DoubleFloat
                                                   Time: 0.02 (IN) = 0.02 sec
% + %pi/2*sqrt(1/3) + 3/2*log(3)


   (5)  - 0.57721566490153275
                                                 Type: Expression DoubleFloat
                           Time: 0.52 (IN) + 0.10 (EV) + 0.05 (OT) = 0.67 sec
-- Bessel function of the first kind of order 2 => 0.04158 + 0.24740 i
besselJ(2, 1 + %i)


   (6)  0.041579886943962127 + 0.2473976415133064%i
                                                    Type: Complex DoubleFloat
```

```
               Time: 0.12 (IN) + 0.02 (EV) + 0.05 (OT) + 0.03 (GC) = 0.22 sec
-- => 12/pi^2   [Gradshteyn and Ryzhik 8.464(6)]
besselJ(-5/2, %pi/2)


   (7)  1.2158542037080535
                                                      Type: DoubleFloat
                                   Time: 0.07 (IN) + 0.03 (OT) = 0.10 sec
% - 12/%pi**2


   (8)  2.2204460492503131e-16
                                                      Type: DoubleFloat
                                                 Time: 0.03 (IN) = 0.03 sec
-- => sqrt(2/(pi z)) (sin z/z - cos z)   [Gradshteyn and Ryzhik 8.464(3)]
besselJ(3/2, z)


               3
   (9)  besselJ(-,z)
               2
                                                 Type: Expression Integer
                         Time: 0.30 (IN) + 0.02 (EV) + 0.02 (OT) = 0.33 sec
-- d/dz J_0(z) => - J_1(z)   [Gradshteyn and Ryzhik 8.473(4)]
D(besselJ(0, z), z)


         - besselJ(1,z) + besselJ(- 1,z)
   (10)  -------------------------------
                       2
                                                 Type: Expression Integer
                         Time: 0.05 (IN) + 0.05 (EV) + 0.02 (GC) = 0.12 sec
-- Associated Legendre (spherical) function of the 1st kind: P^mu_nu(0)
-- => 2^mu sqrt(pi) / [Gamma([nu - mu]/2 + 1) Gamma([- nu - mu + 1]/2)]
--    [Gradshteyn and Ryzhik 8.756(1)]
--P(mu, nu, 0)
-- P^1_3(x) => -3/2 sqrt(1 - x^2) (5 x^2 - 1)
--            [Gradshteyn and Ryzhik 8.813(4)]
--P(1, 3, x)
-- nth Chebyshev polynomial of the 1st kind: T_n(x) => 0
-- [Gradshteyn and Ryzhik 8.941(1)]
chebyshevT(1008, x) - 2*x*chebyshevT(1007, x) + chebyshevT(1006, x)


   (11)  0
                                                 Type: Polynomial Integer
```

```
                        Time: 0.03 (IN) + 30.50 (EV) + 11.38 (GC) = 41.91 sec
-- T_n(-1) => (-1)^n   [Gradshteyn and Ryzhik 8.944(2)]
chebyshevT(n, -1)

   There are 1 exposed and 0 unexposed library operations named
      chebyshevT having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                           )display op chebyshevT
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      chebyshevT with argument type(s)
                                 Variable n
                                   Integer

-- => arcsin z/z   [Gradshteyn and Ryzhik 9.121(26)]
--hypergeometric([1/2, 1/2], [3/2], z**2)
-- => sin(n z)/(n sin z cos z)   [Gradshteyn and Ryzhik 9.121(17)]
--hypergeometric([(n + 2)/2, -(n - 2)/2], [3/2], sin(z)**2)
-- zeta'(0) => - 1/2 log(2 pi)   [Gradshteyn and Ryzhik 9.542(4)]
--subst(D(zeta(x), x), x = 0)
-- Dirac delta distribution => 3 f(4/5) + g'(1)
--f:= operator('f);
--g:= operator('g);
--integrate(f((x + 2)/5)*delta((x - 2)/3) - g(x)*D(delta(x - 1), x), x = 0..3)
--)clear properties f g
-- Define an antisymmetric function f
f:= operator('f);



                                                                Type: BasicOperator
                                                         Time: 0.07 (IN) = 0.07 sec
-- Test it out => [-f(a, b, c), 0]
[f(c, b, a), f(c, b, c)]


   (13)  [f(c,b,a),f(c,b,c)]
                                                      Type: List Expression Integer
                                             Time: 0.17 (IN) + 0.02 (OT) = 0.18 sec
)clear properties f

-- ---------- Quit ----------
)quit
```

```
real   68.3
user   34.3
sys    0.4
```

11 The Complex Domain

```
-- ---------- The Complex Domain ----------
-- Complex functions---separate into their real and imaginary parts.
-- Here, variables default to REAL.
-- [Re(x + i y), Im(x + i y)] => [Re(x) - Im(y), Im(x) + Re(y)]
-- for x and y complex
[real(x + %i*y), imag(x + %i*y)]


   (1)  [x,y]
                                                 Type: List Expression Integer
               Time: 0.55 (IN) + 0.30 (EV) + 0.22 (OT) + 0.25 (GC) = 1.32 sec
x : Complex Expression Integer

                                                                    Type: Void
                                                                   Time: 0 sec
y : Complex Expression Integer

                                                                    Type: Void
                                                                   Time: 0 sec
[real(x + %i*y), imag(x + %i*y)]


   (4)  [x,y]
                                                 Type: List Expression Integer
                                        Time: 0.52 (IN) + 0.07 (OT) = 0.58 sec
)clear properties x y

-- => 1   [W. Kahan]
abs(3 - sqrt(7) + %i*sqrt(6*sqrt(7) - 15))


               +----------+
               |  +-+          +-+
   (5)  abs(%i\|6\|7  - 15  - \|7  + 3)
                                              Type: Expression Complex Integer
                           Time: 0.20 (IN) + 0.10 (EV) + 0.05 (OT) = 0.35 sec
```

```
complexForm(%)


   (6)  1
                                             Type: Complex Expression Integer
                                                   Time: 0.07 (EV) = 0.07 sec
-- => 1/sqrt(a^2 + (1/a + b)^2) for real a, b
abs(1/(a + %i/a + %i*b))


                  %i a
   (7)  abs(---------------)
                          2
            a b - %i a  + 1
                                             Type: Expression Complex Integer
                           Time: 0.78 (IN) + 0.02 (EV) + 0.07 (OT) = 0.87 sec
complexForm(%)


         +--------------------+
         |          2
         |         a
   (8)   |--------------------
         | 2 2           4
        \|a b  + 2a b + a  + 1
                                             Type: Complex Expression Integer
                                       Time: 0.12 (EV) + 0.02 (GC) = 0.13 sec
-- => log 5 + i arctan(4/3)
complexForm(log(3 + 4*%i))


        log(25)        4
   (9)  ------- + atan(-)%i
           2           3
                                             Type: Complex Expression Integer
                                       Time: 0.03 (IN) + 0.05 (EV) = 0.08 sec
-- => [sin(x) cos(x) + i sinh(y) cosh(y)] / [cos(x)^2 + sinh(y)^2]
simplify(complexForm(tan(x + %i*y)))


                  - 2y          +---+  - 4y    +---+
         4cos(x)%e    sin(x) - \|- 1 %e     + \|- 1
   (10)  -------------------------------------------
                      2       - 2y      - 4y
              (4cos(x)  - 2)%e     + %e     + 1
                                                     Type: Expression Integer
```

```
               Time: 0.10 (IN) + 0.85 (EV) + 0.03 (OT) + 0.03 (GC) = 1.02 sec
simplify(complexNormalize(%))


                      +---+
            +---+  2x\|- 1  - 2y    +---+
         - \|- 1 %e              + \|- 1
   (11)  --------------------------------
                     +---+
                  2x\|- 1  - 2y
                %e              + 1
                                                     Type: Expression Integer
                                       Time: 1.60 (EV) + 0.02 (OT) = 1.62 sec
simp(e) == [simplify(e), normalize(e), complexNormalize(e)]

                                                                   Type: Void
                                                                  Time: 0 sec
-- Check for branch abuse.  See 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.  This first
-- expression can simplify to sqrt(x y)/sqrt(x), but no further in general
-- (consider what happens when x, y = -1).  sqrt(x y) = sqrt(x) sqrt(y) if
-- either x >= 0 or y >= 0 or both x and y lie in the right-half plane
-- (Re x, Re y > 0) [considering principal values].
sqrt(x*y*abs(z)**2) / (sqrt(x)*abs(z))


          +-----------+
          |          2
         \|x y abs(z)
   (13)  --------------
                  +-+
           abs(z)\|x
                                                     Type: Expression Integer
               Time: 0.55 (IN) + 0.03 (EV) + 0.02 (OT) + 0.35 (GC) = 0.95 sec
simp(%)

   Compiling function simp with type Expression Integer -> List 
      Expression Integer 

           +-----------+  +-----------+  +-----------+
           |          2   |          2   |          2
          \|x y abs(z)   \|x y abs(z)   \|x y abs(z)
   (14)  [--------------,--------------,--------------]
                   +-+            +-+            +-+
            abs(z)\|x      abs(z)\|x      abs(z)\|x
```

```
                                                  Type: List Expression Integer
                            Time: 0.03 (IN) + 0.13 (EV) + 0.03 (OT) = 0.20 sec
x : Complex Expression Integer

                                                                     Type: Void
                                                                    Time: 0 sec
y : Complex Expression Integer

                                                                     Type: Void
                                                     Time: 0.02 (IN) = 0.02 sec
z : Complex Expression Integer

                                                                     Type: Void
                                                                    Time: 0 sec
sqrt(x*y*abs(z)**2) / (sqrt(x)*abs(z))


          +-----------+
          |          2
         \|x y abs(z)
   (18)  --------------
                  +-+
           abs(z)\|x
                                               Type: Complex Expression Integer
                                        Time: 0.10 (IN) + 0.05 (OT) = 0.15 sec
simp(%)

   Compiling function simp with type Complex Expression Integer -> List
      Expression Complex Integer

           +-----------+  +-----------+  +-----------+
           |          2   |          2   |          2
          \|x y abs(z)   \|x y abs(z)   \|x y abs(z)
   (19)  [--------------,--------------,--------------]
                   +-+            +-+            +-+
            abs(z)\|x      abs(z)\|x      abs(z)\|x
                                          Type: List Expression Complex Integer
                            Time: 0.32 (IN) + 0.13 (EV) + 0.07 (OT) = 0.52 sec
)clear properties x y z

-- Special case: sqrt(x y |z|^2)/(sqrt(x) |z|) => sqrt(y) [PV] for y >= 0
-- sqrt(1/z) = 1/sqrt(z) except when z is real and negative, in which case
-- sqrt(1/z) = - 1/sqrt(z) [considering principal values]
sqrt(1/z) - 1/sqrt(z)
```

```
   (20)  0
                                                     Type: Expression Integer
                                                   Time: 0.13 (IN) = 0.13 sec
z : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
sqrt(1/z) - 1/sqrt(z)


   (22)  0
                                             Type: Complex Expression Integer
                           Time: 0.02 (IN) + 0.02 (EV) + 0.02 (OT) = 0.05 sec
)clear properties z

-- Special case: sqrt(1/z) - 1/sqrt(z) => 0 [PV] for z > 0
-- Special case: sqrt(1/z) + 1/sqrt(z) => 0 [PV] for z < 0
-- sqrt(e^z) = e^(z/2) if and only if Im z is contained in the interval
-- ((4 n - 1) pi, (4 n + 1) pi] for n an integer: ..., (-5 pi, -3 pi],
-- (-pi, pi], (3 pi, 5 pi], ...; otherwise, sqrt(e^z) = - e^(z/2) [considering
-- principal values]
z : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
sqrt(%e**z) - %e**(z/2)


                    z
          +---+     -
          |  z      2
   (24)  \|%e   - %e
                                             Type: Complex Expression Integer
                           Time: 0.15 (IN) + 0.05 (EV) + 0.02 (OT) = 0.22 sec
simp(%)


                        +------+        +------+
                     z  |   z 2      z  |   z 2      z
           +---+     -  |   -        -  |   -        -
           |  z      2  |   2        2  |   2        2
   (25)  [\|%e   - %e ,\|(%e )   - %e ,\|(%e )   - %e ]
                                        Type: List Expression Complex Integer
                           Time: 0.22 (IN) + 0.42 (EV) + 0.03 (OT) = 0.67 sec
)clear properties z
```

```
-- Special case: sqrt(e^z) - e^(z/2) => 0 [PV] for z real
sqrt(%e**z) - %e**(z/2)


                    z
          +---+     -
          |  z      2
   (26)  \|%e   - %e
                                                      Type: Expression Integer
                           Time: 0.52 (IN) + 0.07 (EV) + 0.07 (OT) = 0.65 sec
simp(%)


                        +------+        +------+
                     z  |   z 2      z  |   z 2      z
           +---+     -  |   -        -  |   -        -
           |  z      2  |   2        2  |   2        2
   (27)  [\|%e   - %e ,\|(%e )   - %e ,\|(%e )   - %e ]
                                                 Type: List Expression Integer
                                        Time: 0.27 (EV) + 0.02 (OT) = 0.28 sec
-- The principal value of this expression is - e^(3 i) = - cos 3 - i sin 3
sqrt(%e**(6*%i))


          +-----+
          |  6%i
   (28)  \|%e
                                              Type: Expression Complex Integer
                           Time: 0.15 (IN) + 0.05 (EV) + 0.02 (OT) = 0.22 sec
simplify(complexForm(%))


                        sin(6)              sin(6)
                   atan(------)        atan(------)
          +---+         cos(6)              cos(6)
   (29)  \|- 1 sin(------------) + cos(------------)
                         2                   2
                                                      Type: Expression Integer
                           Time: 0.02 (IN) + 0.15 (EV) + 0.02 (OT) = 0.18 sec
% :: Complex Float


   (30)  0.9899924966 0044545727 - 0.1411200080 598672221 %i
                                                           Type: Complex Float
                                                    Time: 0.20 (IN) = 0.20 sec
-- log(e^z) = z if and only if Im z is contained in the interval (-pi, pi]
```

```
-- [considering principal values]
z : Complex Expression Integer

                                                                    Type: Void
                                                                   Time: 0 sec
log(%e**z)


   (32)  z
                                              Type: Complex Expression Integer
                                                    Time: 0.05 (EV) = 0.05 sec
)clear properties z

-- Special case: log(e^z) => z [PV] for z real
log(%e**z)


   (33)  z
                                                      Type: Expression Integer
                                        Time: 0.02 (IN) + 0.03 (EV) = 0.05 sec
-- The principal value of this expression is (10 - 4 pi) i
log(%e**(10*%i))


               10%i
   (34)  log(%e    )
                                              Type: Expression Complex Integer
                            Time: 0.02 (IN) + 0.03 (EV) + 0.03 (OT) = 0.08 sec
simplify(complexForm(%))


          +---+     sin(10)
   (35)  \|- 1 atan(-------)
                    cos(10)
                                                      Type: Expression Integer
                            Time: 0.02 (IN) + 0.07 (EV) + 0.02 (OT) = 0.10 sec
% :: Complex Float


   (36)  0.5752220392 3062028461 %i
                                                           Type: Complex Float
                                                    Time: 0.02 (IN) = 0.02 sec
-- (x y)^n = x^n y^n if either x > 0 or y > 0 or both x and y lie in the
-- right-half plane (Re x, Re y > 0) or n is an integer [considering principal
-- values]
(x*y)**(1/n) - x**(1/n)*y**(1/n)
```

```
              1    1 1
              -    - -
              n    n n
   (37)  (x y)  - x y
                                                     Type: Expression Integer
                           Time: 0.35 (IN) + 0.02 (EV) + 0.05 (OT) = 0.42 sec
simp(%)


               1    1 1
               -    - -
               n    n n
   (38)  [(x y)  - x y ,0,0]
                                                Type: List Expression Integer
                                                   Time: 0.62 (EV) = 0.62 sec
x : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
y : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
(x*y)**(1/n) - x**(1/n)*y**(1/n)


                2 2           2        2
           log(x y )     log(x )  log(y )
           ---------     -------  -------
               2n           2n       2n
   (41)  %e          - %e       %e
                                             Type: Complex Expression Integer
                           Time: 0.27 (IN) + 0.08 (EV) + 0.03 (OT) = 0.38 sec
simp(%)


                 2 2           2         2
            log(x y )     log(y ) + log(x )
            ---------     -----------------
                2n                  2n
   (42)  [%e          - %e                 ,0,0]
                                        Type: List Expression Complex Integer
               Time: 0.22 (IN) + 1.18 (EV) + 0.03 (OT) + 0.35 (GC) = 1.78 sec
-- Special case: (x y)^(1/n) - x^(1/n) y^(1/n) => 0 [PV] for y > 0
```

```
-- Special case: (x y)^n - x^n y^n => 0 [PV] for integer n
(x*y)**n - x**n*y**n


              n    n n
   (43)  (x y)  - x y
                                                     Type: Expression Integer
                                       Time: 0.18 (IN) + 0.02 (OT) = 0.20 sec
simp(%)


               n    n n
   (44)  [(x y)  - x y ,0,0]
                                                Type: List Expression Integer
                                                    Time: 0.62 (EV) = 0.62 sec
)clear properties x y

-- arctan(tan(z)) = z for z real if and only if z is contained in the interval
-- (-pi/2, pi/2] [considering principal values]
atan(tan(z))


   (45)  z
                                                     Type: Expression Integer
                                                   Time: 0.02 (IN) = 0.02 sec
z : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
atan(tan(z))


   (47)
                    2cos(z)sin(z)
         - atan(----------------------)
                      2         2
                sin(z)  - cos(z)  - 1
       +
                                 2cos(z)sin(z)
         atan(---------------------------------------------------)
                    4           2           2         4         2
              sin(z)  + (2cos(z)  - 1)sin(z)  + cos(z)  + cos(z)
    /
       2
   +
                                  4          2      2          4
```

```
                           4sin(z)  + 8cos(z) sin(z)  + 4cos(z)
           - log(--------------------------------------------------------)
                       4           2           2         4          2
                 sin(z)  + (2cos(z)  - 2)sin(z)  + cos(z)  + 2cos(z)  + 1
         +
                                           4
           log(--------------------------------------------------------)
                     4           2           2         4          2
               sin(z)  + (2cos(z)  - 2)sin(z)  + cos(z)  + 2cos(z)  + 1
      /
         4
    *
       %i
                                             Type: Complex Expression Integer
                           Time: 0.02 (IN) + 0.17 (EV) + 0.05 (OT) = 0.23 sec
simplify(%)


              sin(z)
   (48)  atan(------)
              cos(z)
                                                     Type: Expression Integer
                                       Time: 0.23 (EV) + 0.02 (OT) = 0.25 sec
)clear properties z

-- Special case: arctan(tan(z)) => z [PV] for -pi/2 < z < pi/2
-- The principal value of this expression is 10 - 3 pi
atan(tan(10))


   (49)  10
                                                     Type: Expression Integer
                                                   Time: 0.02 (IN) = 0.02 sec
-- The principal value of this expression is 11 - 4 pi + 30 i = -1.56637 + 30 i
atan(tan(11 + 30*%i))


   (50)  11 + 30%i
                                             Type: Expression Complex Integer
                                                   Time: 0.03 (OT) = 0.03 sec
atan(tan(11.0 + 30.0*%i))


   >> Error detected within library code:
   log 0 generated
```

```
initial (51) ->
real   47.6
user   15.8
sys    0.4

-------------------------------------------------------------------------------

Wed Jan 28 05:31:43 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- The Complex Domain ----------
-- This is a challenge problem proposed by W. Kahan: simplify the following
-- expression for complex z.  Expanding out the expression produces
-- (z^2 + 1)/(2 z) +- (z + 1)*(z - 1)/(2 z) => z or 1/z in each of its branches
z : Complex Expression Integer

                                                                   Type: Void
                                                                  Time: 0 sec
w:= (z + 1/z)/2


         2
        z  + 1
   (2)  ------
          2z
                                             Type: Complex Expression Integer
               Time: 0.33 (IN) + 0.08 (EV) + 0.22 (OT) + 0.15 (GC) = 0.78 sec
expr:= w + sqrt(w + 1)*sqrt(w - 1)
```

```
           +-----------+ +-----------+
           | 2           | 2
           |z  - 2z + 1  |z  + 2z + 1     2
        2z |-----------  |-----------  + z  + 1
          \|     2z     \|     2z
   (3)  ---------------------------------------
                           2z
                                             Type: Complex Expression Integer
               Time: 0.03 (IN) + 0.05 (EV) + 0.05 (OT) + 0.02 (GC) = 0.15 sec
)clear properties z w expr

-- ---------- Quit ----------
)quit


real   3.2
user   1.4
sys    0.2
```

12 Determining Zero Equivalence

```
-- ---------- Determining Zero Equivalence ----------
-- The following expressions are all equal to zero
sqrt(997) - (997**3)**(1/6)


   (1)  0
                                                        Type: AlgebraicNumber
               Time: 0.08 (IN) + 0.27 (EV) + 0.17 (OT) + 0.10 (GC) = 0.62 sec
sqrt(999983) - (999983**3)**(1/6)


   (2)  0
                                                        Type: AlgebraicNumber
                           Time: 0.03 (IN) + 0.08 (EV) + 0.02 (OT) = 0.13 sec
(2**(1/3) + 4**(1/3))**3 - 6*(2**(1/3) + 4**(1/3)) - 6


         3+-+3+-+2     3+-+2     3+-+    3+-+
   (3)  3\|2 \|4   + (3\|2   - 6)\|4  - 6\|2
                                                        Type: AlgebraicNumber
                           Time: 0.07 (IN) + 0.10 (EV) + 0.05 (OT) = 0.22 sec
```

```
expand(%)


   (4)  0
                                                         Type: AlgebraicNumber
                           Time: 0.50 (IN) + 0.02 (OT) + 0.02 (GC) = 0.53 sec
cos(x)**3 + cos(x)*sin(x)**2 - cos(x)


                    2         3
   (5)  cos(x)sin(x)  + cos(x)  - cos(x)
                                                      Type: Expression Integer
                           Time: 0.15 (IN) + 0.08 (EV) + 0.05 (OT) = 0.28 sec
simplify(%)


   (6)  0
                                                      Type: Expression Integer
                           Time: 0.02 (IN) + 0.07 (EV) + 0.03 (OT) = 0.12 sec
-- See Joel Moses, ``Algebraic Simplification: A Guide for the Perplexed'',
-- _Communications of the Association of Computing Machinery_, Volume 14,
-- Number 8, August 1971, 527--537.  This expression is zero if Re(x) is
-- contained in the interval ((4 n - 1)/2 pi, (4 n + 1)/2 pi) for n an integer:
-- ..., (-5/2 pi, -3/2 pi), (-pi/2, pi/2), (3/2 pi, 5/2 pi), ...
expr:= log(tan(1/2*x + %pi/4)) - asinh(tan(x))


                2x + %pi
   (7)  log(tan(--------)) - asinh(tan(x))
                    4
                                                      Type: Expression Integer
                            Time: 0.83 (IN) + 0.05 (EV) + 0.12 (OT) = 1.0 sec
complexNormalize(expr)


   (8)
     -
        log
                                +---+ 4
                     (2x + %pi)\|- 1
                     ----------------
                             4
                 ((%e                )  - 1)
              *
                  +-----------------------------------------------------+
                  |                               +---+ 4
```

```
                  |                    (2x + %pi)\|- 1
                  |                    ----------------
                  |                            4
                  |                4(%e                )
                  |- --------------------------------------------------
                  |                +---+ 8                  +---+ 4
                  |     (2x + %pi)\|- 1          (2x + %pi)\|- 1
                  |     ----------------         ----------------
                  |             4                        4
                 \|  (%e                )  - 2(%e                )  + 1
             +
                                     +---+ 4
                          (2x + %pi)\|- 1
                          ----------------
                  +---+           4             +---+
               - \|- 1 (%e                )  - \|- 1
          /
                           +---+ 4
                (2x + %pi)\|- 1
                ----------------
                        4
             (%e                )  - 1
   +
                               +---+ 2
                    (2x + %pi)\|- 1
                    ----------------
            +---+           4             +---+
         - \|- 1 (%e                )  + \|- 1
     log(--------------------------------------)
                              +---+ 2
                   (2x + %pi)\|- 1
                   ----------------
                           4
                (%e                )  + 1
                                                     Type: Expression Integer
               Time: 0.05 (IN) + 1.85 (EV) + 0.10 (OT) + 0.05 (GC) = 2.05 sec
-- Use a roundabout method---show that expr is a constant equal to zero
D(expr, x)


   (9)
                        +------------+
        2x + %pi 2      |      2             2x + %pi       2        2x + %pi
   (tan(--------)  + 1)\|tan(x)  + 1  - 2tan(--------)tan(x)  - 2tan(--------)
            4                                    4                        4
   ----------------------------------------------------------------------------
```

```
                                          +-----------+
                                2x + %pi  |      2
                           2tan(--------)\|tan(x)  + 1
                                    4
                                                     Type: Expression Integer
                                       Time: 0.05 (EV) + 0.02 (OT) = 0.07 sec
simplify(real(complexNormalize(expand(simplify(%)))))


            +------------------------+
            |     x 4        x 2             x 2
         -  |4cos(-)  - 4cos(-)  + 1  + 2cos(-)  - 1
           \|     2          2               2
   (10)  -------------------------------------------
                        x 4        x 2
                   4cos(-)  - 4cos(-)  + 1
                        2          2
                                                     Type: Expression Integer
               Time: 0.08 (IN) + 3.50 (EV) + 0.03 (OT) + 0.52 (GC) = 4.13 sec
normalize(eval(expr, x = 0))


   (11)  0
                                                     Type: Expression Integer
                           Time: 0.83 (IN) + 0.02 (EV) + 0.17 (OT) = 1.02 sec
)clear properties expr

log((2*sqrt(r) + 1)/sqrt(4*r + 4*sqrt(r) + 1))


                   +-+
                 2\|r  + 1
   (12)  log(-----------------)
              +--------------+
              |  +-+
             \|4\|r  + 4r + 1
                                                     Type: Expression Integer
                           Time: 0.10 (IN) + 0.02 (EV) + 0.03 (OT) = 0.15 sec
simplify(%)


                   +-+
                 2\|r  + 1
   (13)  log(-----------------)
              +--------------+
              |  +-+
```

```
             \|4\|r  + 4r + 1
                                                     Type: Expression Integer
                                       Time: 0.03 (EV) + 0.02 (OT) = 0.05 sec
(4*r + 4*sqrt(r) + 1)**(sqrt(r)/(2*sqrt(r) + 1)) _
   * (2*sqrt(r) + 1)**(1/(2*sqrt(r) + 1)) - 2*sqrt(r) - 1


                                                 +-+
                        1                       \|r
                    ---------                ---------
                      +-+                      +-+
            +-+     2\|r  + 1   +-+          2\|r  + 1     +-+
   (14)  (2\|r  + 1)         (4\|r  + 4r + 1)          - 2\|r  - 1
                                                     Type: Expression Integer
                           Time: 0.22 (IN) + 0.05 (EV) + 0.02 (OT) = 0.28 sec
normalize(%)


   (15)  0
                                                     Type: Expression Integer
                                                   Time: 1.02 (EV) = 1.02 sec
-- [Gradshteyn and Ryzhik 9.535(3)]
--2**(1 - z)*Gamma(z)*zeta(z)*cos(z*%pi/2) - %pi^z*zeta(1 - z)
-- ---------- Quit ----------
)quit


real   31.5
user   12.1
sys    0.6
```

13 Equations

```
-- ---------- Equations ----------
-- Manipulate an equation using a natural syntax:
-- (x = 2)/2 + (1 = 1) => x/2 + 1 = 2
(x = 2)/2 + (1 = 1)


        x + 2
   (1)  -----= 2
          2
                                   Type: Equation Fraction Polynomial Integer
               Time: 0.55 (IN) + 0.08 (EV) + 0.20 (OT) + 0.08 (GC) = 0.92 sec
```

```
-- Solve various nonlinear equations---this cubic polynomial has all real roots
radicalSolve(3*x**3 - 18*x**2 + 33*x - 19 = 0, x)


   (2)
                        +----------+2                 +----------+
                        | +---+                       | +---+
            +---+       |\|- 3  + 1       +---+       |\|- 3  + 1
       (- 3\|- 3  + 3)  |----------  + (6\|- 3  + 6)  |---------- - 2
                       3|    +---+                   3|    +---+
                       \|  6\|- 3                    \|  6\|- 3
   [x= --------------------------------------------------------------,
                                        +----------+
                                        | +---+
                            +---+       |\|- 3  + 1
                         (3\|- 3  + 3)  |----------
                                       3|    +---+
                                       \|  6\|- 3
                        +----------+2                 +----------+
                        | +---+                       | +---+
            +---+       |\|- 3  + 1       +---+       |\|- 3  + 1
       (- 3\|- 3  - 3)  |----------  + (6\|- 3  - 6)  |---------- + 2
                       3|    +---+                   3|    +---+
                       \|  6\|- 3                    \|  6\|- 3
    x= --------------------------------------------------------------,
                                        +----------+
                                        | +---+
                            +---+       |\|- 3  + 1
                         (3\|- 3  - 3)  |----------
                                       3|    +---+
                                       \|  6\|- 3
          +----------+2     +----------+
          | +---+           | +---+
          |\|- 3  + 1       |\|- 3  + 1
       3  |----------  + 6  |---------- + 1
         3|    +---+       3|    +---+
         \|  6\|- 3        \|  6\|- 3
    x= ------------------------------------]
                     +----------+
                     | +---+
                     |\|- 3  + 1
                  3  |----------
                    3|    +---+
                    \|  6\|- 3
                                       Type: List Equation Expression Integer
               Time: 0.10 (IN) + 0.55 (EV) + 0.15 (OT) + 0.08 (GC) = 0.88 sec
```

```
map(e +-> lhs(e) = simplify(complexForm(rhs(e))), %)


   (3)
          +-+    %pi        %pi      +-+       +-+    %pi        %pi      +-+
       - \|3 sin(---) - cos(---) + 2\|3       \|3 sin(---) - cos(---) + 2\|3
                  18         18                         18         18
   [x= ---------------------------------, x= --------------------------------,
                      +-+                                    +-+
                     \|3                                    \|3
            %pi      +-+
       2cos(---) + 2\|3
             18
    x= -----------------]
               +-+
              \|3
                                        Type: List Equation Expression Integer
                Time: 0.25 (IN) + 1.62 (EV) + 0.12 (OT) + 0.02 (GC) = 2.0 sec
-- Some simple seeming problems can have messy answers:
-- x = {  [sqrt(5) - 1]/4 +/- 5^(1/4) sqrt(sqrt(5) + 1)/[2 sqrt(2)] i,
--      - [sqrt(5) + 1]/4 +/- 5^(1/4) sqrt(sqrt(5) - 1)/[2 sqrt(2)] i}
eqn:= x**4 + x**3 + x**2 + x + 1 = 0


         4    3    2
   (4)  x  + x  + x  + x + 1= 0
                                             Type: Equation Polynomial Integer
                                        Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
radicalSolve(eqn, x)


   (5)
   [
     x =
           -
                2
             *
                ROOT
                                +-------------+2      +-------------+
                                |    +---+            |    +---+
                                |    |  5             |    |  5
                                |45  |- - - 25        |45  |- - - 25
                                |   \|  3             |   \|  3
                          (- 4 3|-------------  - 10 3|------------- - 40)
                               \|      2             \|      2
                       *
```

```
        +-------------------------------------------+
        |   +-------------+2     +-------------+
        |   |    +---+           |    +---+
        |   |    |  5            |    |  5
        |   |45  |- - - 25       |45  |- - - 25
        |   |   \|  3            |   \|  3
        |4 3|-------------  - 5 3|------------- + 40
        |  \|      2            \|      2
        |-------------------------------------------
        |                 +-------------+
        |                 |    +---+
        |                 |    |  5
        |                 |45  |- - - 25
        |                 |   \|  3
        |             12 3|-------------
       \|                \|      2
   +
           +-------------+
           |    +---+
           |    |  5
           |45  |- - - 25
           |   \|  3
     - 15 3|-------------
          \|      2
/
         +-------------+
         |    +---+
         |    |  5
         |45  |- - - 25
         |   \|  3
     12 3|-------------
        \|      2
  *
      +-------------------------------------------+
      |   +-------------+2     +-------------+
      |   |    +---+           |    +---+
      |   |    |  5            |    |  5
      |   |45  |- - - 25       |45  |- - - 25
      |   |   \|  3            |   \|  3
      |4 3|-------------  - 5 3|------------- + 40
      |  \|      2            \|      2
      |-------------------------------------------
      |                 +-------------+
      |                 |    +---+
      |                 |    |  5
      |                 |45  |- - - 25
```

```
                   |                 |   \|  3
                   |             12 3|-------------
                  \|                \|      2
    +
        +-------------------------------------------+
        |   +-------------+2     +-------------+
        |   |    +---+           |    +---+
        |   |    |  5            |    |  5
        |   |45  |- - - 25       |45  |- - - 25
        |   |   \|  3            |   \|  3
        |4 3|-------------  - 5 3|------------- + 40
        |  \|      2            \|      2
      2 |-------------------------------------------  - 1
        |                 +-------------+
        |                 |    +---+
        |                 |    |  5
        |                 |45  |- - - 25
        |                 |   \|  3
        |             12 3|-------------
       \|                \|      2
 /
    4
,

x =
        2
     *
        ROOT
                       +-------------+2      +-------------+
                       |    +---+            |    +---+
                       |    |  5             |    |  5
                       |45  |- - - 25        |45  |- - - 25
                       |   \|  3             |   \|  3
                 (- 4 3|-------------  - 10 3|------------- - 40)
                      \|      2             \|      2
              *
                  +-------------------------------------------+
                  |   +-------------+2     +-------------+
                  |   |    +---+           |    +---+
                  |   |    |  5            |    |  5
                  |   |45  |- - - 25       |45  |- - - 25
                  |   |   \|  3            |   \|  3
                  |4 3|-------------  - 5 3|------------- + 40
                  |  \|      2            \|      2
                  |-------------------------------------------
                  |                 +-------------+
```

```
              |                 |    +---+
              |                 |    |  5
              |                 |45  |- - - 25
              |                 |   \|  3
              |             12 3|-------------
             \|                \|      2
         +
                 +-------------+
                 |    +---+
                 |    |  5
                 |45  |- - - 25
                 |   \|  3
           - 15 3|-------------
                \|      2
      /
               +-------------+
               |    +---+
               |    |  5
               |45  |- - - 25
               |   \|  3
           12 3|-------------
              \|      2
        *
            +--------------------------------------------+
            |   +-------------+2     +-------------+
            |   |    +---+          |    +---+
            |   |    |  5           |    |  5
            |   |45  |- - - 25       |45  |- - - 25
            |   |   \|  3           |   \|  3
            |4 3|-------------  - 5 3|------------- + 40
            |  \|      2            \|      2
            |--------------------------------------------
            |                 +-------------+
            |                 |    +---+
            |                 |    |  5
            |                 |45  |- - - 25
            |                 |   \|  3
            |             12 3|-------------
           \|                \|      2
+
    +--------------------------------------------+
    |   +-------------+2     +-------------+
    |   |    +---+          |    +---+
    |   |    |  5           |    |  5
    |   |45  |- - - 25       |45  |- - - 25
    |   |   \|  3           |   \|  3
```

```
        |4 3|-------------  - 5 3|------------- + 40
        |  \|      2            \|      2
      2 |-------------------------------------------  - 1
        |                 +-------------+
        |                 |    +---+
        |                 |    |  5
        |                 |45  |- - - 25
        |                 |   \|  3
        |             12 3|-------------
       \|                \|      2
 /
    4
,

x =
      -
           2
        *
           ROOT
                          +-------------+2      +-------------+
                          |    +---+            |    +---+
                          |    |  5             |    |  5
                          |45  |- - - 25        |45  |- - - 25
                          |   \|  3             |   \|  3
                    (- 4 3|-------------  - 10 3|------------- - 40)
                         \|      2             \|      2
                 *
                     +--------------------------------------------+
                     |   +-------------+2     +-------------+
                     |   |    +---+           |    +---+
                     |   |    |  5            |    |  5
                     |   |45  |- - - 25       |45  |- - - 25
                     |   |   \|  3            |   \|  3
                     |4 3|-------------  - 5 3|------------- + 40
                     |  \|      2            \|      2
                     |--------------------------------------------
                     |                 +-------------+
                     |                 |    +---+
                     |                 |    |  5
                     |                 |45  |- - - 25
                     |                 |   \|  3
                     |             12 3|-------------
                    \|                \|      2
                +
                      +-------------+
                      |    +---+
```

```
                  |    |  5
                  |45  |- - - 25
                  |   \|  3
              15 3|-------------
                 \|      2
         /
                  +-------------+
                  |    +---+
                  |    |  5
                  |45  |- - - 25
                  |   \|  3
              12 3|-------------
                 \|      2
           *
               +-------------------------------------------+
               |   +-------------+2     +-------------+
               |   |    +---+           |    +---+
               |   |    |  5            |    |  5
               |   |45  |- - - 25       |45  |- - - 25
               |   |   \|  3            |   \|  3
               |4 3|-------------  - 5 3|------------- + 40
               |  \|      2            \|      2
               |-------------------------------------------
               |                 +-------------+
               |                 |    +---+
               |                 |    |  5
               |                 |45  |- - - 25
               |                 |   \|  3
               |             12 3|-------------
              \|                \|      2
+
      +-------------------------------------------+
      |   +-------------+2     +-------------+
      |   |    +---+           |    +---+
      |   |    |  5            |    |  5
      |   |45  |- - - 25       |45  |- - - 25
      |   |   \|  3            |   \|  3
      |4 3|-------------  - 5 3|------------- + 40
      |  \|      2            \|      2
  - 2 |-------------------------------------------  - 1
      |                 +-------------+
      |                 |    +---+
      |                 |    |  5
      |                 |45  |- - - 25
      |                 |   \|  3
      |             12 3|-------------
```

```
         \|               \|      2
 /
    4
,

x =
        2
     *
        ROOT
                      +-------------+2     +-------------+
                      |    +---+           |    +---+
                      |    |  5            |    |  5
                      |45  |- - - 25       |45  |- - - 25
                      |   \|  3            |   \|  3
                (- 4 3|------------- - 10 3|------------- - 40)
                     \|      2            \|      2
              *
                 +------------------------------------------+
                 |   +-------------+2     +-------------+
                 |   |    +---+           |    +---+
                 |   |    |  5            |    |  5
                 |   |45  |- - - 25       |45  |- - - 25
                 |   |   \|  3            |   \|  3
                 |4 3|------------- - 5 3|------------- + 40
                 |  \|      2            \|      2
                 |------------------------------------------
                 |                +-------------+
                 |                |    +---+
                 |                |    |  5
                 |                |45  |- - - 25
                 |                |   \|  3
                 |            12 3|-------------
                \|               \|      2
             +
                  +-------------+
                  |    +---+
                  |    |  5
                  |45  |- - - 25
                  |   \|  3
              15 3|-------------
                 \|      2
          /
                  +-------------+
                  |    +---+
                  |    |  5
                  |45  |- - - 25
```

```
                        |   \|  3
                    12 3|-------------
                       \|      2
                 *
                     +-------------------------------------------+
                     |   +-------------+2     +-------------+
                     |   |    +---+           |    +---+
                     |   |    |  5            |    |  5
                     |   |45  |- - - 25       |45  |- - - 25
                     |   |   \|  3            |   \|  3
                     |4 3|-------------  - 5 3|------------- + 40
                     |  \|      2            \|      2
                     |-------------------------------------------
                     |                 +-------------+
                     |                 |    +---+
                     |                 |    |  5
                     |                 |45  |- - - 25
                     |                 |   \|  3
                     |             12 3|-------------
                    \|                \|      2
         +
               +-------------------------------------------+
               |   +-------------+2     +-------------+
               |   |    +---+           |    +---+
               |   |    |  5            |    |  5
               |   |45  |- - - 25       |45  |- - - 25
               |   |   \|  3            |   \|  3
               |4 3|-------------  - 5 3|------------- + 40
               |  \|      2            \|      2
           - 2 |-------------------------------------------  - 1
               |                 +-------------+
               |                 |    +---+
               |                 |    |  5
               |                 |45  |- - - 25
               |                 |   \|  3
               |             12 3|-------------
              \|                \|      2
      /
         4
     ]
                                       Type: List Equation Expression Integer
                                       Time: 0.05 (EV) + 0.52 (OT) = 0.57 sec
-- Check one of the answers
eval(eqn, %.1)
```

```
   (6)  0= 0
                                            Type: Equation Expression Integer
                           Time: 0.47 (IN) + 0.58 (EV) + 0.07 (OT) = 1.12 sec
)clear properties eqn

-- x = {2^(1/3) +- sqrt(3), +- sqrt(3) - 1/2^(2/3) +- i sqrt(3)/2^(2/3)}
--     [Mohamed Omar Rayes]
solve(x**6 - 9*x**4 - 4*x**3 + 27*x**2 - 36*x - 23 = 0, x)


          6     4     3      2
   (7)  [x  - 9x  - 4x  + 27x  - 36x - 23= 0]
                              Type: List Equation Fraction Polynomial Integer
                           Time: 0.23 (IN) + 0.08 (EV) + 0.13 (OT) = 0.45 sec
-- x = {1, e^(+- 2 pi i/7), e^(+- 4 pi i/7), e^(+- 6 pi i/7)}
solve(x**7 - 1 = 0, x)


               6    5    4    3    2
   (8)  [x= 1,x  + x  + x  + x  + x  + x + 1= 0]
                              Type: List Equation Fraction Polynomial Integer
                                                   Time: 0.05 (EV) = 0.05 sec
-- x = 1 +- sqrt(+-sqrt(+-4 sqrt(3) - 3) - 3)/sqrt(2)   [Richard Liska]
solve(x**8 - 8*x**7 + 34*x**6 - 92*x**5 + 175*x**4 - 236*x**3 + 226*x**2 _
      - 140*x + 46 = 0, x)


          8     7      6      5       4       3       2
   (9)  [x  - 8x  + 34x  - 92x  + 175x  - 236x  + 226x  - 140x + 46= 0]
                              Type: List Equation Fraction Polynomial Integer
                           Time: 0.05 (IN) + 0.02 (EV) + 0.05 (OT) = 0.12 sec
-- The following equations have an infinite number of solutions (let n be an
-- arbitrary integer):
-- x = {log(sqrt(z) - 1), log(sqrt(z) + 1) + i pi} [+ n 2 pi i, + n 2 pi i]
%e**(2*x) + 2*%e**x + 1 = z


           2x      x
   (10)  %e   + 2%e  + 1= z
                                            Type: Equation Expression Integer
                           Time: 0.20 (IN) + 0.07 (EV) + 0.05 (OT) = 0.32 sec
solve(%, x)


                  +-+                +-+
   (11)  [x= log(\|z  - 1),x= log(- \|z  - 1)]
```

```
                                       Type: List Equation Expression Integer
               Time: 0.32 (IN) + 0.85 (EV) + 0.12 (OT) + 0.38 (GC) = 1.67 sec
-- x = (1 +- sqrt(9 - 8 n pi i))/2.  Real solutions correspond to n = 0 =>
-- x = {-1, 2}
solve(exp(2 - x**2) = exp(-x), x)


   (12)  []
                                       Type: List Equation Expression Integer
                                       Time: 0.17 (EV) + 0.02 (OT) = 0.18 sec
-- x = -W[n](-1) [e.g., -W[0](-1) = 0.31813 - 1.33724 i] where W[n](x) is the
-- nth branch of Lambert's W function
solve(exp(x) = x, x)


   (13)  []
                                       Type: List Equation Expression Integer
                           Time: 0.07 (IN) + 0.07 (EV) + 0.02 (OT) = 0.15 sec
-- x = {-1, 1}
solve(x**x = x, x)


   (14)  []
                                       Type: List Equation Expression Integer
                           Time: 0.03 (IN) + 0.08 (EV) + 0.02 (OT) = 0.13 sec
-- This equation is already factored and so *should* be easy to solve:
-- x = {-1, 2*{+-arcsinh(1) i + n pi}, 3*{pi/6 + n pi/3}}
(x + 1) * (sin(x)**2 + 1)**2 * cos(3*x)**3 = 0


                       3      4                  3      2                 3
   (15)  (x + 1)cos(3x) sin(x)  + (2x + 2)cos(3x) sin(x)  + (x + 1)cos(3x) = 0
                                            Type: Equation Expression Integer
                           Time: 0.05 (IN) + 0.03 (EV) + 0.03 (OT) = 0.12 sec
solve(%, x)


   (16)  []
                                       Type: List Equation Expression Integer
                                       Time: 2.88 (EV) + 0.02 (GC) = 2.90 sec
-- x = pi/4 [+ n pi]
solve(sin(x) = cos(x), x)


             %pi
   (17)  [x= ---]
```

```
              4
                                       Type: List Equation Expression Integer
                                       Time: 0.02 (IN) + 0.15 (EV) = 0.17 sec
solve(tan(x) = 1, x)


             %pi
   (18)  [x= ---]
              4
                                       Type: List Equation Expression Integer
                                                   Time: 0.03 (EV) = 0.03 sec
-- x = {pi/6, 5 pi/6} [ + n 2 pi, + n 2 pi ]
solve(sin(x) = 1/2, x)


                  1
   (19)  [x= asin(-)]
                  2
                                       Type: List Equation Expression Integer
                           Time: 0.08 (IN) + 0.02 (EV) + 0.02 (OT) = 0.12 sec
map(e +-> lhs(e) = normalize(rhs(e)), %)


             %pi
   (20)  [x= ---]
              6
                                       Type: List Equation Expression Integer
                                       Time: 0.05 (IN) + 0.02 (EV) = 0.07 sec
-- x = {0, 0} [+ n pi, + n 2 pi]
solve(sin(x) = tan(x), x)


   (21)  [x= 0]
                                       Type: List Equation Expression Integer
                                                   Time: 0.15 (EV) = 0.15 sec
-- x = {0, 0, 0}
solve(asin(x) = atan(x), x)


   (22)  [x= 0]
                                       Type: List Equation Expression Integer
                                       Time: 0.68 (EV) + 0.02 (OT) = 0.70 sec
-- x = sqrt[(sqrt(5) - 1)/2]
solve(acos(x) = atan(x), x)
```

```
   (23)
          +-----------+     +-----------+       +---------+     +---------+
          |    +-+          |    +-+            |  +-+          |  +-+
         \|- 2\|5  - 2     \|- 2\|5  - 2       \|2\|5  - 2     \|2\|5  - 2
   [x= - --------------,x= --------------,x= - ------------,x= ------------]
                2                 2                  2               2
                                       Type: List Equation Expression Integer
               Time: 0.02 (IN) + 1.08 (EV) + 0.03 (OT) + 0.38 (GC) = 1.52 sec
-- x = 2
solve((x - 2)/x**(1/3) = 0, x)


   (24)  [x= 2]
                                       Type: List Equation Expression Integer
                           Time: 0.10 (IN) + 0.03 (EV) + 0.03 (OT) = 0.17 sec
-- This equation has no solutions
solve(sqrt(x**2 + 1) = x - 2, x)


             3
   (25)  [x= -]
             4
                                       Type: List Equation Expression Integer
                                       Time: 0.05 (IN) + 0.05 (EV) = 0.10 sec
-- x = 1
solve(x + sqrt(x) = 2, x)


   (26)  [x= 4,x= 1]
                                       Type: List Equation Expression Integer
                                       Time: 0.03 (IN) + 0.03 (EV) = 0.07 sec
-- x = 1/16
solve(2*sqrt(x) + 3*x**(1/4) - 2 = 0, x)


                           +---+             +---+
                    1    3\|- 7  - 31    - 3\|- 7  - 31
   (27)  [x= 16,x= --,x= ------------,x= --------------]
                   16         32               32
                                       Type: List Equation Expression Integer
                           Time: 0.03 (IN) + 0.18 (EV) + 0.03 (OT) = 0.25 sec
-- x = {sqrt[(sqrt(5) - 1)/2], -i sqrt[(sqrt(5) + 1)/2]}
solve(x = 1/sqrt(1 + x**2), x)


   (28)
```

```
        +---------+       +---------+     +-----------+       +-----------+
        |  +-+            |  +-+          |    +-+            |    +-+
       \|2\|5  - 2       \|2\|5  - 2     \|- 2\|5  - 2       \|- 2\|5  - 2
   [x= ------------,x= - ------------,x= --------------,x= - --------------]
             2                 2                2                   2
                                       Type: List Equation Expression Integer
                           Time: 0.07 (IN) + 0.17 (EV) + 0.03 (OT) = 0.27 sec
-- This problem is from a computational biology talk => 1 - log_2[m (m - 1)]
solve(binomial(m, 2)*2**k = 1, k)


                    2
             log(------)
                  2
                 m  - m
   (29)  [k= -----------]
                log(2)
                                       Type: List Equation Expression Integer
                                       Time: 0.13 (IN) + 0.12 (EV) = 0.25 sec
-- x = log(c/a) / log(b/d) for a, b, c, d != 0 and b, d != 1   [Bill Pletsch]
solve(a*b**x = c*d**x, x)


   (30)  []
                                       Type: List Equation Expression Integer
                           Time: 0.15 (IN) + 0.18 (EV) + 0.02 (OT) = 0.35 sec
-- x = {1, e^4}
solve(sqrt(log(x)) = log(sqrt(x)), x)


   (31)  [x= 0,x= 1]
                                       Type: List Equation Expression Integer
                           Time: 0.02 (IN) + 0.10 (EV) + 0.03 (OT) = 0.15 sec
-- Recursive use of inverses, including multiple branches of rational
-- fractional powers   [Richard Liska]
-- => x = +-(b + sin(1 + cos(1/e^2)))^(3/2)
solve(log(acos(asin(x**(2/3) - b) - 1)) + 2 = 0, x)


   (32)
                                 +----------------------+
                   1             |         1
   [x= (- sin(cos(---) + 1) - b) |sin(cos(---) + 1) + b ,
                    2            |          2
                  %e            \|        %e
                               +----------------------+
```

```
                 1             |         1
    x= (sin(cos(---) + 1) + b) |sin(cos(---) + 1) + b ]
                  2            |          2
                %e            \|        %e
                                       Type: List Equation Expression Integer
               Time: 0.05 (IN) + 0.25 (EV) + 0.03 (OT) + 0.02 (GC) = 0.35 sec
-- x = {-0.784966, -0.016291, 0.802557}  From Metha Kamminga-van Hulsen,
-- ``Hoisting the Sails and Casting Off with Maple'', _Computer Algebra
-- Nederland Nieuwsbrief_, Number 13, December 1994, ISSN 1380-1260, 27--40.
eqn:= 5*x + exp((x - 5)/2) = 8*x**3


           x - 5
           -----
             2           3
   (33)  %e      + 5x= 8x
                                            Type: Equation Expression Integer
                           Time: 0.50 (IN) + 0.02 (EV) + 0.02 (OT) = 0.53 sec
solve(eqn, x)


   (34)  []
                                       Type: List Equation Expression Integer
                                                   Time: 0.08 (EV) = 0.08 sec
--root_by_bisection(eqn, x, -1,  -0.5)
--root_by_bisection(eqn, x, -0.5, 0.5)
--root_by_bisection(eqn, x,  0.5, 1)
)clear properties eqn

-- x = {-1, 3}
solve(abs(x - 1) = 2, x)


   (35)  []
                                       Type: List Equation Expression Integer
                                       Time: 0.03 (IN) + 0.03 (EV) = 0.07 sec
-- x = {-1, -7}
solve(abs(2*x + 5) = abs(x - 2), x)


   (36)  []
                                       Type: List Equation Expression Integer
                           Time: 0.02 (IN) + 0.05 (EV) + 0.02 (OT) = 0.08 sec
-- x = +-3/2
solve(1 - abs(x) = max(-x - 2, x - 2), x)
```

```
   (37)  []
                                         Type: List Equation Expression Integer
                           Time: 0.03 (IN) + 0.03 (EV) + 0.02 (OT) = 0.08 sec
-- x = {-1, 3}
solve(max(2 - x**2, x) = max(-x, x**3/9), x)


   (38)  [x= 3,x= 0,x= - 3]
                                Type: List Equation Fraction Polynomial Integer
                           Time: 0.58 (IN) + 0.02 (EV) + 0.12 (OT) = 0.72 sec
-- x = {+-3, -3 [1 + sqrt(3) sin t + cos t]} = {+-3, -1.554894}
-- where t = (arctan[sqrt(5)/2] - pi)/3.  The third answer is the root of
-- x^3 + 9 x^2 - 18 = 0 in the interval (-2, -1).
solve(max(2 - x**2, x) = x**3/9, x)


   (39)  [x= 3,x= 0,x= - 3]
                                Type: List Equation Fraction Polynomial Integer
                                         Time: 0.08 (IN) + 0.03 (OT) = 0.12 sec
-- z = 2 + 3 i
z : Complex Expression Integer

                                                                     Type: Void
                                                                    Time: 0 sec
(1 + %i)*z + (2 - %i)*conjugate(z) = -3*%i


   (41)  3z= - 3%i
                                      Type: Equation Complex Expression Integer
                           Time: 0.47 (IN) + 0.02 (EV) + 0.05 (OT) = 0.53 sec
)clear properties z

(1 + %i)*(x + %i*y) + (2 - %i)*conjugate(x + %i*y) = -3*%i

   There are 4 exposed and 1 unexposed library operations named
      conjugate having 1 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op conjugate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      conjugate with argument type(s)
                         Polynomial Complex Integer
```

```
(1 + %i)*(x + %i*y) + (2 - %i)*(x - %i*y) = -3*%i


   (42)  (- 2 - %i)y + 3x= - 3%i
                                    Type: Equation Polynomial Complex Integer
                                       Time: 0.07 (IN) + 0.03 (OT) = 0.10 sec
solve(%, [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                     Equation Polynomial Complex Integer
                       List OrderedVariableList [x,y]

-- => {f^(-1)(1), f^(-1)(-2)} assuming f is invertible
f:= operator('f);



                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
solve(f(x)**2 + f(x) - 2 = 0, x)


   (44)  []
                                       Type: List Equation Expression Integer
                           Time: 0.32 (IN) + 0.02 (EV) + 0.07 (OT) = 0.40 sec
)clear properties f

-- Solve a 3 x 3 system of linear equations
eqn1:=   x +   y +   z =  6


   (45)  z + y + x= 6
                                            Type: Equation Polynomial Integer
                                                   Time: 0.08 (IN) = 0.08 sec
eqn2:= 2*x +   y + 2*z = 10
```

```
   (46)  2z + y + 2x= 10
                                            Type: Equation Polynomial Integer
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
eqn3:=   x + 3*y +   z = 10


   (47)  z + 3y + x= 10
                                            Type: Equation Polynomial Integer
                                                   Time: 0.02 (OT) = 0.02 sec
-- Note that the solution is parametric: x = 4 - z, y = 2
solve([eqn1, eqn2, eqn3], [x, y, z])


   (48)  [[x= - %DW + 4,y= 2,z= %DW]]
                         Type: List List Equation Fraction Polynomial Integer
                           Time: 0.68 (IN) + 0.05 (EV) + 0.08 (OT) = 0.82 sec
-- A linear system arising from the computation of a truncated power series
-- solution to a differential equation.  There are 189 equations to be solved
-- for 49 unknowns.  42 of the equations are repeats of other equations; many
-- others are trivial.  Solving this system directly by Gaussian elimination
-- is *not* a good idea.  Solving the easy equations first is probably a better
-- method.  The solution is actually rather simple.   [Stanly Steinberg]
-- => k1 = ... = k22 = k24 = k25 = k27 = ... = k30 = k32 = k33 = k35 = ...
--    = k38 = k40 = k41 = k44 = ... = k49 = 0, k23 = k31 = k39,
--    k34 = b/a k26, k42 = c/a k26, {k23, k26, k43} are arbitrary
eqns:= [
 -b*k8/a+c*k8/a = 0, -b*k11/a+c*k11/a = 0, -b*k10/a+c*k10/a+k2 = 0, _
 -k3-b*k9/a+c*k9/a = 0, -b*k14/a+c*k14/a = 0, -b*k15/a+c*k15/a = 0, _
 -b*k18/a+c*k18/a-k2 = 0, -b*k17/a+c*k17/a = 0, -b*k16/a+c*k16/a+k4 = 0, _
 -b*k13/a+c*k13/a-b*k21/a+c*k21/a+b*k5/a-c*k5/a = 0, b*k44/a-c*k44/a = 0, _
 -b*k45/a+c*k45/a = 0, -b*k20/a+c*k20/a = 0, -b*k44/a+c*k44/a = 0, _
  b*k46/a-c*k46/a = 0, b**2*k47/a**2-2*b*c*k47/a**2+c**2*k47/a**2 = 0, k3 = 0, _
 -k4 = 0, -b*k12/a+c*k12/a-a*k6/b+c*k6/b = 0, _
 -b*k19/a+c*k19/a+a*k7/c-b*k7/c = 0, b*k45/a-c*k45/a = 0, _
 -b*k46/a+c*k46/a = 0, -k48+c*k48/a+c*k48/b-c**2*k48/(a*b) = 0, _
 -k49+b*k49/a+b*k49/c-b**2*k49/(a*c) = 0, a*k1/b-c*k1/b = 0, _
  a*k4/b-c*k4/b = 0, a*k3/b-c*k3/b+k9 = 0, -k10+a*k2/b-c*k2/b = 0, _
  a*k7/b-c*k7/b = 0, -k9 = 0, k11 = 0, b*k12/a-c*k12/a+a*k6/b-c*k6/b = 0, _
  a*k15/b-c*k15/b = 0, k10+a*k18/b-c*k18/b = 0, -k11+a*k17/b-c*k17/b = 0, _
  a*k16/b-c*k16/b = 0, -a*k13/b+c*k13/b+a*k21/b-c*k21/b+a*k5/b-c*k5/b = 0, _
 -a*k44/b+c*k44/b = 0, a*k45/b-c*k45/b = 0, _
  a*k14/c-b*k14/c+a*k20/b-c*k20/b = 0, a*k44/b-c*k44/b = 0, _
 -a*k46/b+c*k46/b = 0, -k47+c*k47/a+c*k47/b-c**2*k47/(a*b) = 0, _
  a*k19/b-c*k19/b = 0, -a*k45/b+c*k45/b = 0, a*k46/b-c*k46/b = 0, _
  a**2*k48/b**2-2*a*c*k48/b**2+c**2*k48/b**2 = 0, _
 -k49+a*k49/b+a*k49/c-a**2*k49/(b*c) = 0, k16 = 0, -k17 = 0, _
```

```
-a*k1/c+b*k1/c = 0, -k16-a*k4/c+b*k4/c = 0, -a*k3/c+b*k3/c = 0, _
 k18-a*k2/c+b*k2/c = 0, b*k19/a-c*k19/a-a*k7/c+b*k7/c = 0, _
-a*k6/c+b*k6/c = 0, -a*k8/c+b*k8/c = 0, -a*k11/c+b*k11/c+k17 = 0, _
-a*k10/c+b*k10/c-k18 = 0, -a*k9/c+b*k9/c = 0, _
-a*k14/c+b*k14/c-a*k20/b+c*k20/b = 0, _
-a*k13/c+b*k13/c+a*k21/c-b*k21/c-a*k5/c+b*k5/c = 0, a*k44/c-b*k44/c = 0, _
-a*k45/c+b*k45/c = 0, -a*k44/c+b*k44/c = 0, a*k46/c-b*k46/c = 0, _
-k47+b*k47/a+b*k47/c-b**2*k47/(a*c) = 0, -a*k12/c+b*k12/c = 0, _
 a*k45/c-b*k45/c = 0, -a*k46/c+b*k46/c = 0, _
-k48+a*k48/b+a*k48/c-a**2*k48/(b*c) = 0, _
 a**2*k49/c**2-2*a*b*k49/c**2+b**2*k49/c**2 = 0, k8 = 0, k11 = 0, -k15 = 0, _
 k10-k18 = 0, -k17 = 0, k9 = 0, -k16 = 0, -k29 = 0, k14-k32 = 0, _
-k21+k23-k31 = 0, -k24-k30 = 0, -k35 = 0, k44 = 0, -k45 = 0, k36 = 0, _
 k13-k23+k39 = 0, -k20+k38 = 0, k25+k37 = 0, b*k26/a-c*k26/a-k34+k42 = 0, _
-2*k44 = 0, k45 = 0, k46 = 0, b*k47/a-c*k47/a = 0, k41 = 0, k44 = 0, _
-k46 = 0, -b*k47/a+c*k47/a = 0, k12+k24 = 0, -k19-k25 = 0, _
-a*k27/b+c*k27/b-k33 = 0, k45 = 0, -k46 = 0, -a*k48/b+c*k48/b = 0, _
 a*k28/c-b*k28/c+k40 = 0, -k45 = 0, k46 = 0, a*k48/b-c*k48/b = 0, _
 a*k49/c-b*k49/c = 0, -a*k49/c+b*k49/c = 0, -k1 = 0, -k4 = 0, -k3 = 0, _
 k15 = 0, k18-k2 = 0, k17 = 0, k16 = 0, k22 = 0, k25-k7 = 0, _
 k24+k30 = 0, k21+k23-k31 = 0, k28 = 0, -k44 = 0, k45 = 0, -k30-k6 = 0, _
 k20+k32 = 0, k27+b*k33/a-c*k33/a = 0, k44 = 0, -k46 = 0, _
-b*k47/a+c*k47/a = 0, -k36 = 0, k31-k39-k5 = 0, -k32-k38 = 0, _
 k19-k37 = 0, k26-a*k34/b+c*k34/b-k42 = 0, k44 = 0, -2*k45 = 0, k46 = 0, _
 a*k48/b-c*k48/b = 0, a*k35/c-b*k35/c-k41 = 0, -k44 = 0, k46 = 0, _
 b*k47/a-c*k47/a = 0, -a*k49/c+b*k49/c = 0, -k40 = 0, k45 = 0, -k46 = 0, _
-a*k48/b+c*k48/b = 0, a*k49/c-b*k49/c = 0, k1 = 0, k4 = 0, k3 = 0, _
-k8 = 0, -k11 = 0, -k10+k2 = 0, -k9 = 0, k37+k7 = 0, -k14-k38 = 0, _
-k22 = 0, -k25-k37 = 0, -k24+k6 = 0, -k13-k23+k39 = 0, _
-k28+b*k40/a-c*k40/a = 0, k44 = 0, -k45 = 0, -k27 = 0, -k44 = 0, _
 k46 = 0, b*k47/a-c*k47/a = 0, k29 = 0, k32+k38 = 0, k31-k39+k5 = 0, _
-k12+k30 = 0, k35-a*k41/b+c*k41/b = 0, -k44 = 0, k45 = 0, _
-k26+k34+a*k42/c-b*k42/c = 0, k44 = 0, k45 = 0, -2*k46 = 0, _
-b*k47/a+c*k47/a = 0, -a*k48/b+c*k48/b = 0, a*k49/c-b*k49/c = 0, k33 = 0, _
-k45 = 0, k46 = 0, a*k48/b-c*k48/b = 0, -a*k49/c+b*k49/c = 0 _
];


                              Type: List Equation Fraction Polynomial Integer
               Time: 6.35 (IN) + 0.72 (EV) + 1.33 (OT) + 0.87 (GC) = 9.26 sec
vars:= [k1, k2, k3, k4, k5, k6, k7, k8, k9, k10, k11, k12, k13, k14, k15, k16, _
        k17, k18, k19, k20, k21, k22, k23, k24, k25, k26, k27, k28, k29, k30, _
        k31, k32, k33, k34, k35, k36, k37, k38, k39, k40, k41, k42, k43, k44, _
        k45, k46, k47, k48, k49];
```

```
Type: List OrderedVariableList [k1,k2,k3,k4,k5,k6,k7,k8,k9,k10,k11,k12,k13,k14,k15,k16,k17,
                                                   Time: 0.05 (IN) = 0.05 sec
solve(eqns, vars)


   (51)
   [
     [k1= 0, k2= 0, k3= 0, k4= 0, k5= 0, k6= 0, k7= 0, k8= 0, k9= 0, k10= 0,
      k11= 0, k12= 0, k13= 0, k14= 0, k15= 0, k16= 0, k17= 0, k18= 0, k19= 0,
                                                             %DY a
      k20= 0, k21= 0, k22= 0, k23= %DX, k24= 0, k25= 0, k26= -----, k27= 0,
                                                               c
                                                             %DY b
      k28= 0, k29= 0, k30= 0, k31= %DX, k32= 0, k33= 0, k34= -----, k35= 0,
                                                               c
      k36= 0, k37= 0, k38= 0, k39= %DX, k40= 0, k41= 0, k42= %DY, k43= %DZ,
      k44= 0, k45= 0, k46= 0, k47= 0, k48= 0, k49= 0]
     ]
                         Type: List List Equation Fraction Polynomial Integer
                           Time: 0.69 (IN) + 1.42 (EV) + 0.15 (OT) = 2.25 sec
)clear properties eqns vars

-- Solve a 3 x 3 system of nonlinear equations
eqn1:= x**2*y + 3*y*z - 4 = 0


                 2
   (52)  3y z + x y - 4= 0
                                            Type: Equation Polynomial Integer
                                                   Time: 0.03 (OT) = 0.03 sec
eqn2:= -3*x**2*z + 2*y**2 + 1 = 0


             2      2
   (53)  - 3x z + 2y  + 1= 0
                                            Type: Equation Polynomial Integer
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
eqn3:= 2*y*z**2 - z**2 - 1 = 0


                  2
   (54)  (2y - 1)z  - 1= 0
                                            Type: Equation Polynomial Integer
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
-- Solving this by hand would be a nightmare
solve([eqn1, eqn2, eqn3], [x, y, z])
```

```
   (55)
   [[x= 1,y= 1,z= 1], [x= - 1,y= 1,z= 1],
             2                      2
    [- 3z + x  + 2= 0,y= - 3z + 1,3z  - 2z + 1= 0],

                                                4      3      2
         4      3      2          2        - 18z  + 24z  + 21z  + 12z + 3
     [12z  - 12z  - 30z  + 7z + 3x = 0, y= ------------------------------,
                                                          2
        5     4     3     2
      6z  - 6z  - 9z  - 7z  - 3z - 1= 0]
     ]
                         Type: List List Equation Fraction Polynomial Integer
               Time: 0.02 (IN) + 1.18 (EV) + 0.03 (OT) + 0.03 (GC) = 1.27 sec
)clear properties eqn1 eqn2 eqn3

-- ---------- Quit ----------
)quit


real   118.9
user   38.5
sys    0.9
```

14 Inequalities

```
-- ---------- Inequalities ----------
-- => True
%e**%pi > %pi**%e


   (1)  false
                                                                 Type: Boolean
               Time: 0.30 (IN) + 0.38 (EV) + 0.18 (OT) + 0.15 (GC) = 1.02 sec
-- => [True, False]
[x**4 - x + 1 > 0, x**4 - x + 1 > 1]


   (2)  [true,true]
                                                            Type: List Boolean
                            Time: 0.05 (IN) + 0.02 (EV) + 0.02 (OT) = 0.08 sec
```

```
-- => True
--assume(abs(x) < 1)
-1 < x and x < 1


   (3)  false
                                                       Type: Boolean
                                    Time: 0.23 (IN) + 0.02 (OT) = 0.25 sec
-- x > y > 0 and k, n > 0   =>   k x^n > k y^n
--assume(x > y, y > 0)
2*x**2 > 2*y**2


   (4)  false
                                                       Type: Boolean
                                    Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
--assume(k > 0)
k*x**2 > k*y**2


   (5)  false
                                                       Type: Boolean
                                                Time: 0.02 (EV) = 0.02 sec
--assume(n > 0)
k*x**n > k*y**n


   (6)  false
                                                       Type: Boolean
                                    Time: 0.18 (IN) + 0.03 (OT) = 0.22 sec
-- x > 1 and y >= x - 1   =>   y > 0
--assume(x > 1, y >= x - 1)
y > 0


   (7)  true
                                                       Type: Boolean
                                                Time: 0.03 (IN) = 0.03 sec
-- x >= y, y >= z, z >= x   =>   x = y = z
--assume(x >= y, y >= z, z >= x)
[x = y, x = z, y = z]


   (8)  [x= y,x= z,y= z]
                                           Type: List Equation Symbol
                                    Time: 0.13 (IN) + 0.02 (OT) = 0.15 sec
```

```
-- x < -1 or x > 3
solve(abs(x - 1) > 2, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                    Boolean
                                  Variable x

-- x < 1 or 2 < x < 3 or 4 < x < 5
solve((x - 1)*(x - 2)*(x - 3)*(x - 4)*(x - 5) < 0, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                    Boolean
                                  Variable x

-- x < 3 or x >= 5
solve(6/(x - 3) <= 3, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                    Boolean
```

```
                              Variable x

-- => 0 <= x < 4
solve(sqrt(x) < 2, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                           )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                Boolean
                              Variable x

-- => x is real
solve(sin(x) < 2, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                           )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                Boolean
                              Variable x

-- => x != pi/2 + n 2 pi
solve(sin(x) < 1, x)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                           )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
```

```
      with argument type(s)
                                Boolean
                              Variable x

-- The next two examples come from Abdubrahim Muhammad Farhat, _Stability
-- Analysis of Finite Difference Schemes_, Ph.D. dissertation, University of
-- New Mexico, Albuquerque, New Mexico, December 1993 => 0 <= A <= 1/2
solve(abs(2*A*(cos(t) - 1) + 1) <= 1, A)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                Boolean
                              Variable A

-- => 125 A^4 + 24 A^2 - 48 < 0   or   |A| < 2/5 sqrt([8 sqrt(6) - 3]/5)
solve(A**2*(cos(t) - 4)**2*sin(t)**2 < 9, A)

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                                Boolean
                              Variable A

-- => |x| < y
solve([x + y > 0, x - y < 0], [x, y])

   There are 18 exposed and 3 unexposed library operations named solve
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op solve
      to learn more about the available operations. Perhaps
```

```
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                              List Boolean
                       List OrderedVariableList [x,y]

-- ---------- Quit ----------
)quit


real   24.5
user   6.0
sys    0.4
```

15 Vector Analysis

```
-- ---------- Vector Analysis ----------
-- Vector norm => sqrt(15)
Norm(v) ==
   local V
   V : Matrix Complex Integer := v
   sqrt(map(conjugate, transpose(V)) * V)

                                                                   Type: Void
                                                                  Time: 0 sec
Norm(vector([1 + %i, -2, 3*%i]))

   Compiling function Norm with type Vector Complex Integer ->
      Expression Complex Integer

         +--+
   (2)  \|15
                                              Type: Expression Complex Integer
               Time: 0.97 (IN) + 0.23 (EV) + 0.33 (OT) + 0.17 (GC) = 1.70 sec
)clear properties Norm

   Compiled code for Norm has been cleared.
-- Cross product: (2, 2, -3) x (1, 3, 1) => (11, -5, 4)
--cross(vector([2, 2, -3]), vector([1, 3, 1]))
-- (a x b) . (c x d) => (a . c) (b . d) - (a . d) (b . c)
--cross(a, b) . cross(c, d)
-- => (2 y z^3 - 2 x^2 y^2 z,   x y,   2 x y^2 z^2 - x z)
```

```
--curl(vector([x*y*z, x**2*y**2*z**2, y**2*z**3]))
-- DEL . (f x g) => g . (DEL x f) - f . (DEL x g)
--div(cross(f, g))
-- Express DEL . a in spherical coordinates (r, theta, phi) for
-- a = (a_r(r, theta, phi), a_theta(r, theta, phi), a_phi(r, theta, phi)).
-- Here, phi is in the x-y plane and theta is the angle with the z-axis.
-- => 1/r^2 d/dr[r^2 a_r] + 1/[r sin(theta)] d/dtheta[sin(theta) a_theta]
--    + 1/[r sin(theta)] da_phi/dphi
-- => da_r/dr + (2 a_r)/r + 1/r da_theta/dtheta + a_theta/[r tan(theta)]
--    + 1/[r sin(theta)] da_phi/dphi
-- See Keith R. Symon, _Mechanics_, Third Edition, Addison-Wesley Publishing
-- Company, 1971, p. 103.
--coordinates == spherical
--div([a_r(r, theta, phi), a_theta(r, theta, phi), a_phi(r, theta, phi)])
-- Express dR/dt in spherical coordinates (r, theta, phi) where R is the
-- position vector r*Rhat(theta, phi) with Rhat being the unit vector in the
-- direction of R => (dr/dt, r dtheta/dt, r sin(theta) dphi/dt)
-- [Symon, p. 98]
r:= operator('r);


                                                        Type: BasicOperator
                                                 Time: 0.08 (IN) = 0.08 sec
rhat:= operator('rhat);


                                                        Type: BasicOperator
                                                 Time: 0.03 (IN) = 0.03 sec
theta:= operator('theta);


                                                        Type: BasicOperator
                                                 Time: 0.03 (IN) = 0.03 sec
phi:= operator('phi);


                                                        Type: BasicOperator
                                         Time: 0.02 (IN) + 0.03 (OT) = 0.05 sec
v:= vector([r(t)*rhat(theta(t), phi(t)), 0, 0])


   (7)  [r(t)rhat(theta(t),phi(t)),0,0]
                                                Type: Vector Expression Integer
                                 Time: 0.58 (IN) + 0.03 (EV) + 0.14 (OT) = 0.75 sec
D(v, t)
```

```
   There are 5 exposed and 0 unexposed library operations named D
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op D
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named D
      with argument type(s)
                          Vector Expression Integer
                                 Variable t

map(e +-> D(e, t), v)


   (8)
   [
              ,                                      ,
       r(t)phi (t)rhat  (theta(t),phi(t)) + r(t)theta (t)rhat  (theta(t),phi(t))
                      ,2                                      ,1
     +
                             ,
       rhat(theta(t),phi(t))r (t)


     ,
    0, 0]
                                              Type: Vector Expression Integer
                           Time: 0.02 (IN) + 0.10 (EV) + 0.03 (OT) = 0.15 sec
)clear properties r rhat theta phi v

-- Scalar potential => x^2 y + y + 2 z^3
--potential(vector([2*x*y, x**2 + 1, 6*z**2]))
-- Vector potential => (x y z, x^2 y^2 z^2, y^2 z^3) is one possible solution.
-- See Harry F. Davis and Arthur David Snider, _Introduction to Vector
-- Analysis_, Third Edition, Allyn and Bacon, Inc., 1975, p. 97.
--vectorpotential(vector([2*y*z**3 - 2*x**2*y**2*z, x*y, 2*x*y**2*z**2 - x*z]))
--curl(%)
-- Orthogonalize the following vectors (Gram-Schmidt).  See Lee W. Johnson and
-- R. Dean Riess, _Introduction to Linear Algebra_, Addison-Wesley Publishing
-- Company, 1981, p. 104 => [[0 1 2 1], [0 -1 1 -1], [2 1 0 -1]]^T
[transpose(matrix([[0, 1, 2, 1]])), transpose(matrix([[0, 1, 3, 1]])), _
 transpose(matrix([[1, 1, 1, 0]])), transpose(matrix([[1, 3, 6, 2]]))]


         +0+ +0+ +1+ +1+
```

```
         | | | | | | | |
         |1| |1| |1| |3|
   (9)  [| |,| |,| |,| |]
         |2| |3| |1| |6|
         | | | | | | | |
         +1+ +1+ +0+ +2+
                                                    Type: List Matrix Integer
                                       Time: 0.02 (IN) + 0.03 (OT) = 0.05 sec
gramschmidt(%)


              +   1   +
              | ----  |
              |  +-+  |
              |  |3   | +  0   + + 0  +
              |  |-   | |      | |    |
              | \|2   | |   +-+| |  1 |
              |       | |  \|3 | |----|
          +0+ |   1   | |- ----| | +-+|
          | | | ----- | |    3 | |\|6 |
          |0| |    +-+| |      | |    |
   (10)  [| |,|    |3 |,|  +-+ |,|  2 |]
          |0| | 2  |- | | \|3  | |----|
          | | |   \|2 | | ---- | | +-+|
          +0+ |       | |   3  | |\|6 |
              |   0   | |      | |    |
              |       | |   +-+| |  1 |
              |    1  | |  \|3 | |----|
              |- -----| |- ----| | +-+|
              |     +-| +    3 + +\|6 +
              |     |3|
              |  2  |-|
              +    \|2+
                                         Type: List Matrix Expression Integer
               Time: 0.10 (IN) + 0.20 (EV) + 0.07 (OT) + 0.03 (GC) = 0.40 sec
-- ---------- Quit ----------
)quit


real   7.6
user   4.3
sys    0.5
```

16 Matrix Theory

```
-- ---------- Matrix Theory ----------
-- Extract the superdiagonal => [2, 6]
matrix([[1, 2, 3], [4, 5, 6], [7, 8, 9]]);


                                                          Type: Matrix Integer
                           Time: 0.03 (IN) + 0.02 (EV) + 0.02 (GC) = 0.07 sec
[%(j - 1, j) for j in 2..ncols(%)]


   (2)  [2,6]
                                                            Type: List Integer
                                        Time: 0.05 (OT) + 0.02 (GC) = 0.07 sec
-- (2, 3)-minor => [[1, 2], [7, 8]]
--minor(matrix([[1, 2, 3], [4, 5, 6], [7, 8, 9]]), 2, 3)
-- Create the 7 x 6 matrix B from rearrangements of the elements of the 4 x 4
-- matrix A (this is easiest to do with a MATLAB style notation):
-- B = [A(1:3,2:4), A([1,2,4],[3,1,4]); A, [A(1:2,3:4); A([4,1],[3,2])]]
-- => [[12 13 14|13 11 14],
--     [22 23 24|23 21 24],
--     [32 33 34|43 41 44],
--     [--------+--+-----]
--     [11 12 13 14|13 14],
--     [21 22 23 24|23 24],
--     [           +-----]
--     [31 32 33 34|43 42],
--     [41 42 43 44|13 12]].  See Michael James Wester, _Symbolic Calculation
-- and Expression Swell Analysis of Matrix Determinants and Eigenstuff_, Ph.D.
-- dissertation, University of New Mexico, Albuquerque, New Mexico, December
-- 1992, p. 89.
A:= matrix([[11, 12, 13, 14], _
            [21, 22, 23, 24], _
            [31, 32, 33, 34], _
            [41, 42, 43, 44]]);


                                                          Type: Matrix Integer
                                                    Time: 0.02 (IN) = 0.02 sec
vertConcat(horizConcat(subMatrix(A, 1, 3, 2, 4), _
                       matrix([[A(1,3), A(1,1), A(1,4)], _
                               [A(2,3), A(2,1), A(2,4)], _
                               [A(4,3), A(4,1), A(4,4)]])), _
           horizConcat(A, vertConcat(subMatrix(A, 1, 2, 3, 4), _
```

```
                                     matrix([[A(4,3), A(4,2)], _
                                             [A(1,3), A(1,2)]]))))


        +12  13  14  13  11  14+
        |                      |
        |22  23  24  23  21  24|
        |                      |
        |32  33  34  43  41  44|
        |                      |
   (4)  |11  12  13  14  13  14|
        |                      |
        |21  22  23  24  23  24|
        |                      |
        |31  32  33  34  43  42|
        |                      |
        +41  42  43  44  13  12+
                                                         Type: Matrix Integer
                                       Time: 0.07 (IN) + 0.03 (OT) = 0.10 sec
)clear properties A

-- Create a block diagonal matrix
diagonalMatrix( _
   @[matrix({\Tt{}a,\ 1],[0,\ a\nwendquote}), b, matrix({\Tt{}c,\ 1,\ 0],[0,\ c,\ 1],[0,\ 0,


   (5)  diagonalMatrix
                               +c  1  0+
                      +a  1+   |       |
                      |    |,b,|0  c  1|
                      +0  a+   |       |
                               +0  0  c+
                                                                 Type: Symbol
                           Time: 0.30 (IN) + 0.22 (OT) + 0.02 (GC) = 0.53 sec
-- => [[1 1], [1 0]]
matrix([[7, 11], [3, 8]]) :: Matrix(PrimeField(2))


        +1  1+
   (6)  |    |
        +1  0+
                                                     Type: Matrix PrimeField 2
                           Time: 0.07 (IN) + 0.05 (OT) + 0.05 (GC) = 0.17 sec
-- => [[-cos t, -sin t], [sin t, -cos t]]
matrix([[cos(t), sin(t)], [-sin(t), cos(t)]])
```

```
        + cos(t)   sin(t)+
   (7)  |                |
        +- sin(t)  cos(t)+
                                              Type: Matrix Expression Integer
               Time: 0.22 (IN) + 0.23 (EV) + 0.07 (OT) + 0.10 (GC) = 0.62 sec
D(%, t, 2)


        +- cos(t)  - sin(t)+
   (8)  |                  |
        + sin(t)   - cos(t)+
                                     Type: SquareMatrix(2,Expression Integer)
                                       Time: 0.32 (IN) + 0.07 (EV) = 0.38 sec
-- => [[(a + 7) x + (2 a - 8) y,   (3 a - 9) x + (4 a + 10) y,
--      (5 a + 11) x + (6 a - 12) y]]
matrix([[x, y]]) * (a*matrix([[1, 3, 5], [2, 4, 6]]) _
                    + matrix({\Tt{}7,\ -9,\ 11],\ [-8,\ 10,\ -12\nwendquote}))


   (9)
   @[(2a - 8)y + (a + 7)x  (4a + 10)y + (3a - 9)x  (6a - 12)y + (5a + 11)x]
                                              Type: Matrix Polynomial Integer
                                       Time: 1.58 (IN) + 0.35 (OT) = 1.93 sec
-- Matrix norms: infinity norm => 7
norm(matrix([[1, -2*%i], [-3*%i, 4]]), %plusInfinity)

   There are 5 exposed and 6 unexposed library operations named norm
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op norm
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named norm
      with argument type(s)
                           Matrix Complex Integer
                          OrderedCompletion Integer

-- Frobenius norm => (a^2 + b^2 + c^2)/(|a| |b| |c|)   (a, b, c real)
norm(matrix([[a/(b*c), 1/c, 1/b], [1/c, b/(a*c), 1/a], [1/b, 1/a, c/(a*b)]]), _
     ’f)

   There are 5 exposed and 6 unexposed library operations named norm
      having 2 argument(s) but none was determined to be applicable.
```

```
      Use HyperDoc Browse, or issue
                              )display op norm
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named norm
      with argument type(s)
                     Matrix Fraction Polynomial Integer
                                 Variable f

-- Hermitian (complex conjugate transpose) => [[1, f(4 + 5 i)], [2 - 3 i, 6]]
-- (This assumes f is a real valued function.  In general, the (1, 2) entry
-- will be conjugate[f(4 - 5 i)] = conjugate(f)(4 + 5 i).)
f:= operator('f);



                                                          Type: BasicOperator
                                                   Time: 0.07 (IN) = 0.07 sec
map('conjugate, transpose(matrix([[1, 2 + 3*%i], [f(4 - 5*%i), 6]])) _
                   :: Matrix Complex Expression Integer)


   Cannot convert from type Matrix Expression Complex Integer to Matrix
      Complex Expression Integer for value
   +   1     f(4 - 5%i)+
   |                   |
   +2 + 3%i      6     +

m:= matrix([[a, b], [1, a*b]])


         +a   b +
   (11)  |      |
         +1  a b+
                                              Type: Matrix Polynomial Integer
                                       Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
-- Invert the matrix => 1/(a^2 - 1) [[a, -1], [-1/b, a/b]]
minv:= inverse(m)


         +     a            1   +
         |  ------     - ------ |
         |   2            2     |
         |  a  - 1       a  - 1 |
   (12)  |                      |
```

```
         |      1          a    |
         |- ---------  ---------|
         |    2          2      |
         +  (a  - 1)b  (a  - 1)b+
                          Type: Union(Matrix Fraction Polynomial Integer,...)
                           Time: 0.05 (IN) + 0.03 (EV) + 0.02 (OT) = 0.10 sec
m * minv


         +1  0+
   (13)  |    |
         +0  1+
                                     Type: Matrix Fraction Polynomial Integer
                                       Time: 0.38 (IN) + 0.07 (OT) = 0.45 sec
)clear properties m minv

-- Inverse of a triangular partitioned (or block) matrix
-- => [[A_11^(-1), -A_11^(-1) A_12 A_22^(-1)], [0, A_22^(-1)]].
-- See Charles G. Cullen, _Matrices and Linear Transformations_, Second
-- Edition, Dover Publications Inc., 1990, p. 35.
matrix([[A11, A12], [0, A22]])**(-1)


         + 1       A12  +
         |---  - -------|
         |A11    A11 A22|
   (14)  |              |
         |         1    |
         | 0      ---   |
         +        A22   +
                                     Type: Matrix Fraction Polynomial Integer
                                       Time: 0.15 (IN) + 0.02 (OT) = 0.17 sec
-- LU decomposition of a symbolic matrix   [David Wood]
-- [ 1    0   0] [1  x-2  x-3]   [ 1      x-2        x-3    ]
-- [x-1   1   0] [0   4   x-5] = [x-1  x^2-3x+6   x^2-3x-2  ]
-- [x-2  x-3  1] [0   0   x-7]   [x-2    x^2-8   2x^2-12x+14]
matrix([[ 1,     x-2,          x-3      ], _
        [x-1, x**2-3*x+6,   x**2-3*x-2  ], _
        [x-2,   x**2-8,   2*x**2-12*x+14]]);


                                               Type: Matrix Polynomial Integer
                                                    Time: 0.08 (IN) = 0.08 sec
-- Reduced row echelon form   [Cullen, p. 43]
-- => [[1 0 -1 0 2], [0 1 2 0 -1], [0 0 0 1 3], [0 0 0 0 0]]
matrix([[1, 2, 3, 1, 3], _
```

```
        [3, 2, 1, 1, 7], _
        [0, 2, 4, 1, 1], _
        [1, 1, 1, 1, 4]]);


                                                         Type: Matrix Integer
                                                   Time: 0.03 (IN) = 0.03 sec
rowEchelon(%)


         +1  0  - 1  0   2 +
         |                 |
         |0  1   2   0  - 1|
   (17)  |                 |
         |0  0   0   1   3 |
         |                 |
         +0  0   0   0   0 +
                                                         Type: Matrix Integer
                                       Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
-- => 2.  See Gerald L. Bradley, _A Primer of Linear Algebra_, Prentice-Hall,
--    Inc., 1975, p. 135.
rank(matrix([[-1, 3, 7, -5], [4, -2, 1, 3], [2, 4, 15, -7]]))


   (18)  2
                                                        Type: PositiveInteger
                                                   Time: 0.02 (OT) = 0.02 sec
-- => 1
rank(matrix([[2*sqrt(2), 8], [6*sqrt(6), 24*sqrt(3)]]))


   (19)  1
                                                        Type: PositiveInteger
                           Time: 0.05 (IN) + 0.35 (EV) + 0.03 (OT) = 0.43 sec
-- => 1
rank(matrix([[sin(2*t), cos(2*t)], _
             [2*(1 - cos(t)**2)*cos(t), (1 - 2*sin(t)**2)*sin(t)]]))


   (20)  2
                                                        Type: PositiveInteger
                           Time: 0.10 (IN) + 0.02 (EV) + 0.02 (OT) = 0.13 sec
-- Null space => [[2 4 1 0], [0 -3 0 1]]^T or variant   [Bradley, p. 207]
nullSpace(matrix([[1, 0, -2, 0], [-2, 1, 0, 3], [-1, 2, -6, 6]]))
```

```
   (21)  [[2,4,1,0],[0,- 3,0,1]]
                                                    Type: List Vector Integer
                                       Time: 0.03 (EV) + 0.02 (OT) = 0.05 sec
-- Define a Vandermonde matrix (useful for doing polynomial interpolations)
matrix([[1,    1,    1,    1   ], _
        [w,    x,    y,    z   ], _
        [w**2, x**2, y**2, z**2], _
        [w**3, x**3, y**3, z**3]])


         +1   1   1   1 +
         |              |
         |w   x   y   z |
         |              |
   (22)  | 2   2   2   2|
         |w   x   y   z |
         |              |
         | 3   3   3   3|
         +w   x   y   z +
                                              Type: Matrix Polynomial Integer
                                       Time: 0.15 (IN) + 0.05 (OT) = 0.20 sec
determinant(%)


   (23)
              2       2    2        2    2   3
     ((x - w)y  + (- x  + w )y + w x  - w x)z
   +
                3     3    3        3    3   2
     ((- x + w)y  + (x  - w )y - w x  + w x)z
   +
        2    2  3       3    3  2    2 3    3 2           2    2   3
     ((x  - w )y  + (- x  + w )y  + w x  - w x )z + (- w x  + w x)y
   +
         3    3   2       2 3    3 2
     (w x  - w x)y  + (- w x  + w x )y
                                                     Type: Polynomial Integer
                                                   Time: 0.07 (OT) = 0.07 sec
-- The following formula implies a general result:
-- => (w - x) (w - y) (w - z) (x - y) (x - z) (y - z)
factor(%)


   (24)  (x - w)(y - x)(y - w)(z - y)(z - x)(z - w)
                                            Type: Factored Polynomial Integer
                           Time: 0.07 (IN) + 0.25 (EV) + 0.03 (OT) = 0.35 sec
```

```
-- Minimum polynomial => (lambda - 1)^2 (lambda + 1)   [Cullen, p. 181]
matrix([[17,  -8, -12, 14], _
        [46, -22, -35, 41], _
        [-2,   1,   4, -4], _
        [ 4,  -2,  -2,  3]]);


                                                          Type: Matrix Integer
                                                                   Time: 0 sec
minimalPolynomial(% :: SquareMatrix(4, Integer))

   There are 2 exposed and 1 unexposed library operations named
      minimalPolynomial having 1 argument(s) but none was determined to
      be applicable. Use HyperDoc Browse, or issue
                        )display op minimalPolynomial
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      minimalPolynomial with argument type(s)
                           SquareMatrix(4,Integer)

-- Compute the eigenvalues of a matrix from its characteristic polynomial
-- => lambda = {1, -2, 3}
m:= matrix([[ 5, -3, -7], _
            [-2,  1,  2], _
            [ 2, -3, -4]])


         + 5   - 3  - 7+
         |             |
   (26)  |- 2   1    2 |
         |             |
         + 2   - 3  - 4+
                                                          Type: Matrix Integer
                                                    Time: 0.02 (OT) = 0.02 sec
characteristicPolynomial(m, lambda)


                 3          2
   (27)  - lambda  + 2lambda  + 5lambda - 6
                                                      Type: Polynomial Integer
                                        Time: 0.02 (IN) + 0.03 (OT) = 0.05 sec
solve(% = 0, lambda)
```

```
   (28)  [lambda= 3,lambda= 1,lambda= - 2]
                            Type: List Equation Fraction Polynomial Integer
                         Time: 0.47 (IN) + 0.18 (EV) + 0.07 (OT) = 0.72 sec
)clear properties m

-- In theory, an easy eigenvalue problem! => lambda = {2 - a} for k = 1..100
-- [Wester, p. 154]
identityMatrix(n) == diagonalMatrix([1 for i in 1..n])

                                                                 Type: Void
                                                                Time: 0 sec
eigenvalues((2 - a)*identityMatrix(100))

   Compiling function identityMatrix with type PositiveInteger ->
      Matrix Integer

   (30)  [- a + 2]
Type: List Union(Fraction Polynomial Integer,SuchThat(Symbol,Polynomial Integer))
             Time: 31.44 (IN) + 7.85 (EV) + 4.32 (OT) + 5.17 (GC) = 48.77 sec
-- => lambda = {4 sin^2(pi k/[2 (n + 1)])} for k = 1..n for an n x n matrix.
--    For n = 5, lambda = {2 - sqrt(3), 1, 2, 3, 2 + sqrt(3)}
-- See J. H. Wilkinson, _The Algebraic Eigenvalue Problem_, Oxford University
-- Press, 1965, p. 307.
matrix([[2, 1, 0, 0, 0], _
        [1, 2, 1, 0, 0], _
        [0, 1, 2, 1, 0], _
        [0, 0, 1, 2, 1], _
        [0, 0, 0, 1, 2]])


         +2  1  0  0  0+
         |             |
         |1  2  1  0  0|
         |             |
   (31)  |0  1  2  1  0|
         |             |
         |0  0  1  2  1|
         |             |
         +0  0  0  1  2+
                                                        Type: Matrix Integer
                                               Time: 0.02 (IN) = 0.02 sec
radicalEigenvalues(%)


             +-+      +-+
```

```
   (32)  [- \|3  + 2,\|3  + 2,1,2,3]
                                                Type: List Expression Integer
                           Time: 0.08 (IN) + 0.08 (EV) + 0.02 (OT) = 0.18 sec
-- Eigenvalues of the Rosser matrix.  This matrix is notorious for causing
-- numerical eigenvalue routines to fail.   [Wester, p. 146 (Cleve Moler)]
-- => {-10 sqrt(10405), 0, 510 - 100 sqrt(26), 1000, 1000,
--     510 + 100 sqrt(26), 1020, 10 sqrt(10405)} =
--    {-1020.049, 0, 0.098, 1000, 1000, 1019.902, 1020, 1020.049}
rosser:= matrix({\Tt{}\ 611,\ \ 196,\ -192,\ \ 407,\ \ \ -8,\ \ -52,\ \ -49,\ \ \ 29],\ {\_
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ 196,\ \ 899,\ \ 113,\ -192,\ \ -71,\ \ -43,\ \ \ -8,\ \
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [-192,\ \ 113,\ \ 899,\ \ 196,\ \ \ 61,\ \ \ 49,\ \ \ \ 8,
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ 407,\ -192,\ \ 196,\ \ 611,\ \ \ \ 8,\ \ \ 44,\ \ \ 59,
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ \ -8,\ \ -71,\ \ \ 61,\ \ \ \ 8,\ \ 411,\ -599,\ \ 208,
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ -52,\ \ -43,\ \ \ 49,\ \ \ 44,\ -599,\ \ 411,\ \ 208,\
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ -49,\ \ \ -8,\ \ \ \ 8,\ \ \ 59,\ \ 208,\ \ 208,\ \ \
\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ [\ \ 29,\ \ -44,\ \ \ 52,\ \ -23,\ \ 208,\ \ 208,\ -911,\


                                                         Type: Matrix Integer
                                                   Time: 0.02 (IN) = 0.02 sec
radicalEigenvalues(rosser)


   (34)
       +-----+      +-----+       +---+           +---+
   [10\|10405 ,- 10\|10405 ,- 100\|26  + 510,100\|26  + 510,0,1000,1000,1020]
                                                Type: List Expression Integer
                            Time: 0.12 (IN) + 1.0 (EV) + 0.02 (OT) = 1.13 sec
realEigenvalues(rosser, 1.0e-6)


   (35)
   [- 1020.0490183830 26123, 0.0, 1020.0490183830 26123, 1020.0,
    1019.9019513130 187988, 1000.0, 0.0980486869 8120117187 5]
                                                             Type: List Float
              Time: 0.07 (IN) + 0.28 (EV) + 0.05 (OT) + 0.02 (GC) = 0.42 sec
eigenvalues(rosser :: Matrix(Float))

   WARNING (genufact): No known algorithm to factor
        125            124                 123                       122
       ?    - 86215.0 ?    + 3672643562.0 ?    - 10305 2524136076.0 ?
     +
                               121                                 120
       214242065 1760625974.0 ?    - 0.3519437981 9139045528 E 23 ?
     +
                                     119                                   118
```

```
  0.4757930276 0524392937 E 27 ?    - 0.5443719612 8759169508 E 31 ?
+
                                117                                 116
  0.5379979009 8770356924 E 35 ?    - 0.4664749268 2226816519 E 39 ?
+
                                115                                 114
  0.3592087692 2534478781 E 43 ?    - 0.2480907901 4852864426 E 47 ?
+
                                113                                 112
  0.1549276891 3929107162 E 51 ?    - 0.8807036930 5149411468 E 54 ?
+
                                111                                 110
  0.4583391720 1504761731 E 58 ?    - 0.2194388254 7216326309 E 62 ?
+
                               109                                 108
  0.9705762773 604715712 E 65 ?    - 0.3980292705 4424329091 E 69 ?
+
                                107                                 106
  0.1518264867 7783075257 E 73 ?    - 0.5401752426 0724778564 E 76 ?
+
                                105                                 104
  0.1796955195 1515805598 E 80 ?    - 0.5601295905 4406117874 E 83 ?
+
                                103                                 102
  0.1639113863 1096220906 E 87 ?    - 0.4510439265 9444505639 E 90 ?
+
                                101                                 100
  0.1168814463 6800724178 E 94 ?    - 0.2855827986 1382495038 E 97 ?
+
                                 99                                  98
  0.6586280305 3086327597 E 100 ?   - 0.1435005623 1770466372 E 104 ?
+
                                 97                                  96
  0.2955815573 3367270221 E 107 ?   - 0.5758845873 428492513 E 110 ?
+
                                 95                                  94
  0.1061610141 8172155474 E 114 ?   - 0.1851850768 7922934268 E 117 ?
+
                                 93                                  92
  0.3056218491 6782132505 E 120 ?   - 0.4769657381 1684487203 E 123 ?
+
                                 91                                  90
  0.7032699330 7418158372 E 126 ?   - 0.9782596656 7436389234 E 129 ?
+
                                 89                                  88
  0.1280860874 3712484582 E 133 ?   - 0.1573106414 8242673085 E 136 ?
```

```
+
                                87                                  86
  0.1802439290 923378293 E 139 ?   - 0.1909575702 2321405039 E 142 ?
+
                                 85                                  84
  0.1841324046 7558627001 E 145 ?   - 0.1565578204 6290786175 E 148 ?
+
                                 83                                  82
  0.1084018298 6945156255 E 151 ?   - 0.4377988417 1758608725 E 153 ?
+
                                   81                                  80
  - 0.2968496265 1894553792 E 156 ?   + 0.1023002729 7224474459 E 160 ?
+
                                   79                                  78
  - 0.1643770493 4269005245 E 163 ?   + 0.2083326959 4734561364 E 166 ?
+
                                   77                                  76
  - 0.2301999742 2965920355 E 169 ?   + 0.2301309174 9423652886 E 172 ?
+
                                   75                                  74
  - 0.2118385504 6752677393 E 175 ?   + 0.1812642708 7694905254 E 178 ?
+
                                   73                                  72
  - 0.1449610412 5042294701 E 181 ?   + 0.1086839448 8924848295 E 184 ?
+
                                   71                                  70
  - 0.7651391596 9967540208 E 186 ?   + 0.5060299956 3124704326 E 189 ?
+
                                   69                                  68
  - 0.3142367184 7713335751 E 192 ?   + 0.1829574772 0415689208 E 195 ?
+
                                   67                                  66
  - 0.9961784415 3166827826 E 197 ?   + 0.5051793379 2698179236 E 200 ?
+
                                   65                                  64
  - 0.2370885486 7640128904 E 203 ?   + 0.1019224176 5460569951 E 206 ?
+
                                   63                                 62
  - 0.3942260312 5442307776 E 208 ?   + 0.1323834699 481648387 E 211 ?
+
                                   61                                  60
  - 0.3524632481 6587139553 E 213 ?   + 0.4907904491 2672068684 E 215 ?
+
                                 59                                  58
  0.1935153311 3162677451 E 218 ?   - 0.2135886927 6123692419 E 221 ?
+
```

```
                                 57                                  56
  0.1223237295 5917448702 E 224 ?   - 0.5462064456 7735058373 E 226 ?
+
                                 55                                  54
  0.2062619424 5435174697 E 229 ?   - 0.6737922494 4291403287 E 231 ?
+
                                 53                                52
  0.1898827082 1604329724 E 234 ?   - 0.4486781196 52298524 E 236 ?
+
                                 51                                  50
  0.8133738511 6364504076 E 238 ?   - 0.7358667283 4876131493 E 240 ?
+
                                  49                                  48
  - 0.1986000215 428423829 E 243 ?   + 0.1331591187 7791407655 E 246 ?
+
                                   47                                  46
  - 0.4492807815 4925031508 E 248 ?   + 0.1106530639 7295677952 E 251 ?
+
                                   45                                  44
  - 0.2046039509 2572215835 E 253 ?   + 0.2430619124 0670333847 E 255 ?
+
                                 43                                  42
  0.3252259818 2888244489 E 256 ?   - 0.1131150344 9655271066 E 260 ?
+
                                 41                                  40
  0.3691051035 7773753178 E 262 ?   - 0.7900011265 6063011579 E 264 ?
+
                                 39                                  38
  0.1233981952 5026790198 E 267 ?   - 0.1270418736 3851064032 E 269 ?
+
                                 37                                  36
  0.1660218706 9618209188 E 270 ?   + 0.2862105879 5857092645 E 273 ?
+
                                   35                                  34
  - 0.8110417668 3161233591 E 275 ?   + 0.1484448521 8686301081 E 278 ?
+
                                  33                                  32
  - 0.2123387008 794925024 E 280 ?   + 0.2506751820 5612670323 E 282 ?
+
                                   31                                  30
  - 0.2494329065 6728197272 E 284 ?   + 0.2109806712 8889498289 E 286 ?
+
                                   29                                  28
  - 0.1521224934 1012072561 E 288 ?   + 0.9351456154 4154324412 E 289 ?
+
                                   27                                  26
```

```
    - 0.4897636603 2042451012 E 291 ?   + 0.2184259738 2222779488 E 293 ?
  +
                                     25                                  24
    - 0.8298614578 7219241205 E 294 ?   + 0.2689816827 0170685903 E 296 ?
  +
                                     23                                  22
    - 0.7455272800 2234756452 E 297 ?   + 0.1771957532 6653063304 E 299 ?
  +
                                     21                                  20
    - 0.3622167611 4512303391 E 300 ?   + 0.6385537866 7592677628 E 301 ?
  +
                                     19                                  18
    - 0.9730487915 5887722779 E 302 ?   + 0.1283818926 2924796953 E 304 ?
  +
                                     17                                  16
    - 0.1467911675 9006671118 E 305 ?   + 0.1454586351 2070495541 E 306 ?
  +
                                     15                                  14
    - 0.1247986846 9975071194 E 307 ?   + 0.9251453429 5406258973 E 307 ?
  +
                                     13                                  12
    - 0.5905679202 8110037735 E 308 ?   + 0.3230290678 3860457938 E 309 ?
  +
                                     11                                  10
    - 0.1503623501 5158820289 E 310 ?   + 0.5900666022 0150867207 E 310 ?
  +
                                     9                                  8
    - 0.1927643191 6531249396 E 311 ?  + 0.5152198729 6418385783 E 311 ?
  +
                                     7                                  6
    - 0.1099680204 0615001798 E 312 ?  + 0.1809295237 1502506625 E 312 ?
  +
                                     5                                  4
    - 0.2172461015 8315085132 E 312 ?  + 0.1733446261 3560586487 E 312 ?
  +
                                     3                                  2
    - 0.7597264055 3053863391 E 311 ?  + 0.1050398077 8828775171 E 311 ?
  +
    - 0.4443971993 2926345443 E 309 ? - 0.7155498947 0116891371 E 291
  , trying square-free.

(36)
[
    %G
  |
        125             124                  123                        122
```

```
  %G    - 86215.0 %G    + 3672643562.0 %G    - 10305 2524136076.0 %G
+
                           121                                  120
  214242065 1760625974.0 %G    - 0.3519437981 9139045528 E 23 %G
+
                                 119                                  118
  0.4757930276 0524392937 E 27 %G    - 0.5443719612 8759169508 E 31 %G
+
                                 117                                  116
  0.5379979009 8770356924 E 35 %G    - 0.4664749268 2226816519 E 39 %G
+
                                 115                                  114
  0.3592087692 2534478781 E 43 %G    - 0.2480907901 4852864426 E 47 %G
+
                                 113                                  112
  0.1549276891 3929107162 E 51 %G    - 0.8807036930 5149411468 E 54 %G
+
                                 111                                  110
  0.4583391720 1504761731 E 58 %G    - 0.2194388254 7216326309 E 62 %G
+
                                109                                  108
  0.9705762773 604715712 E 65 %G    - 0.3980292705 4424329091 E 69 %G
+
                                 107                                  106
  0.1518264867 7783075257 E 73 %G    - 0.5401752426 0724778564 E 76 %G
+
                                 105                                  104
  0.1796955195 1515805598 E 80 %G    - 0.5601295905 4406117874 E 83 %G
+
                                 103                                  102
  0.1639113863 1096220906 E 87 %G    - 0.4510439265 9444505639 E 90 %G
+
                                 101                                  100
  0.1168814463 6800724178 E 94 %G    - 0.2855827986 1382495038 E 97 %G
+
                                  99                                   98
  0.6586280305 3086327597 E 100 %G   - 0.1435005623 1770466372 E 104 %G
+
                                  97                                  96
  0.2955815573 3367270221 E 107 %G   - 0.5758845873 428492513 E 110 %G
+
                                  95                                   94
  0.1061610141 8172155474 E 114 %G   - 0.1851850768 7922934268 E 117 %G
+
                                  93                                   92
  0.3056218491 6782132505 E 120 %G   - 0.4769657381 1684487203 E 123 %G
```

```
+
                                   91                                    90
  0.7032699330 7418158372 E 126 %G   - 0.9782596656 7436389234 E 129 %G
+
                                   89                                    88
  0.1280860874 3712484582 E 133 %G   - 0.1573106414 8242673085 E 136 %G
+
                                  87                                    86
  0.1802439290 923378293 E 139 %G   - 0.1909575702 2321405039 E 142 %G
+
                                   85                                    84
  0.1841324046 7558627001 E 145 %G   - 0.1565578204 6290786175 E 148 %G
+
                                   83                                    82
  0.1084018298 6945156255 E 151 %G   - 0.4377988417 1758608725 E 153 %G
+
                                     81
  - 0.2968496265 1894553792 E 156 %G
+
                                   80                                    79
  0.1023002729 7224474459 E 160 %G   - 0.1643770493 4269005245 E 163 %G
+
                                   78                                    77
  0.2083326959 4734561364 E 166 %G   - 0.2301999742 2965920355 E 169 %G
+
                                   76                                    75
  0.2301309174 9423652886 E 172 %G   - 0.2118385504 6752677393 E 175 %G
+
                                   74                                    73
  0.1812642708 7694905254 E 178 %G   - 0.1449610412 5042294701 E 181 %G
+
                                   72                                    71
  0.1086839448 8924848295 E 184 %G   - 0.7651391596 9967540208 E 186 %G
+
                                   70                                    69
  0.5060299956 3124704326 E 189 %G   - 0.3142367184 7713335751 E 192 %G
+
                                   68                                    67
  0.1829574772 0415689208 E 195 %G   - 0.9961784415 3166827826 E 197 %G
+
                                   66                                    65
  0.5051793379 2698179236 E 200 %G   - 0.2370885486 7640128904 E 203 %G
+
                                   64                                    63
  0.1019224176 5460569951 E 206 %G   - 0.3942260312 5442307776 E 208 %G
+
```

```
                                 62                                    61
  0.1323834699 481648387 E 211 %G   - 0.3524632481 6587139553 E 213 %G
+
                                  60                                    59
  0.4907904491 2672068684 E 215 %G   + 0.1935153311 3162677451 E 218 %G
+
                                    58
  - 0.2135886927 6123692419 E 221 %G
+
                                  57                                    56
  0.1223237295 5917448702 E 224 %G   - 0.5462064456 7735058373 E 226 %G
+
                                  55                                    54
  0.2062619424 5435174697 E 229 %G   - 0.6737922494 4291403287 E 231 %G
+
                                  53                                  52
  0.1898827082 1604329724 E 234 %G   - 0.4486781196 52298524 E 236 %G
+
                                  51                                    50
  0.8133738511 6364504076 E 238 %G   - 0.7358667283 4876131493 E 240 %G
+
                                   49
  - 0.1986000215 428423829 E 243 %G
+
                                  48                                    47
  0.1331591187 7791407655 E 246 %G   - 0.4492807815 4925031508 E 248 %G
+
                                  46                                    45
  0.1106530639 7295677952 E 251 %G   - 0.2046039509 2572215835 E 253 %G
+
                                  44                                    43
  0.2430619124 0670333847 E 255 %G   + 0.3252259818 2888244489 E 256 %G
+
                                    42
  - 0.1131150344 9655271066 E 260 %G
+
                                  41                                    40
  0.3691051035 7773753178 E 262 %G   - 0.7900011265 6063011579 E 264 %G
+
                                  39                                    38
  0.1233981952 5026790198 E 267 %G   - 0.1270418736 3851064032 E 269 %G
+
                                  37                                    36
  0.1660218706 9618209188 E 270 %G   + 0.2862105879 5857092645 E 273 %G
+
                                    35
```

```
  - 0.8110417668 3161233591 E 275 %G
+
                                 34                                 33
  0.1484448521 8686301081 E 278 %G  - 0.2123387008 794925024 E 280 %G
+
                                 32                                  31
  0.2506751820 5612670323 E 282 %G  - 0.2494329065 6728197272 E 284 %G
+
                                 30                                  29
  0.2109806712 8889498289 E 286 %G  - 0.1521224934 1012072561 E 288 %G
+
                                 28                                  27
  0.9351456154 4154324412 E 289 %G  - 0.4897636603 2042451012 E 291 %G
+
                                 26                                  25
  0.2184259738 2222779488 E 293 %G  - 0.8298614578 7219241205 E 294 %G
+
                                 24                                  23
  0.2689816827 0170685903 E 296 %G  - 0.7455272800 2234756452 E 297 %G
+
                                 22                                  21
  0.1771957532 6653063304 E 299 %G  - 0.3622167611 4512303391 E 300 %G
+
                                 20                                  19
  0.6385537866 7592677628 E 301 %G  - 0.9730487915 5887722779 E 302 %G
+
                                 18                                  17
  0.1283818926 2924796953 E 304 %G  - 0.1467911675 9006671118 E 305 %G
+
                                 16                                  15
  0.1454586351 2070495541 E 306 %G  - 0.1247986846 9975071194 E 307 %G
+
                                 14                                  13
  0.9251453429 5406258973 E 307 %G  - 0.5905679202 8110037735 E 308 %G
+
                                 12                                  11
  0.3230290678 3860457938 E 309 %G  - 0.1503623501 5158820289 E 310 %G
+
                                 10                                  9
  0.5900666022 0150867207 E 310 %G  - 0.1927643191 6531249396 E 311 %G
+
                                 8                                  7
  0.5152198729 6418385783 E 311 %G - 0.1099680204 0615001798 E 312 %G
+
                                 6                                  5
  0.1809295237 1502506625 E 312 %G - 0.2172461015 8315085132 E 312 %G
```

```
       +
                                         4                                    3
         0.1733446261 3560586487 E 312 %G  - 0.7597264055 3053863391 E 311 %G
       +
                                         2
         0.1050398077 8828775171 E 311 %G  - 0.4443971993 2926345443 E 309 %G
       +
         - 0.7155498947 0116891371 E 291
     ]
Type: List Union(Fraction Polynomial Float,SuchThat(Symbol,Polynomial Float))
             Time: 0.32 (IN) + 12.73 (EV) + 0.82 (OT) + 1.97 (GC) = 15.83 sec
)clear properties rosser

-- Eigenvalues of the generalized hypercompanion matrix of
-- (x^5 + a4*x^4 + a3*x^3 + a2*x^2 + a1*x + a0)*(x^2 + x + 1)^2
-- => {[-1 +- sqrt(3) i]/2, [-1 +- sqrt(3) i]/2,
--     RootsOf(x^5 + a4*x^4 + a3*x^3 + a2*x^2 + a1*x + a0)}
matrix([[-a4, -a3, -a2, -a1, -a0,  0,  0,  0,  0], _
        [  1,   0,   0,   0,   0,  0,  0,  0,  0], _
        [  0,   1,   0,   0,   0,  0,  0,  0,  0], _
        [  0,   0,   1,   0,   0,  0,  0,  0,  0], _
        [  0,   0,   0,   1,   0,  0,  0,  0,  0], _
        [  0,   0,   0,   0,   0, -1, -1,  0,  0], _
        [  0,   0,   0,   0,   0,  1,  0,  0,  0], _
        [  0,   0,   0,   0,   0,  0,  1, -1, -1], _
        [  0,   0,   0,   0,   0,  0,  0,  1,  0]]);


                                                    Type: Matrix Polynomial Integer
                                       Time: 0.15 (IN) + 0.03 (OT) = 0.18 sec
radicalEigenvalues(%)


             +---+        +---+      +---+      +---+
          - \|- 3  - 1 - \|- 3  - 1 \|- 3  - 1 \|- 3  - 1
   (38)  [------------,------------,----------,----------]
                2            2           2          2
                                                      Type: List Expression Integer
                           Time: 0.17 (IN) + 0.07 (EV) + 0.03 (OT) = 0.27 sec
-- Eigenvalues and eigenvectors => lambda = {a, a, a, 1 - i, 1 + i},
-- eigenvectors = [[1 0 0 0 0], [0 0 1 0 0], [0 0 0 1 0],
--                 [0, (1 + i)/2, 0, 0, 1], [0, (1 - i)/2, 0, 0, 1]]^T
matrix([[a,  0, 0, 0, 0], _
        [0,  0, 0, 0, 1], _
        [0,  0, a, 0, 0], _
        [0,  0, 0, a, 0], _
```

```
        [0, -2, 0, 0, 2]]);


                                              Type: Matrix Polynomial Integer
                                                   Time: 0.05 (IN) = 0.05 sec
radicalEigenvectors(%)


   (40)
                                             +     0      +
                                             |            |
                                             |   +---+    |
                                             |- \|- 1  + 1|
              +---+                          |------------|
   [[radval= \|- 1  + 1,radmult= 1,radvect= [|      2     |]],
                                             |            |
                                             |     0      |
                                             |            |
                                             |     0      |
                                             |            |
                                             +     1      +
                                               +    0     +
                                               |          |
                                               | +---+    |
                                               |\|- 1  + 1|
                +---+                          |----------|
    [radval= - \|- 1  + 1,radmult= 1,radvect= [|    2     |]],
                                               |          |
                                               |    0     |
                                               |          |
                                               |    0     |
                                               |          |
                                               +    1     +
                                    +0+ +0+ +1+
                                    | | | | | |
                                    |0| |0| |0|
                                    | | | | | |
    [radval= a,radmult= 3,radvect= [|0|,|1|,|0|]]]
                                    | | | | | |
                                    |1| |0| |0|
                                    | | | | | |
                                    +0+ +0+ +0+
Type: List Record(radval: Expression Integer,radmult: Integer,radvect: List Matrix Expressio
                           Time: 0.03 (IN) + 0.17 (EV) + 0.08 (OT) = 0.28 sec
-- Eigenvalues and generalized eigenvectors   [Johnson and Riess, p. 193]
-- => lambda = {1, 1, 1}, eigenvectors = [[4 -1 4], [1 -1 2], [3 -1 3]]^T
```

```
matrix([[-1,  -8, 1], _
        [-1,  -3, 2], _
        [-4, -16, 7]]);


                                                         Type: Matrix Integer
                                                                  Time: 0 sec
radicalEigenvectors(%)


                                          +- 4+
                                          |   |
   (42)  [[radval= 1,radmult= 3,radvect= [| 1 |]]]
                                          |   |
                                          + 0 +
Type: List Record(radval: Expression Integer,radmult: Integer,radvect: List Matrix Expressio
                           Time: 0.02 (IN) + 0.02 (EV) + 0.02 (OT) = 0.05 sec
-- Eigenvalues and generalized eigenvectors   [Johnson and Riess, p. 199]
-- => lambda = {1, 1, 1, 1, 2, 2}, eigenvectors =
--    [[1 -1  0  0  0 0], [-1 0 0 1 0 0], [0 0 1 -1 0 -1],
--     [0  0 -1 -2 -1 3], [ 0 2 0 0 0 0], [2 0 1  1 0  0]]^T
matrix([[1, 0, 1, 1, 0, 1], _
        [1, 2, 0, 0, 0, 0], _
        [0, 0, 2, 0, 1, 1], _
        [0, 0, 1, 1, 0, 0], _
        [0, 0, 0, 0, 1, 0], _
        [0, 0, 0, 0, 1, 1]]);


                                                         Type: Matrix Integer
                                                   Time: 0.02 (IN) = 0.02 sec
radicalEigenvectors(%)


   (44)
                                    +0+
                                    | |
                                    |1|
                                    | |
                                    |0|
   [[radval= 2,radmult= 2,radvect= [| |]],
                                    |0|
                                    | |
                                    |0|
                                    | |
                                    +0+
```

```
                                    +- 1+
                                    |   |
                                    | 1 |
                                    |   |
                                    | 0 |
    [radval= 1,radmult= 4,radvect= [|   |]]]
                                    | 0 |
                                    |   |
                                    | 0 |
                                    |   |
                                    + 0 +
Type: List Record(radval: Expression Integer,radmult: Integer,radvect: List Matrix Expressio
                           Time: 0.05 (IN) + 0.02 (EV) + 0.05 (OT) = 0.12 sec
-- Jordan form => diag([[1 1],[0 1]], {\Tt{}1\ 1],[0\ 1\nwendquote}, -1)   [Gantmacher, p. 1
matrix([[1,  0,  0,  1, -1], _
        [0,  1, -2,  3, -3], _
        [0,  0, -1,  2, -2], _
        [1, -1,  1,  0,  1], _
        [1, -1,  1, -1,  2]]);



                                                         Type: Matrix Integer
                                                                  Time: 0 sec
-- Smith normal form => [[1, 0], [0, x^4 - x^2 + 1]]   [Cullen, p. 230]
matrix([[x**2, x - 1], [x + 1, x**2]])


         +  2         +
         | x     x - 1|
   (46)  |            |
         |         2  |
         +x + 1   x   +
                                              Type: Matrix Polynomial Integer
                                                   Time: 0.03 (IN) = 0.03 sec
-- Matrix exponential => e [[cos 2, -sin 2], [sin 2, cos 2]]
exp(matrix([[1, -2], [2, 1]]))

   There are 2 exposed and 6 unexposed library operations named exp
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op exp
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named exp
```

```
      with argument type(s)
                           Matrix Integer

-- Matrix exponential   [Rick Niles] =>
-- [[1, 4 sin(w t)/w - 3 t , 6 [w t - sin(w t)], 2/w [1 - cos(w t)] ],
--  [0, 4 cos(w t) - 3     , 6 w [1 - cos(w t)], 2 sin(w t)        ],
--  [0, -2/w [1 - cos(w t)], 4 - 3 cos(w t)    , sin(w t)/w        ],
--  [0, -2 sin(w t)        , 3 w sin(w t)      , cos(w t)          ]]
matrix([[0, 1,    0,      0  ], _
        [0, 0,    0,      2*w], _
        [0, 0,    0,      1  ], _
        [0, -2*w, 3*w**2, 0  ]]);


                                               Type: Matrix Polynomial Integer
                                                    Time: 0.05 (IN) = 0.05 sec
exp(%*t)

   There are 2 exposed and 6 unexposed library operations named exp
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op exp
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named exp
      with argument type(s)
                          Matrix Polynomial Integer

-- Sine of a Jordan matrix => diag([[sin a, cos a],[0, sin a]], sin b,
-- [[sin c, cos c, -sin(c)/2],[0, sin c, cos c],[0, 0, sin c]])
-- See F. R. Gantmacher, _The Theory of Matrices_, Volume One, Chelsea
-- Publishing Company, 1977, p. 100 to see how to do a general function.
matrix([[a, 1, 0, 0, 0, 0], _
        [0, a, 0, 0, 0, 0], _
        [0, 0, b, 0, 0, 0], _
        [0, 0, 0, c, 1, 0], _
        [0, 0, 0, 0, c, 1], _
        [0, 0, 0, 0, 0, c]]);


                                               Type: Matrix Polynomial Integer
                                        Time: 0.03 (IN) + 0.03 (OT) = 0.07 sec
sin(%)
```

```
   There are 2 exposed and 6 unexposed library operations named sin
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op sin
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named sin
      with argument type(s)
                       Matrix Polynomial Integer

-- Sine of a matrix => [[1 0 0], [0 1 0], [0 0 1]]   [Cullen, p. 261]
%pi/2*matrix([[2, 1, 1], [2, 3, 2], [1, 1, 2]]);


                                                            Type: Matrix Pi
                                      Time: 0.30 (IN) + 0.05 (OT) = 0.35 sec
sin(%)

   There are 2 exposed and 6 unexposed library operations named sin
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op sin
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named sin
      with argument type(s)
                               Matrix Pi

-- Matrix square root => {+-[[3 1], [1 4]], +-1/sqrt(5) [[-1 7], [7 6]]}
matrix([[10, 7], [7, 17]])


         +10  7 +
   (50)  |      |
         +7   17+
                                                       Type: Matrix Integer
                                                                Time: 0 sec
sqrt(%)

   There are 2 exposed and 0 unexposed library operations named sqrt
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
```

```
                              )display op sqrt
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named sqrt
      with argument type(s)
                               Matrix Integer

-- Square root of a non-singular matrix   [Gantmacher, p. 233]
-- => [[e, (e - n) v w + e/2, (n - e) v], [0, e, 0], [0, (e - n) w, n]]
-- for arbitrary v and w with arbitrary signs e and n = +-1
matrix([[1, 1, 0], [0, 1, 0], [0, 0, 1]]);



                                                          Type: Matrix Integer
                                                                   Time: 0 sec
sqrt(%)

   There are 2 exposed and 0 unexposed library operations named sqrt
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op sqrt
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named sqrt
      with argument type(s)
                               Matrix Integer

-- Square root of a singular matrix   [Gantmacher, p. 239]
-- => [[0 a b], [0 0 0], [0 1/b 0]] for arbitrary a and b
matrix([[0, 1, 0], [0, 0, 0], [0, 0, 0]]);



                                                          Type: Matrix Integer
                                                    Time: 0.02 (IN) = 0.02 sec
sqrt(%)

   There are 2 exposed and 0 unexposed library operations named sqrt
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op sqrt
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
```

```
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named sqrt
      with argument type(s)
                              Matrix Integer

-- Singular value decomposition
-- => [1/sqrt(14)  3/sqrt(10) 1/sqrt(35) ] [2 sqrt(7) 0] [1/sqrt(2)  1/sqrt(2)]
--    [2/sqrt(14)  0           -sqrt(5/7)] [0         0] [1/sqrt(2) -1/sqrt(2)]
--    [3/sqrt(14) -1/sqrt(10) 3/sqrt(35) ] [0         0]
--    = U Sigma V^T --- singular values are [2 sqrt(7), 0]
matrix([[1, 1], [2, 2], [3, 3]])


         +1  1+
         |    |
   (53)  |2  2|
         |    |
         +3  3+
                                                         Type: Matrix Integer
                                                   Time: 0.02 (OT) = 0.02 sec
-- Jacobian of (r cos t, r sin t) => [[cos t, -r sin t], [sin t, r cos t]]
[r*cos(t), r*sin(t)]


   (54)  [r cos(t),r sin(t)]
                                                Type: List Expression Integer
                                    Time: 0.07 (IN) + 0.02 (OT) = 0.08 sec
-- Hessian of r^2 sin t => [[2 sin t, 2 r cos t], [2 r cos t, -r^2 sin t]]
r**2*sin(t)


          2
   (55)  r sin(t)
                                                     Type: Expression Integer
                                    Time: 0.08 (IN) + 0.02 (OT) = 0.10 sec
-- Wronskian of (cos t, sin t) => [[cos t, sin t], [-sin t, cos t]]
[cos(t), sin(t)]


   (56)  [cos(t),sin(t)]
                                                Type: List Expression Integer
                                                   Time: 0.02 (IN) = 0.02 sec
-- How easy is it to define functions to do the last three operations?
-- Jacobian of (r cos t, r sin t) => [[cos t, -r sin t], [sin t, r cos t]]
MYjacobian(e, v) == matrix([[D(f, x) for x in v] for f in e])
```

```
                                                                   Type: Void
                                                                   Time: 0 sec
MYjacobian([r*cos(t), r*sin(t)], [r, t])

   Compiling function MYjacobian with type (List Expression Integer,
      List OrderedVariableList [r,t]) -> Matrix Expression Integer

         +cos(t)  - r sin(t)+
   (58)  |                  |
         +sin(t)   r cos(t) +
                                               Type: Matrix Expression Integer
                           Time: 0.08 (IN) + 0.02 (EV) + 0.05 (OT) = 0.15 sec
-- Hessian of r^2 sin t => [[2 sin t, 2 r cos t], [2 r cos t, -r^2 sin t]]
MYhessian(f, x) ==
   local n
   n:= #x
   matrix([[D(f, [x . i, x . j]) for j in 1..n] for i in 1..n])

                                                                   Type: Void
                                                                   Time: 0 sec
MYhessian(r**2*sin(t), [r, t])

   Compiling function MYhessian with type (Expression Integer,List
      OrderedVariableList [r,t]) -> Matrix Expression Integer

         + 2sin(t)   2r cos(t) +
   (60)  |                     |
         |             2       |
         +2r cos(t)  - r sin(t)+
                                               Type: Matrix Expression Integer
                           Time: 0.13 (IN) + 0.03 (EV) + 0.03 (OT) = 0.20 sec
-- Wronskian of (cos t, sin t) => [[cos t, sin t], [-sin t, cos t]]
MYwronskian(f, x) ==
   local n
   n:= #f
   matrix([[D(f . j, x, i-1) for j in 1..n] for i in 1..n])

                                                                   Type: Void
                                                                   Time: 0 sec
MYwronskian([cos(t), sin(t)], t)

   Your expression cannot be fully compiled because it contains an
      integer expression (for - ) whose sign cannot be determined (in
      general) and so must be specified by you. Perhaps you can try
      substituting something like
```

```
                                (- :: PI)
                                   or
                               (- :: NNI)
      into your expression for - .
   AXIOM will attempt to step through and interpret the code.

         + cos(t)   sin(t)+
   (62)  |               |
         +- sin(t)  cos(t)+
                                              Type: Matrix Expression Integer
                                  Time: 0.15 (IN) + 0.05 (EV) = 0.20 sec
-- ---------- Quit ----------
)quit


real   160.6
user   78.2
sys    2.1
```

17 Tensor Analysis

```
-- ---------- Tensor Analysis ----------
-- Generalized Kronecker delta: delta([j, h], [i, k]) =
-- delta(j, i) delta(h, k) - delta(j, k) delta(h, i).  See David Lovelock and
-- Hanno Rund, _Tensors, Differential Forms, & Variational Principles_,  John
-- Wiley & Sons, Inc., 1975, p. 109.
delta : CartesianTensor(1, 4, Integer) := kroneckerDelta()


        +1  0  0  0+
        |          |
        |0  1  0  0|
   (1)  |          |
        |0  0  1  0|
        |          |
        +0  0  0  1+
                                          Type: CartesianTensor(1,4,Integer)
                        Time: 0.02 (EV) + 0.03 (OT) + 0.02 (GC) = 0.07 sec
delta([i, k], [j, h])

   There are no exposed library operations named delta but there is one
      unexposed operation with that name. Use HyperDoc Browse or issue
                               )display op delta
      to learn more about the available operation.
```

```
   Cannot find a definition or applicable library operation named delta
      with argument type(s)
                       List OrderedVariableList [i,k]
                       List OrderedVariableList [j,h]

)clear properties delta

-- Levi-Civita symbol: [epsilon(2,1,3), epsilon(1,3,1)] => [-1, 0]
epsilon : CartesianTensor(1, 3, Integer) := leviCivitaSymbol()


         +0   0   0+ +0  0  - 1+ + 0   1  0+
         |         | |          | |         |
   (2)  [|0   0   1|,|0  0   0 |,|- 1  0  0|]
         |         | |          | |         |
         +0  - 1  0+ +1  0   0 + + 0   0  0+
                                            Type: CartesianTensor(1,3,Integer)
                                        Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
[epsilon(2, 1, 3), epsilon(1, 3, 1)]


   (3)  [- 1,0]
                                                            Type: List Integer
                           Time: 0.02 (IN) + 0.02 (EV) + 0.02 (GC) = 0.05 sec
)clear properties epsilon

-- Tensor outer product:                   [[  5  6] [-10 -12]]
--                       [1 -2]   [ 5 6]   [[ -7  8] [ 14 -16]]
--  ij      ij           [3  4] X [-7 8] = [                  ]
-- c     = a   b                           [[ 15 18] [ 20  24]]
--    kl        kl                         [[-21 24] [-28  32]]
a:= matrix([[1, -2], [3, 4]]);


                                                           Type: Matrix Integer
               Time: 0.02 (IN) + 0.02 (EV) + 0.02 (OT) + 0.03 (GC) = 0.09 sec
b:= matrix([[5, 6], [-7, 8]]);


                                                           Type: Matrix Integer
                                                    Time: 0.02 (IN) = 0.02 sec
product(a :: CartesianTensor(1, 2, Integer), _
        b :: CartesianTensor(1, 2, Integer))
```

```
        + + 5   6+   +- 10  - 12++
        | |      |   |          ||
        | +- 7  8+   + 14   - 16+|
   (6)  |                        |
        |+ 15   18+   + 20   24+ |
        ||        |   |        | |
        ++- 21  24+   +- 28  32+ +
                                           Type: CartesianTensor(1,2,Integer)
                                       Time: 0.15 (IN) + 0.08 (OT) = 0.23 sec
)clear properties a b

-- Definition of the Christoffel symbol of the first kind (a is the metric
-- tensor) [Lovelock and Rund, p. 81]
--              d a     d a     d a
--           1     kh      hl      lk
-- Chr1    = - (----- + ----- - -----)
--     lhk   2      l       k       h
--               d x     d x     d x
-- Partial covariant derivative of a type (1, 1) tensor field (Chr2 is the
-- Christoffel symbol of the second kind) [Lovelock and Rund, p. 77]
--  i      d    i        i   m        m   i
-- T    = ---- T  + Chr2    T  - Chr2    T
--  j|k      k  j       m k  j       j k  m
--        d x
T:= operator('T');


                                                          Type: BasicOperator
                          Time: 0.08 (IN) + 0.02 (EV) + 0.02 (GC) = 0.12 sec
T([i], [j])


   (8)  T([i],[j])
                                                 Type: Expression List Symbol
              Time: 0.65 (IN) + 0.02 (EV) + 0.12 (OT) + 0.05 (GC) = 0.83 sec
-- Verify the Bianchi identity for a symmetric connection (K is the Riemann
-- curvature tensor) [Lovelock and Rund, p. 94]
--   h         h          h
-- K       + K        + K       = 0
--  i jk|l    i kl|j     i lj|k
-- ---------- Quit ----------
)quit


real   4.6
user   2.7
```

```
sys    0.2
```

18 Sums

```
-- ---------- Sums ----------
-- Simplify the sum below to sum(x[i]^2, i = 1..n) - sum(x[i], i = 1..n)^2/n
x:= operator('x);


                                                          Type: BasicOperator
                                       Time: 0.07 (IN) + 0.10 (OT) = 0.17 sec
xbar:= sum(x(j), j = 1..n) / n


         n
        --+
        >     x(j)
        --+
        j= 1
   (2)  ----------
             n
                                                     Type: Expression Integer
               Time: 0.98 (IN) + 0.41 (EV) + 0.32 (OT) + 0.15 (GC) = 1.87 sec
sum((x(i) - xbar)**2, i = 1..n)


               n        2           n
              --+                  --+           2    2
              >     x(j)  - 2n x(i)>     x(j) + n x(i)
         n    --+                  --+
        --+   j= 1                 j= 1
   (3)  >     -----------------------------------------
        --+                        2
        i= 1                      n
                                                     Type: Expression Integer
               Time: 0.18 (IN) + 0.22 (EV) + 0.07 (OT) + 0.02 (GC) = 0.48 sec
)clear properties x xbar

-- Derivation of the least squares fitting of data points (x[i], y[i]) to a
-- line y = m x + b.  See G. Keady, ``Using Maple's linalg package with Zill
-- and Cullen _Advanced Engineering Mathematics_, Part II: Vectors, Matrices
-- and Vector Calculus'', University of Western Australia,
-- ftp://maths.uwa.edu.au/pub/keady/
```

```
x:= operator(’x);


                                                          Type: BasicOperator
                                                   Time: 0.02 (EV) = 0.02 sec
y:= operator(’y);


                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
f:= sum((y(i) - m*x(i) - b)**2, i = 1..n);


                                                     Type: Expression Integer
                           Time: 0.17 (IN) + 0.15 (EV) + 0.02 (OT) = 0.33 sec
solve([D(f, m) = 0, D(f, b) = 0], [m, b])


   (7)  [[]]
                                  Type: List List Equation Expression Integer
               Time: 1.45 (IN) + 0.30 (EV) + 0.20 (OT) + 0.08 (GC) = 2.03 sec
)clear properties x y f

-- Indefinite sum => (-1)^n binomial(2 n, n).  See Herbert S, Wilf,
-- ‘‘IDENTITIES and their computer proofs’’, University of Pennsylvania.
sum((-1)**k * binomial(2*n, k)**2, k)


         k              2
        --+       %A 2n
   (8)  >    (- 1)  (  )
        --+          %A
        %A
                                                     Type: Expression Integer
               Time: 0.17 (IN) + 0.90 (EV) + 0.03 (OT) + 0.03 (GC) = 1.13 sec
-- Check whether the full Gosper algorithm is implemented
-- => 1/2^(n + 1) binomial(n, k - 1)
sum(binomial(n, k)/2**n - binomial(n + 1, k)/2**(n + 1), k)


                      n n + 1                n + 1   n
        (- n + k - 2)2 (     ) + (n - k + 2)2     (     )
                        k - 1                      k - 1
   (9)  -------------------------------------------------
                                    n n + 1
                       (n - 2k + 3)2 2
```

```
                                                     Type: Expression Integer
                           Time: 0.12 (IN) + 0.92 (EV) + 0.02 (OT) = 1.05 sec
normalize(%)


                          n!
   (10)  -----------------------------------
           (n + 1)log(2)
         %e             (k - 1)!(n - k + 1)!
                                                     Type: Expression Integer
                           Time: 0.03 (IN) + 0.40 (EV) + 0.03 (OT) = 0.47 sec
-- Dixon's identity (check whether Zeilberger's algorithm is implemented).
-- Note that the indefinite sum is equivalent to the definite
-- sum(..., k = -min(a, b, c)..min(a, b, c)) => (a + b + c)!/(a! b! c!)
-- [Wilf]
sum((-1)**k * binomial(a+b, a+k) * binomial(b+c, b+k) _
            * binomial(c+a, c+k), k)


          k
         --+       %A b + a   c + a   c + b
   (11)  >    (- 1)  (      )(      )(      )
         --+          a + %A  c + %A  b + %A
         %A
                                                     Type: Expression Integer
               Time: 0.27 (IN) + 6.12 (EV) + 0.05 (OT) + 1.02 (GC) = 7.45 sec
-- Telescoping sum => g(n + 1) - g(0)
g:= operator('g);


                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
sum(g(k + 1) - g(k), k = 0..n)


          n
         --+
   (13)  >     g(k + 1) - g(k)
         --+
         k= 0
                                                     Type: Expression Integer
                           Time: 0.25 (IN) + 0.05 (EV) + 0.03 (OT) = 0.33 sec
)clear properties g

-- => n^2 (n + 1)^2 / 4
sum(k**3, k = 1..n)
```

```
          4     3    2
         n  + 2n  + n
   (14)  -------------
               4
                                            Type: Fraction Polynomial Integer
                           Time: 0.03 (IN) + 0.12 (EV) + 0.02 (OT) = 0.17 sec
factorFraction(%)


          2       2
         n (n + 1)
   (15)  ----------
              4
                                   Type: Fraction Factored Polynomial Integer
                           Time: 0.02 (IN) + 0.08 (EV) + 0.02 (OT) = 0.12 sec
-- See Daniel I. A. Cohen, _Basic Techniques of Combinatorial Theory_, John
-- Wiley and Sons, 1978, p. 60.  The following two sums can be derived directly
-- from the binomial theorem:
-- sum(k^2 * binomial(n, k) * x^k, k = 1..n) = n x (1 + n x) (1 + x)^(n - 2)
-- => n (n + 1) 2^(n - 2)   [Cohen, p. 60]
sum(k**2 * binomial(n, k), k = 1..n)


          n
         --+    2 n
   (16)  >     k ( )
         --+      k
         k= 1
                                                     Type: Expression Integer
                           Time: 0.10 (IN) + 0.20 (EV) + 0.02 (OT) = 0.32 sec
-- => [2^(n + 1) - 1]/(n + 1)   [Cohen, p. 83]
sum(binomial(n, k)/(k + 1), k = 0..n)


                 n
                ( )
          n      k
         --+
   (17)  >     -----
         --+   k + 1
         k= 0
                                                     Type: Expression Integer
                           Time: 0.03 (IN) + 0.20 (EV) + 0.02 (OT) = 0.25 sec
-- Vandermonde's identity => binomial(n + m, r)   [Cohen, p. 31]
```

```
sum(binomial(n, k) * binomial(m, r - k), k = 0..r)


          r
         --+      m    n
   (18)  >     (     )( )
         --+    r - k  k
         k= 0
                                                     Type: Expression Integer
                           Time: 0.08 (IN) + 0.55 (EV) + 0.03 (OT) = 0.67 sec
-- => Fibonacci[2 n]   [Cohen, p. 88]
sum(binomial(n, k) * fibonacci(k), k = 0..n)

   There are 1 exposed and 0 unexposed library operations named
      fibonacci having 1 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op fibonacci
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      fibonacci with argument type(s)
                                 Variable k

-- => Fibonacci[n] Fibonacci[n + 1]   [Cohen, p. 65]
sum(fibonacci(k)**2, k = 1..n)

   There are 1 exposed and 0 unexposed library operations named
      fibonacci having 1 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op fibonacci
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      fibonacci with argument type(s)
                                 Variable k

-- => 1/2 cot(x/2) - cos([2 n + 1] x/2)/[2 sin(x/2)]
-- See Konrad Knopp, _Theory and Application of Infinite Series_, Dover
-- Publications, Inc., 1990, p. 480.
sum(sin(k*x), k = 1..n)
```

```
          n
         --+
   (19)  >     sin(k x)
         --+
         k= 1
                                                    Type: Expression Integer
                          Time: 0.07 (IN) + 0.27 (EV) + 0.03 (OT) = 0.37 sec
-- => sin(n x)^2/sin x   [Gradshteyn and Ryzhik 1.342(3)]
sum(sin((2*k - 1)*x), k = 1..n)


          n
         --+
   (20)  >     sin((2k - 1)x)
         --+
         k= 1
                                                    Type: Expression Integer
                          Time: 0.05 (IN) + 0.20 (EV) + 0.03 (OT) = 0.28 sec
-- => Fibonacci[n + 1]   [Cohen, p. 87]
sum(binomial(n - k, k), k = 0..floor(n/2))

   There are 2 exposed and 0 unexposed library operations named floor
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op floor
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named floor
      with argument type(s)
                           Polynomial Fraction Integer

-- => pi^2 / 6 + zeta(3) =~ 2.84699
limit(sum(1/k**2 + 1/k**3, k = 1..n), n = %plusInfinity)


   (21)  "failed"
                                                   Type: Union("failed",...)
                Time: 0.45 (IN) + 1.0 (EV) + 0.05 (OT) + 0.55 (GC) = 2.05 sec
-- => pi^2/12 - 1/2 (log 2)^2   [Gradshteyn and Ryzhik 0.241(2)]
limit(sum(1/(2**k*k**2), k = 1..n), n = %plusInfinity)


   (22)  "failed"
                                                   Type: Union("failed",...)
```

```
                          Time: 0.10 (IN) + 0.32 (EV) + 0.07 (OT) = 0.48 sec
-- => pi/12 sqrt(3) - 1/4 log 3   [Knopp, p. 268]
limit(sum(1/((3*k + 1)*(3*k + 2)*(3*k + 3)), k = 0..n), n = %plusInfinity)


   (23)  "failed"
                                                    Type: Union("failed",...)
                          Time: 0.10 (IN) + 0.30 (EV) + 0.02 (OT) = 0.42 sec
-- => 1/2 (2^(n - 1) + 2^(n/2) cos(n pi/4))   [Gradshteyn and Ryzhik 0.153(1)]
limit(sum(binomial(n, 4*k), k = 0..nn), nn = %plusInfinity)


   (24)  "failed"
                                                    Type: Union("failed",...)
              Time: 0.13 (IN) + 4.83 (EV) + 0.02 (OT) + 0.53 (GC) = 5.52 sec
-- => 1   [Knopp, p. 233]
limit(sum(1/(sqrt(k*(k + 1)) * (sqrt(k) + sqrt(k + 1))), k = 1..n), _
      n = %plusInfinity)


   (25)  "failed"
                                                    Type: Union("failed",...)
                          Time: 0.10 (IN) + 0.58 (EV) + 0.03 (OT) = 0.72 sec
-- => 1/sqrt([1 - x y]^2 - 4 x^2)   (| x y | < 1 and -1 <= x < 1).
--    From Evangelos A. Coutsias, Michael J. Wester and Alan S. Perelson, ``A
--    Nucleation Theory of Cell Surface Capping'', draft.
limit(sum(sum(binomial(n, k)*binomial(n - k, n - 2*k)*x**n*y**(n - 2*k), _
              k = 0..floor(n/2)), _
          n = 0..nn), nn = %plusInfinity)

   There are 2 exposed and 0 unexposed library operations named floor
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op floor
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named floor
      with argument type(s)
                        Polynomial Fraction Integer

-- An equivalent summation to the above is:
limit(limit(sum(sum(factorial(n)/(factorial(k)**2*factorial(n - 2*k))* _
                       (x/y)**k*(x*y)**(n - k), _
                    n = 2*k..nn), _
```

```
                k = 0..mm), _
            mm = %plusInfinity), _
      nn = %plusInfinity)

   There are 3 exposed and 0 unexposed library operations named limit
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op limit
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named limit
      with argument type(s)
                                   failed
                Equation OrderedCompletion Polynomial Integer

-- => pi/2   [Knopp, p. 269]
limit(sum(product(k/(2*k - 1), k = 1..m), m = 2..n), n = %plusInfinity)


   (26)  "failed"
                                                    Type: Union("failed",...)
                           Time: 0.37 (IN) + 0.33 (EV) + 0.05 (OT) = 0.75 sec
-- ---------- Quit ----------
)quit


real   210.8
user   33.1
sys    1.6
```

19 Products

```
-- ---------- Products ----------
-- => [640 pi^3]/[2187 sqrt(3)]   [Gradshteyn and Ryzhik 8.338(5)]
product(Gamma(k/3), k = 1..8)


         _ 1  _ 2  _ 4  _ 5  _ 7  _ 8
   (1)  | (-)| (-)| (-)| (-)| (-)| (-)
           3    3    3    3    3    3
                                                     Type: Expression Integer
               Time: 1.18 (IN) + 0.25 (EV) + 0.42 (OT) + 0.18 (GC) = 2.03 sec
```

```
-- => n! = gamma(n + 1)
product(k, k = 1..n)


          n
        ++-++
   (2)   | |   k
         | |
        k= 1
                                                     Type: Expression Integer
                                       Time: 0.45 (IN) + 0.07 (OT) = 0.52 sec
-- => x^[n (n + 1)/2]
product(x**k, k = 1..n)


          n
        ++-++   k
   (3)   | |   x
         | |
        k= 1
                                                     Type: Expression Integer
                           Time: 0.07 (IN) + 0.02 (EV) + 0.03 (OT) = 0.12 sec
-- => n
product((1 + 1/k), k = 1..n - 1)


        n - 1
        ++-++  k + 1
   (4)   | |   -----
         | |     k
        k= 1
                                                     Type: Expression Integer
                           Time: 0.27 (IN) + 0.03 (EV) + 0.03 (OT) = 0.33 sec
-- => 1/2^(2 n) binomial(2 n, n)   [Knopp, p. 385]
product((2*k - 1)/(2*k), k = 1..n)


          n
        ++-++  2k - 1
   (5)   | |   ------
         | |     2k
        k= 1
                                                     Type: Expression Integer
                           Time: 0.07 (IN) + 0.02 (EV) + 0.02 (OT) = 0.10 sec
-- => [x^(2 n) - 1]/(x^2 - 1)   [Gradshteyn and Ryzhik 1.396(1)]
product(x**2 - 2*x*cos(k*%pi/n) + 1, k = 1..n - 1)
```

```
        n - 1
        ++-++          k %pi     2
   (6)   | |   - 2x cos(-----) + x  + 1
         | |             n
        k= 1
                                                      Type: Expression Integer
               Time: 0.56 (IN) + 0.07 (EV) + 0.18 (OT) + 0.03 (GC) = 0.85 sec
-- => 2/3   [Knopp, p. 228]
limit(product((k**3 - 1)/(k**3 + 1), k = 2..n), n = %plusInfinity)


   (7)  "failed"
                                                     Type: Union("failed",...)
               Time: 0.45 (IN) + 0.27 (EV) + 0.09 (OT) + 0.03 (GC) = 0.83 sec
-- => 2/pi   [Gradshteyn and Ryzhik 0.262(2)]
limit(product(1 - 1/(2*k)**2, k = 1..n), n = %plusInfinity)


   (8)  "failed"
                                                     Type: Union("failed",...)
                            Time: 0.10 (IN) + 0.08 (EV) + 0.03 (OT) = 0.22 sec
-- => sqrt(2)   [Gradshteyn and Ryzhik 0.261]
limit(product(1 + (-1)**(k + 1)/(2*k - 1), k = 1..n), n = %plusInfinity)


   (9)  "failed"
                                                     Type: Union("failed",...)
                                        Time: 0.08 (IN) + 0.17 (EV) = 0.25 sec
-- => -1   [Knopp, p. 436]
limit(product((k*(k +  1) + 1 + %i)/(k*(k + 1) + 1 - %i), k = 0..n), _
      n = %plusInfinity)


   (10)  "failed"
                                                     Type: Union("failed",...)
               Time: 0.98 (IN) + 0.50 (EV) + 0.20 (OT) + 0.03 (GC) = 1.72 sec
-- ---------- Quit ----------
)quit


real   23.4
user   7.8
sys    0.5
```

20 Limits

```
-- ---------- Limits ----------
-- Start with a famous example => e
limit((1 + 1/n)**n, n = %plusInfinity)


   (1)  %e
                        Type: Union(OrderedCompletion Expression Integer,...)
               Time: 1.07 (IN) + 0.58 (EV) + 0.28 (OT) + 0.25 (GC) = 2.18 sec
-- => 1/2
limit((1 - cos(x))/x**2, x = 0)


        1
   (2)  -
        2
                        Type: Union(OrderedCompletion Expression Integer,...)
                           Time: 0.38 (IN) + 0.12 (EV) + 0.03 (OT) = 0.53 sec
-- See Dominik Gruntz, _On Computing Limits in a Symbolic Manipulation System_,
-- Ph.D. dissertation, Swiss Federal Institute of Technology, Zurich,
-- Switzerland, 1996. => 5
limit((3**x + 5**x)**(1/x), x = %plusInfinity)


   (3)  "failed"
                                                    Type: Union("failed",...)
               Time: 0.21 (IN) + 0.45 (EV) + 0.02 (OT) + 0.02 (GC) = 0.70 sec
-- => 1
limit(log(x)/(log(x) + sin(x)), x = %plusInfinity)


   (4)  "failed"
                                                    Type: Union("failed",...)
               Time: 0.05 (IN) + 1.62 (EV) + 0.03 (OT) + 0.02 (GC) = 1.72 sec
-- => - e^2   [Gruntz]
limit((exp(x*exp(-x)/(exp(-x) + exp(-2*x**2/(x + 1)))) - exp(x))/x, _
      x = %plusInfinity)


   (5)  "failed"
                                                    Type: Union("failed",...)
               Time: 0.17 (IN) + 4.52 (EV) + 0.03 (OT) + 0.52 (GC) = 5.23 sec
-- => 1/3   [Gruntz]
limit(x*log(x)*log(x*exp(x) - x**2)**2/log(log(x**2 + _
```

```
                                              2*exp(exp(3*x**3*log(x))))), _
      x = %plusInfinity)


   (6)  "failed"
                                                     Type: Union("failed",...)
                             Time: 0.17 (IN) + 2.21 (EV) + 0.03 (OT) = 2.42 sec
-- => 1/e   [Knopp, p. 73]
limit(1/n * factorial(n)**(1/n), n = %plusInfinity)


   (7)  "failed"
                                                     Type: Union("failed",...)
                             Time: 0.08 (IN) + 1.45 (EV) + 0.07 (OT) = 1.60 sec
-- Rewrite the above problem slightly => 1/e
limit(1/n * Gamma(n + 1)**(1/n), n = %plusInfinity)


   (8)  "failed"
                                                     Type: Union("failed",...)
               Time: 0.12 (IN) + 1.50 (EV) + 0.05 (OT) + 0.02 (GC) = 1.68 sec
-- => 1   [Gradshteyn and Ryzhik 8.328(2)]
limit(Gamma(z + a)/Gamma(z)*exp(-a*log(z)), z = %plusInfinity)


   (9)  "failed"
                                                     Type: Union("failed",...)
                             Time: 0.32 (IN) + 2.22 (EV) + 0.08 (OT) = 2.62 sec
-- => e^z   [Gradshteyn and Ryzhik 9.121(8)]
--limit(hypergeometric([1, k], [1], z/k), k = %plusInfinity)
-- => Euler's_constant   [Gradshteyn and Ryzhik 9.536]
--limit(zeta(x) - 1/(x - 1), x = 1)
-- => gamma(x)   [Knopp, p. 385]
limit(n**x/(x * product((1 + x/k), k = 1..n)), n = %plusInfinity)


   (10)  "failed"
                                                     Type: Union("failed",...)
               Time: 0.40 (IN) + 3.75 (EV) + 0.13 (OT) + 0.47 (GC) = 4.75 sec
-- See Angus E. Taylor and W. Robert Mann, _Advanced Calculus_, Second Edition,
-- Xerox College Publishing, 1972, p. 125 => 1
limit(x * integrate(exp(-t**2), t = 0..x)/(1 - exp(-x**2)), x = 0)


   (11)  "failed"
                                                     Type: Union("failed",...)
```

```
               Time: 0.65 (IN) + 2.63 (EV) + 0.08 (OT) + 0.05 (GC) = 3.42 sec
-- => [-1, 1]
[limit(x/abs(x), x = 0, "left"), limit(x/abs(x), x = 0, "right")]


   (12)  [- 1,1]
              Type: List Union(OrderedCompletion Expression Integer,"failed")
                           Time: 0.23 (IN) + 0.30 (EV) + 0.05 (OT) = 0.58 sec
-- => pi/2   [Richard Q. Chen]
limit(atan(-log(x)), x = 0, "right")


         %pi
   (13)  ---
          2
                        Type: Union(OrderedCompletion Expression Integer,...)
                                       Time: 0.03 (IN) + 0.72 (EV) = 0.75 sec
-- ---------- Quit ----------
)quit


real   85.9
user   28.1
sys    0.7
```

21 Calculus

```
-- ---------- Calculus ----------
-- Calculus on a non-smooth (but well defined) function => x/|x| or sign(x)
D(abs(x), x)


        abs(x)
   (1)  ------
           x
                                                      Type: Expression Integer
               Time: 0.20 (IN) + 0.30 (EV) + 0.22 (OT) + 0.20 (GC) = 0.92 sec
-- Calculus on a piecewise defined function
a(x) == if x < 0 then -x else x

                                                                    Type: Void
                                                                   Time: 0 sec
-- => if x < 0 then -1 else 1
```

```
D(a(x), x)

   Compiling function a with type Variable x -> Polynomial Integer

   (3)  1
                                                  Type: Polynomial Integer
                                       Time: 0.28 (IN) + 0.03 (OT) = 0.32 sec
)clear properties a

   Compiled code for a has been cleared.
-- Derivative of a piecewise defined function at a point [Herbert Fischer].
-- f(x) = x^2 - 1 for x = 1 otherwise x^3.  f(1) = 0 and f'(1) = 3
f(x) == if x = 1 then x**2 - 1 else x**3

                                                                Type: Void
                                                 Time: 0.02 (IN) = 0.02 sec
f(1)

   Compiling function f with type PositiveInteger -> Integer

   (5)  0
                                                  Type: NonNegativeInteger
                                       Time: 0.10 (IN) + 0.02 (EV) = 0.12 sec
D(f(x), x)

   Compiling function f with type Variable x -> Polynomial Integer

          2
   (6)  3x
                                                  Type: Polynomial Integer
                                       Time: 0.18 (IN) + 0.08 (OT) = 0.27 sec
subst(%, x = 1)


   (7)  3
                                                  Type: Expression Integer
                           Time: 0.25 (IN) + 0.03 (EV) + 0.05 (OT) = 0.33 sec
)clear properties f

   Compiled code for f has been cleared.
-- d^n/dx^n(x^n) => n!
D(x**n, x, n)

   There are 3 exposed and 0 unexposed library operations named D
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
```

```
                                )display op D
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named D
      with argument type(s)
                             Expression Integer
                                 Variable x
                                 Variable n

-- Apply the chain rule---this is important for PDEs and many other
-- applications => y_xx (x_t)^2 + y_x x_tt
x:= operator('x);


                                                          Type: BasicOperator
                                       Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
y:= operator('y);


                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
D(y(x(t)), t, 2)


          ,   2 ,,          ,       ,,
   (10)  x (t) y  (x(t)) + y (x(t))x  (t)

                                                     Type: Expression Integer
                           Time: 0.42 (IN) + 0.05 (EV) + 0.12 (OT) = 0.58 sec
)clear properties x y

-- => f(h(x)) dh/dx - f(g(x)) dg/dx
f:= operator('f);


                                                          Type: BasicOperator
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
g:= operator('g);


                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
h:= operator('h);
```

```
                                                       Type: BasicOperator
                                     Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
'integrate(f(y), y = g(x)..h(x))


   (14)  integrate(f(y),y= g(x),h(x)    )
                                                        Type: OutputForm
                                                             Time: 0 sec
D(%, x)

   There are 5 exposed and 0 unexposed library operations named D
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op D
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named D
      with argument type(s)
                                 OutputForm
                                 Variable x

)clear properties f g h

-- Exact differential => d(V(P, T)) => dV/dP DP + dV/dT DT
V:= operator('V);


                                                       Type: BasicOperator
                                     Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
D(V(P, T))

   There are 2 exposed and 0 unexposed library operations named D
      having 1 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op D
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named D
      with argument type(s)
                             Expression Integer
```

```
)clear properties V

-- Implicit differentiation => dy/dx = [1 - y sin(x y)] / [1 + x sin(x y)]
y = cos(x*y) + x


   (16)  y= cos(x y) + x
                                            Type: Equation Expression Integer
               Time: 0.22 (IN) + 0.12 (EV) + 0.05 (OT) + 0.02 (GC) = 0.40 sec
-- => 2 (x + y) g'(x^2 + y^2)
f:= operator('f);



                                                          Type: BasicOperator
                                                   Time: 0.02 (OT) = 0.02 sec
g:= operator('g);



                                                          Type: BasicOperator
                                                   Time: 0.02 (OT) = 0.02 sec
D(f(x, y), x) + D(f(x, y), y)


   (19)  f  (x,y) + f  (x,y)
          ,2         ,1
                                                     Type: Expression Integer
                           Time: 0.05 (IN) + 0.03 (EV) + 0.02 (OT) = 0.10 sec
subst(%, f(x, y) = g(x**2 + y**2))


   (20)  f  (x,y) + f  (x,y)
          ,2         ,1
                                                     Type: Expression Integer
                                       Time: 0.10 (IN) + 0.02 (OT) = 0.12 sec
)clear properties f g

-- Residue => - 9/4
--residue((z**3 + 5)/((z**4 - 1)*(z + 1)), z, -1)
-- Differential forms
DeRham:= DERHAM(Integer, [x, y, z])


   (21)  DeRhamComplex(Integer,[x,y,z])
                                                                 Type: Domain
                                                   Time: 0.02 (IN) = 0.02 sec
[dx, dy, dz]:= [generator(i)$DeRham for i in 1..3]
```

```
   (22)  [dx,dy,dz]
                                       Type: List DeRhamComplex(Integer,[x,y,z])
                                          Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
-- (2 dx + dz) /\ (3 dx + dy + dz) /\ (dx + dy + 4 dz) => 8 dx /\ dy /\ dz
(2*dx + dz) * (3*dx + dy + dz) * (dx + dy + 4*dz)


   (23)  8dx dy dz
                                            Type: DeRhamComplex(Integer,[x,y,z])
                                          Time: 0.03 (IN) + 0.02 (EV) = 0.05 sec
-- d(3 x^5 dy /\ dz + 5 x y^2 dz /\ dx + 8 z dx /\ dy)
-- => (15 x^4 + 10 x y + 8) dx /\ dy /\ dz
totalDifferential( _
   3*x**5 * dy * dz + 5*x*y**2 * dz * dx + 8*z * dx * dy :: DeRham)

   Internal Error
   The function * with signature hashcode is missing from domain
      Polynomial(DeRhamComplex (Integer) (x y z))

initial (24) ->
real   12.1
user   5.6
sys    0.4

----------------------------------------------------------------------------------

Sat Jul 11 23:35:58 MET DST 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
--------------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
--------------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on
```

```
)set quit unprotected

-- ---------- Calculus ----------
-- => 1 - 3/8 2^(1/3) = 0.5275296
--minimize(x**4 - x + 1)
-- => [0, 1]
--[minimize(1/(x**2 + y**2 + 1)), maximize(1/(x**2 + y**2 + 1))]
-- Minimize on [-1, 1] x [-1, 1]:
-- => min(a - b - c + d, a - b + c - d, a + b - c - d, a + b + c + d)
--minimize(a + b*x + c*y + d*x*y, [x = -1..1, y = -1..1])
-- => [-1, 1]
--[minimize(x**2*y**3, [x = -1..1, y = -1..1]), _
-- maximize(x**2*y**3, [x = -1..1, y = -1..1])]
-- Linear programming: minimize the objective function z subject to the
-- variables xi being non-negative along with an additional set of constraints.
-- See William R. Smythe, Jr. and Lynwood A. Johnson, _Introduction to Linear
-- Programming, with Applications_, Prentice Hall, Inc., 1966, p. 117:
-- minimize z = 4 x1 - x2 + 2 x3 - 2 x4 => {x1, x2, x3, x4}  = {2, 0, 2, 4}
-- with zmin = 4
--simplex(-(4*x1 - x2 + 2*x3 - 2*x4), [2*x1 + x2 + x3 + x4 <= 10, _
--        x1 - 2*x2 - x3 + x4 >= 4, x1 + x2 + 3*x3 - x4 >= 4])
-- ---------- Quit ----------
)quit


real   0.9
user   0.7
sys    0.1
```

22 Indefinite Integrals

```
-- ---------- Indefinite Integrals ----------
-- This integral only makes sense for x real => x |x|/2
integrate(abs(x), x)



           x
          ++
   (1)    |   abs(%I)d%I
         ++
                                           Type: Union(Expression Integer,...)
               Time: 0.33 (IN) + 0.62 (EV) + 0.13 (OT) + 0.22 (GC) = 1.30 sec
-- Calculus on a piecewise defined function
```

```
a(x) == if x < 0 then -x else x

                                                                   Type: Void
                                                                  Time: 0 sec
-- => if x < 0 then -x^2/2 else x^2/2
integrate(a(x), x)

   Compiling function a with type Variable x -> Polynomial Integer

        1  2
   (3)  - x
        2
                                            Type: Polynomial Fraction Integer
                           Time: 0.30 (IN) + 0.02 (EV) + 0.02 (OT) = 0.33 sec
)clear properties a

   Compiled code for a has been cleared.
-- This would be very difficult to do by hand
-- => 2^(1/3)/6 [1/2 log([x + 2^(1/3)]^2/[x^2 - 2^(1/3) x + 2^(2/3)])
--               + sqrt(3) arctan({[sqrt(3) x]/[2^(4/3) - x]   or
--                                 [2 x - 2^(1/3)]/[2^(1/3) sqrt(3)]})
--    [Gradshteyn and Ryzhik 2.126(1)]
1/(x**3 + 2)


           1
   (4)  ------
         3
        x  + 2
                                            Type: Fraction Polynomial Integer
                                       Time: 0.08 (IN) + 0.03 (OT) = 0.12 sec
integrate(%, x)


   (5)
          +-+     2 3+-+2    3+-+          +-+     3+-+
       - \|3 log(x  \|4  - 2x\|4  + 4) + 2\|3 log(x\|4  + 2)
     +
               +-+3+-+    +-+
             x\|3 \|4  - \|3
       6atan(----------------)
                     3
  /
       +-+3+-+
     6\|3 \|4
                                          Type: Union(Expression Integer,...)
```

```
               Time: 0.10 (IN) + 0.75 (EV) + 0.07 (OT) + 0.07 (GC) = 0.98 sec
D(%, x)


          1
   (6)  ------
         3
        x  + 2
                                                     Type: Expression Integer
                                                   Time: 0.12 (EV) = 0.12 sec
-- This integral is easy if one realizes that 4^x = (2^x)^2
-- => arcsinh(2^x)/log(2)   [Robert Israel in sci.math.symbolic]
integrate(2**x/sqrt(1 + 4**x), x)


   >> Error detected within library code:
   integrate: implementation incomplete (constant residues)

initial (7) ->
real   15.2
user   3.8
sys    0.3

-------------------------------------------------------------------------------

Mon Feb  9 04:37:53 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected
```

```
-- ---------- Indefinite Integrals ----------
-- => (-9 x^2 + 16 x - 41/5)/(2 x - 1)^(5/2)
--    [Gradshteyn and Ryzhik 2.244(8)]
integrate((3*x - 5)**2/(2*x - 1)**(7/2), x)


                 2
            - 45x  + 80x - 41
   (1)  -------------------------
              2           +------+
        (20x  - 20x + 5)\|2x - 1
                                          Type: Union(Expression Integer,...)
              Time: 0.49 (IN) + 0.67 (EV) + 0.20 (OT) + 0.20 (GC) = 1.55 sec
-- => 1/[2 m sqrt(10)] log([-5 + e^(m x) sqrt(10)]/[-5 - e^(m x) sqrt(10)])
--    [Gradshteyn and Ryzhik 2.314]
integrate(1/(2*exp(m*x) - 5*exp(-m*x)), x)


              +--+   m x 2       m x     +--+
            2\|10 (%e   )  - 20%e    + 5\|10
        log(---------------------------------)
                           m x 2
                       2(%e   )  - 5
   (2)  -------------------------------------
                             +--+
                          2m\|10
                                          Type: Union(Expression Integer,...)
              Time: 0.15 (IN) + 0.75 (EV) + 0.08 (OT) + 0.03 (GC) = 1.02 sec
-- => -3/2 x + 1/4 sinh(2 x) + tanh x   [Gradshteyn and Ryzhik 2.423(24)]
integrate(sinh(x)**4/cosh(x)**2, x)


              3            2
        sinh(x)  + (3cosh(x)  + 9)sinh(x) + (- 12x - 8)cosh(x)
   (3)  -------------------------------------------------------
                                8cosh(x)
                                          Type: Union(Expression Integer,...)
                            Time: 0.13 (IN) + 0.17 (EV) + 0.02 (OT) = 0.32 sec
simplify(%)


                2
        (cosh(x)  + 2)sinh(x) + (- 3x - 2)cosh(x)
   (4)  -----------------------------------------
                         2cosh(x)
                                                        Type: Expression Integer
```

```
                           Time: 0.03 (IN) + 0.05 (EV) + 0.02 (OT) = 0.10 sec
-- This example involves several symbolic parameters
-- => 1/sqrt(b^2 - a^2) log([sqrt(b^2 - a^2) tan(x/2) + a + b]/
--                          [sqrt(b^2 - a^2) tan(x/2) - a - b])   (a^2 < b^2)
--    [Gradshteyn and Ryzhik 2.553(3)]
integrate(1/(a + b*cos(x)), x)


   (5)
                         +-------+
                         | 2    2        2    2
        (- a cos(x) - b)\|b  - a   + (- b  + a )sin(x)
    log(----------------------------------------------)
                         b cos(x) + a
   [---------------------------------------------------,
                          +-------+
                          | 2    2
                         \|b  - a
                   +---------+
                   |   2    2
            sin(x)\|- b  + a
    2atan(---------------------)
          (b + a)cos(x) + b + a
    ----------------------------]
             +---------+
             |   2    2
            \|- b  + a
                                     Type: Union(List Expression Integer,...)
                           Time: 0.12 (IN) + 0.67 (EV) + 0.05 (OT) = 0.83 sec
map(simplify, map(f +-> D(f, x), %))


               1            1
   (6)  [------------,------------]
         b cos(x) + a b cos(x) + a
                                                Type: List Expression Integer
                                       Time: 0.32 (EV) + 0.02 (OT) = 0.33 sec
-- The integral of 1/(a + 3 cos x + 4 sin x) can have 4 different forms
-- depending on the value of a !   [Gradshteyn and Ryzhik 2.558(4)]
-- => (a = 3) 1/4 log[3 + 4 tan(x/2)]
integrate(1/(3 + 3*cos(x) + 4*sin(x)), x)


            4sin(x) + 3cos(x) + 3
        log(---------------------)
                  cos(x) + 1
```

```
   (7)  --------------------------
                     4
                                          Type: Union(Expression Integer,...)
                           Time: 0.05 (IN) + 0.20 (EV) + 0.02 (OT) = 0.27 sec
-- => (a = 4) 1/3 log([tan(x/2) + 1]/[tan(x/2) + 7])
integrate(1/(4 + 3*cos(x) + 4*sin(x)), x)


              sin(x) + 7cos(x) + 7        sin(x) + cos(x) + 1
        - log(--------------------) + log(-------------------)
                   cos(x) + 1                 cos(x) + 1
   (8)  ------------------------------------------------------
                                   3
                                          Type: Union(Expression Integer,...)
                                       Time: 0.27 (EV) + 0.02 (OT) = 0.28 sec
-- => (a = 5) -1/[2 + tan(x/2)]
integrate(1/(5 + 3*cos(x) + 4*sin(x)), x)


            - cos(x) - 1
   (9)  --------------------
        sin(x) + 2cos(x) + 2
                                          Type: Union(Expression Integer,...)
                           Time: 0.03 (IN) + 0.08 (EV) + 0.02 (OT) = 0.13 sec
-- => (a = 6) 2/sqrt(11) arctan([3 tan(x/2) + 4]/sqrt(11))
integrate(1/(6 + 3*cos(x) + 4*sin(x)), x)


                 +--+           +--+           +--+
               3\|11 sin(x) + 4\|11 cos(x) + 4\|11
         2atan(-----------------------------------)
                           11cos(x) + 11
   (10)  ------------------------------------------
                             +--+
                            \|11
                                          Type: Union(Expression Integer,...)
                                       Time: 0.22 (EV) + 0.05 (OT) = 0.27 sec
-- => x log|x^2 - a^2| - 2 x + a log|(x + a)/(x - a)|
--    [Gradshteyn and Ryzhik 2.736(1)]
integrate(log(abs(x**2 - a**2)), x)


                    2    2
   (11)  x log(abs(x  - a )) + a log(x + a) - a log(x - a) - 2x
                                          Type: Union(Expression Integer,...)
                           Time: 0.05 (IN) + 0.20 (EV) + 0.02 (OT) = 0.27 sec
```

```
-- => (a x)/2 + (pi x^2)/4 - 1/2 (x^2 + a^2) arctan(x/a)
--        [Gradshteyn and Ryzhik 2.822(4)]   or
--    (a x)/2 + 1/2 (x^2 + a^2) arccot(x/a)   [Gradshteyn and Ryzhik 2.853(2)]
integrate(x*acot(x/a), x)


             2    2        2a x
          (x  + a )atan(-------) + 2a x
                          2    2
                         x  - a
   (12)  -----------------------------
                         4
                                        Type: Union(Expression Integer,...)
               Time: 0.07 (IN) + 0.80 (EV) + 0.02 (OT) + 0.03 (GC) = 0.92 sec
-- => [sin(5 x) Ci(2 x)]/5 - [Si(7 x) + Si(3 x)]/10
--    [Gradshteyn and Ryzhik 5.31(1)]
integrate(cos(5*x)*real(Ei(%i*2*x)), x)


   >> Error detected within library code:
   ker2trigs: cannot convert kernel to gaussian function

initial (13) ->
real   30.1
user   7.6
sys    0.3

-------------------------------------------------------------------------------

Mon Feb  9 04:40:40 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off
```

```
)set messages time on

)set quit unprotected

-- ---------- Indefinite Integrals ----------
-- => 1/2 [f(x) - g(x)]/[f(x) + g(x)]   [Gradshteyn and Ryzhik 2.02(25)]
f:= operator('f);



                                                           Type: BasicOperator
                               Time: 0.03 (IN) + 0.08 (OT) + 0.03 (GC) = 0.15 sec
g:= operator('g);



                                                           Type: BasicOperator
                                                    Time: 0.02 (OT) = 0.02 sec
integrate((D(f(x), x)*g(x) - f(x)*D(g(x), x))/(f(x)**2 - g(x)**2), x)



                   ,             ,
           x f(%I)g (%I) - g(%I)f (%I)
         ++
   (3)   |   ------------------------- d%I
        ++            2        2
                 g(%I)  - f(%I)
                                         Type: Union(Expression Integer,...)
             Time: 0.72 (IN) + 0.82 (EV) + 0.20 (OT) + 0.07 (GC) = 1.80 sec
)clear properties f g

-- ---------- Quit ----------
)quit


real   9.1
user   2.5
sys    0.2
```

23 Definite Integrals

```
-- ---------- Definite Integrals ----------
-- The following two functions have a pole at a.  The first integral has a
-- principal value of zero; the second is divergent
integrate(1/(x - a), x = (a - 1)..(a + 1))
```

```
   >> Error detected within library code:
   integrate: pole in path of integration

initial (1) ->
real   5.3
user   2.5
sys    0.3

--------------------------------------------------------------------------------

Fri Jun 20 00:05:11 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Definite Integrals ----------
integrate(1/(x - a)**2, x = (a - 1)..(a + 1))


   (1)  potentialPole
                                          Type: Union(pole: potentialPole,...)
               Time: 0.63 (IN) + 0.73 (EV) + 0.17 (OT) + 0.22 (GC) = 1.75 sec
-- Different branches of the square root need to be chosen in the intervals
-- [0, 1] and [1, 2].  The correct results are 4/3, [4 - sqrt(8)]/3,
-- [8 - sqrt(8)]/3, respectively
integrate(sqrt(x + 1/x - 2), x = 0..1)


   (2)  potentialPole
```

```
                                        Type: Union(pole: potentialPole,...)
               Time: 0.13 (IN) + 0.13 (EV) + 0.02 (OT) + 0.02 (GC) = 0.30 sec
integrate(sqrt(x + 1/x - 2), x = 0..1, "noPole")


         4
   (3)  - -
         3
                     Type: Union(f1: OrderedCompletion Expression Integer,...)
               Time: 0.25 (IN) + 0.83 (EV) + 0.02 (OT) + 0.08 (GC) = 1.19 sec
integrate(sqrt(x + 1/x - 2), x = 1..2)


   (4)  potentialPole
                                        Type: Union(pole: potentialPole,...)
                           Time: 0.12 (IN) + 0.02 (EV) + 0.02 (OT) = 0.15 sec
integrate(sqrt(x + 1/x - 2), x = 1..2, "noPole")


           +-+
        - 2\|2  + 4
   (5)  -----------
             3
                     Type: Union(f1: OrderedCompletion Expression Integer,...)
                           Time: 0.07 (IN) + 0.20 (EV) + 0.03 (OT) = 0.30 sec
integrate(sqrt(x + 1/x - 2), x = 0..2)


   (6)  potentialPole
                                        Type: Union(pole: potentialPole,...)
                                     Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
integrate(sqrt(x + 1/x - 2), x = 0..2, "noPole")


           +-+
         2\|2
   (7)  - -----
           3
                     Type: Union(f1: OrderedCompletion Expression Integer,...)
                           Time: 0.05 (IN) + 0.18 (EV) + 0.02 (OT) = 0.25 sec
-- => sqrt(2)   [a modification of a problem due to W. Kahan]
integrate(sqrt(2 - 2*cos(2*x))/2, x = -3*%pi/4..-%pi/4)


   (8)  potentialPole
                                        Type: Union(pole: potentialPole,...)
```

```
                                      Time: 0.50 (IN) + 0.07 (OT) = 0.57 sec
integrate(sqrt(2 - 2*cos(2*x))/2, x = -3*%pi/4..-%pi/4, "noPole")


   (9)  0
                  Type: Union(f1: OrderedCompletion Expression Integer,...)
             Time: 0.07 (IN) + 2.90 (EV) + 0.02 (OT) + 0.10 (GC) = 3.08 sec
-- Contour integrals => pi/a e^(-a) for a > 0.  See Norman Levinson and
-- Raymond M. Redheffer, _Complex Variables_, Holden-Day, Inc., 1970, p. 198.
integrate(cos(x)/(x**2 + a**2), x = %minusInfinity..%plusInfinity)


   (10)  potentialPole
                                        Type: Union(pole: potentialPole,...)
                         Time: 0.23 (IN) + 0.03 (EV) + 0.10 (OT) = 0.37 sec
integrate(cos(x)/(x**2 + a**2), x = %minusInfinity..%plusInfinity, "noPole")


   (11)  "failed"
                                               Type: Union(fail: failed,...)
                         Time: 0.07 (IN) + 1.07 (EV) + 0.60 (GC) = 1.73 sec
-- Integrand with a branch point => pi/sin(pi a) for 0 < a < 1
-- [Levinson and Redheffer, p. 212]
integrate(t**(a - 1)/(1 + t), t = 0..%plusInfinity)


   (12)  potentialPole
                                        Type: Union(pole: potentialPole,...)
                         Time: 0.08 (IN) + 0.25 (EV) + 0.02 (GC) = 0.35 sec
integrate(t**(a - 1)/(1 + t), t = 0..%plusInfinity, "noPole")


   (13)  "failed"
                                               Type: Union(fail: failed,...)
                         Time: 0.07 (IN) + 0.62 (EV) + 0.03 (GC) = 0.72 sec
-- Integrand with a residue at infinity => -2 pi [sin(pi/5) + sin(2 pi/5)]
-- (principal value)   [Levinson and Redheffer, p. 234]
integrate(5*x**3/(1 + x + x**2 + x**3 + x**4), _
          x = %minusInfinity..%plusInfinity)


   (14)  "failed"
                                               Type: Union(fail: failed,...)
           Time: 0.08 (IN) + 46.28 (EV) + 0.05 (OT) + 3.97 (GC) = 50.38 sec
-- integrate(1/[1 + x + x^2 + ... + x^(2 n)], x = -infinity..infinity)
-- = 2 pi/(2 n + 1) [1 + cos(pi/[2 n + 1])] csc(2 pi/[2 n + 1])
```

```
-- [Levinson and Redheffer, p. 255] => 2 pi/5 [1 + cos(pi/5)] csc(2 pi/5)
integrate(1/(1 + x + x**2 + x**4), x = %minusInfinity..%plusInfinity)

/usr/local/bin/axiomb: 21068 Terminated

--------------------------------------------------------------------------------

Mon Jun 16 06:14:26 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-------------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-------------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Definite Integrals ----------
-- Integrand with a residue at infinity and a branch cut => pi [sqrt(2) - 1]
-- [Levinson and Redheffer, p. 234]
integrate(sqrt(1 - x**2)/(1 + x**2), x = -1..1)


   (1)  potentialPole
                                        Type: Union(pole: potentialPole,...)
               Time: 0.55 (IN) + 0.37 (EV) + 0.20 (OT) + 0.18 (GC) = 1.30 sec
integrate(sqrt(1 - x**2)/(1 + x**2), x = -1..1, "noPole")


          +-+       2       +-+       1
   (2)  2\|2 atan(----) + 2\|2 atan(----) - %pi
                   +-+               +-+
                  \|2               \|2
                    Type: Union(f1: OrderedCompletion Expression Integer,...)
               Time: 0.20 (IN) + 2.62 (EV) + 0.05 (OT) + 0.18 (GC) = 3.05 sec
```

```
-- This is a common integral in many physics calculations
-- => q/p sqrt(pi/p) e^(q^2/p)   (Re p > 0)   [Gradshteyn and Ryzhik 3.462(6)]
integrate(x*exp(-p*x**2 + 2*q*x), x = %minusInfinity..%plusInfinity)


   (3)  "failed"
                                                Type: Union(fail: failed,...)
               Time: 0.32 (IN) + 0.41 (EV) + 0.05 (OT) + 0.03 (GC) = 0.82 sec
-- => 2 Euler's_constant   [Gradshteyn and Ryzhik 8.367(5-6)]
integrate(1/log(t) + 1/(1 - t) - log(log(1/t)), t = 0..1)


   (4)  potentialPole
                                         Type: Union(pole: potentialPole,...)
                           Time: 0.37 (IN) + 0.02 (EV) + 0.07 (OT) = 0.45 sec
integrate(1/log(t) + 1/(1 - t) - log(log(1/t)), t = 0..1, "noPole")


   (5)  "failed"
                                                Type: Union(fail: failed,...)
                           Time: 0.12 (IN) + 0.75 (EV) + 0.07 (GC) = 0.93 sec
-- This integral comes from atomic collision theory => 0   [John Prentice]
integrate(sin(t)/t*exp(2*%i*t), t = %minusInfinity..%plusInfinity)


   >> Error detected within library code:
   integrate: pole in path of integration

initial (6) ->
real   18.4
user   8.2
sys    0.6

-------------------------------------------------------------------------------

Mon Jun 16 06:19:54 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
```

```
-------------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Definite Integrals ----------
-- => 1/12   [Gradshteyn and Ryzhik 6.443(3)]
integrate(log(Gamma(x))*cos(6*%pi*x), x = 0..1)


   (1)  potentialPole
                                          Type: Union(pole: potentialPole,...)
               Time: 0.67 (IN) + 0.48 (EV) + 0.32 (OT) + 0.20 (GC) = 1.67 sec
integrate(log(Gamma(x))*cos(6*%pi*x), x = 0..1, "noPole")


   (2)  "failed"
                                                 Type: Union(fail: failed,...)
               Time: 0.22 (IN) + 1.67 (EV) + 0.02 (OT) + 0.12 (GC) = 2.02 sec
-- => 36/35   [Gradshteyn and Ryzhik 7.222(2)]
integrate((1 + x)**3*legendreP(1, x)*legendreP(2, x), x = -1..1)


        36
   (3)  --
        35
                    Type: Union(f1: OrderedCompletion Expression Integer,...)
               Time: 0.92 (IN) + 0.40 (EV) + 0.12 (OT) + 0.03 (GC) = 1.47 sec
-- => 1/sqrt(a^2 + b^2)   (a > 0 and b real)
--    [Gradshteyn and Ryzhik 6.611(1)]
integrate(exp(-a*x)*besselJ(0, b*x), x = 0..%plusInfinity)


   (4)  potentialPole
                                          Type: Union(pole: potentialPole,...)
                           Time: 0.37 (IN) + 0.23 (EV) + 0.07 (OT) = 0.67 sec
integrate(exp(-a*x)*besselJ(0, b*x), x = 0..%plusInfinity, "noPole")


   (5)  "failed"
                                                 Type: Union(fail: failed,...)
```

```
                          Time: 0.05 (IN) + 0.37 (EV) + 0.03 (OT) = 0.45 sec
-- Integrand contains a special function => 4/(3 pi)   [Tom Hagstrom]
integrate((besselJ(1, x)/x)**2, x = 0..%plusInfinity)


   (6)  potentialPole
                                        Type: Union(pole: potentialPole,...)
              Time: 0.12 (IN) + 0.22 (EV) + 0.05 (OT) + 0.02 (GC) = 0.40 sec
integrate((besselJ(1, x)/x)**2, x = 0..%plusInfinity, "noPole")


   (7)  "failed"
                                               Type: Union(fail: failed,...)
                          Time: 0.03 (IN) + 0.25 (EV) + 0.02 (OT) = 0.30 sec
-- => (cos 7 - 1)/7   [Gradshteyn and Ryzhik 6.782(3)]
integrate(real(Ei(%i*x))*besselJ(0, 2*sqrt(7*x)), x = 0..%plusInfinity)


   >> Error detected within library code:
   ker2trigs: cannot convert kernel to gaussian function

initial (8) ->
real   16.2
user   8.1
sys    0.5

--------------------------------------------------------------------------------

Mon Jun 16 06:38:04 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on
```

```
)set quit unprotected

-- ---------- Definite Integrals ----------
-- This integral comes from doing a two loop Feynman diagram for a QCD problem
-- => - [17/3 + pi^2]/36 + log 2/9 [35/3 - pi^2/2 - 4 log 2 + log(2)^2]
--    + zeta(3)/4 = 0.210883...   [Rolf Mertig]
--integrate(x**2*polylog(3, 1/(x + 1)), x = 0..1)
--integrate(x**2*polylog(3, 1/(x + 1)), x = 0..1) :: Complex Float
-- - (17/3 + %pi**2)/36 + log(2)/9*(35/3 - %pi**2/2 - 4*log(2) + log(2)**2)
--    + zeta(3)/4 :: Complex Float
-- Integrate a piecewise defined step function s(t) multiplied by cos t, where
-- s(t) = 0   (t < 1);   1   (1 <= t <= 2);   0   (t > 2)
-- => 0   (u < 1);   sin u - sin 1   (1 <= u <= 2);   sin 2 - sin 1   (u > 2)
s(t) == if 1 <= t and t <= 2 then 1 else 0;


                                                                   Type: Void
                                                                  Time: 0 sec
integrate(s(t)*cos(t), t = 0..u)

   Compiling function s with type Variable t -> NonNegativeInteger

   (2)  0
                 Type: Union(f1: OrderedCompletion Expression Integer,...)
            Time: 0.88 (IN) + 0.58 (EV) + 0.37 (OT) + 0.33 (GC) = 2.17 sec
)clear properties s

   Compiled code for s has been cleared.
-- Integrating first with respect to y and then x is much easier than
-- integrating first with respect to x and then y
-- => (|b| - |a|) pi   [W. Kahan]
integrate(integrate(x/(x**2 + y**2), y = %minusInfinity..%plusInfinity), _
          x = a..b)

   There are 11 exposed and 7 unexposed library operations named
      integrate having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Integer
                            SegmentBinding Symbol
```

```
integrate(integrate(x/(x**2 + y**2), y = %minusInfinity..%plusInfinity, _
          "noPole"), x = a..b, "noPole")


   (3)  "failed"
                                                Type: Union(fail: failed,...)
               Time: 0.72 (IN) + 1.10 (EV) + 0.12 (OT) + 0.12 (GC) = 2.05 sec
integrate(integrate(x/(x**2 + y**2), x = a..b), _
          y = %minusInfinity..%plusInfinity)

   There are 11 exposed and 7 unexposed library operations named
      integrate having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Integer
                  SegmentBinding OrderedCompletion Integer

integrate(integrate(x/(x**2 + y**2), x = a..b, "noPole"), _
          y = %minusInfinity..%plusInfinity, "noPole")



           2        1     2        1
   (4)  - a %pi abs(-) + b %pi abs(-)
                    a              b
                    Type: Union(f1: OrderedCompletion Expression Integer,...)
               Time: 0.10 (IN) + 2.18 (EV) + 0.05 (OT) + 0.05 (GC) = 2.38 sec
-- => [log(sqrt(2) + 1) + sqrt(2)]/3   [Caviness et all, section 2.10.1]
integrate(integrate(sqrt(x**2 + y**2), x = 0..1), y = 0..1)

   There are 11 exposed and 7 unexposed library operations named
      integrate having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
```

```
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Intege
                      SegmentBinding NonNegativeInteger

integrate(integrate(sqrt(x**2 + y**2), x = 0..1, "noPole"), y = 0..1, "noPole")


   (5)
        +-+             +-+             +-+              +-+            +-+
   (- 4\|2  + 6)log(- 2\|2  + 3) + (- 2\|2  + 3)log(- 12\|2  + 17) - 24\|2  + 32
   -----------------------------------------------------------------------------
                                       +-+
                                    48\|2  - 72
                    Type: Union(f1: OrderedCompletion Expression Integer,...)
               Time: 0.10 (IN) + 6.17 (EV) + 0.03 (OT) + 0.03 (GC) = 6.33 sec
-- => (pi a)/2   [Gradshteyn and Ryzhik 4.621(1)]
integrate(integrate(sin(a)*sin(y)/sqrt(1 - sin(a)**2*sin(x)**2*sin(y)**2), _
                    x = 0..%pi/2), _
          y = 0..%pi/2)

   There are 11 exposed and 7 unexposed library operations named
      integrate having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Intege
                              SegmentBinding Pi

integrate(integrate(sin(a)*sin(y)/sqrt(1 - sin(a)**2*sin(x)**2*sin(y)**2), _
                    x = 0..%pi/2, "noPole"), _
          y = 0..%pi/2, "noPole")

   There are 4 exposed and 1 unexposed library operations named
      integrate having 3 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
```

```
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Intege
                              SegmentBinding Pi
                                   String

-- => 46/15   [Paul Zimmermann]
integrate(integrate(abs(y - x**2), y = 0..2), x = -1..1)

   There are 11 exposed and 7 unexposed library operations named
      integrate having 2 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Intege
                           SegmentBinding Integer

integrate(integrate(abs(y - x**2), y = 0..2, "noPole"), x = -1..1, "noPole")

   There are 4 exposed and 1 unexposed library operations named
      integrate having 3 argument(s) but none was determined to be
      applicable. Use HyperDoc Browse, or issue
                            )display op integrate
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      integrate with argument type(s)
Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Expression Intege
                           SegmentBinding Integer
                                   String

-- Multiple integrals: volume of a tetrahedron => a b c / 6
integrate(integrate(integrate(1, z = 0..c*(1 - x/a - y/b)), _
                    y = 0..b*(1 - x/a)), _
          x = 0..a)


        a b c
   (6)  -----
          6
                    Type: Union(f1: OrderedCompletion Expression Integer,...)
```

```
               Time: 1.73 (IN) + 0.55 (EV) + 0.40 (OT) + 0.02 (GC) = 2.70 sec
-- ---------- Quit ----------
)quit


real   47.7
user   23.7
sys    0.7
```

24 Series

```
-- ---------- Series ----------
-- Taylor series---this first example comes from special relativity
-- => 1 + 1/2 (v/c)^2 + 3/8 (v/c)^4 + 5/16 (v/c)^6 + O((v/c)^8)
1/sqrt(1 - (v/c)**2)


              1
   (1)  ------------
         +---------+
         |   2    2
         |- v  + c
         |---------
         |     2
        \|    c
                                                     Type: Expression Integer
               Time: 0.43 (IN) + 0.18 (EV) + 0.22 (OT) + 0.22 (GC) = 1.05 sec
series(%, v = 0)


             1   2    3   4     5   6      8
   (2)  1 + --- v  + --- v  + ---- v  + O(v )
              2         4         6
            2c        8c       16c
                        Type: UnivariatePuiseuxSeries(Expression Integer,v,0)
               Time: 0.38 (IN) + 0.27 (EV) + 0.12 (OT) + 0.05 (GC) = 0.82 sec
1/%**2


             1  2      8
   (3)  1 - -- v  + O(v )
             2
            c
                        Type: UnivariatePuiseuxSeries(Expression Integer,v,0)
```

```
                                       Time: 0.27 (IN) + 0.10 (OT) = 0.37 sec
-- Note: sin(x) = x - x^3/6 + x^5/120 - x^7/5040 + O(x^9)
--       cos(x) = 1 - x^2/2 + x^4/24 - x^6/720 + O(x^8)
--       tan(x) = x + x^3/3 + 2/15 x^5 + 17/315 x^7 + O(x^9)
tsin:= series(sin(x), x = 0)


            1  3    1   5     1   7      9
   (4)  x - - x  + --- x  - ---- x  + O(x )
            6      120      5040
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
               Time: 0.12 (IN) + 0.18 (EV) + 0.05 (OT) + 0.03 (GC) = 0.38 sec
tcos:= series(cos(x), x = 0)


            1  2    1  4    1   6      8
   (5)  1 - - x  + -- x  - --- x  + O(x )
            2      24      720
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                                       Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
-- Note that additional terms will be computed as needed
tsin/tcos


            1  3    2  5    17  7      9
   (6)  x + - x  + -- x  + --- x  + O(x )
            3      15      315
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                                       Time: 0.08 (IN) + 0.03 (OT) = 0.12 sec
series(tan(x), x = 0)


            1  3    2  5    17  7      9
   (7)  x + - x  + -- x  + --- x  + O(x )
            3      15      315
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                           Time: 0.02 (IN) + 0.02 (EV) + 0.03 (OT) = 0.07 sec
)clear properties tsin tcos

-- => -x^2/6 - x^4/180 - x^6/2835 - O(x^8)
series(log(sin(x)/x), x = 0)


          1  2    1   4     1   6     1    8      10
   (8)  - - x  - --- x  - ---- x  - ----- x  + O(x  )
          6      180      2835      37800
```

```
                         Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                            Time: 0.02 (IN) + 0.05 (EV) + 0.05 (OT) = 0.12 sec
series(sin(x)/x, x = 0, 7)


            1  2    1   4     1   6      8
   (9)  1 - - x  + --- x  - ---- x  + O(x )
            6      120      5040
                         Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                            Time: 0.05 (IN) + 0.03 (EV) + 0.02 (OT) = 0.10 sec
log(%)


           1  2    1   4     1   6     1    8      10
   (10)  - - x  - --- x  - ---- x  - ----- x  + O(x  )
           6      180      2835      37800
                         Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                                        Time: 0.02 (EV) + 0.02 (OT) = 0.03 sec
-- => [a f'(a d) + g(b d) + integrate(h(c y), y = 0..d)]
--    + [a^2 f''(a d) + b g'(b d) + h(c d)] (x - d)
f:= operator('f);


                                                           Type: BasicOperator
                                                    Time: 0.08 (IN) = 0.08 sec
g:= operator('g);


                                                           Type: BasicOperator
                                                    Time: 0.03 (IN) = 0.03 sec
h:= operator('h);


                                                           Type: BasicOperator
                                                    Time: 0.05 (IN) = 0.05 sec
D(f(a*x), x) + g(b*x) + 'integrate(h(c*y), y = 0..x)



           ,
   (14)  af (a x) + g(b x) + integrate(h(c y),y= 0,x    )

                                                              Type: OutputForm
                            Time: 0.57 (IN) + 0.05 (EV) + 0.20 (OT) = 0.82 sec
series(%, x = d, 1)

   There are 3 exposed and 0 unexposed library operations named series
```

```
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op series
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      series with argument type(s)
                                 OutputForm
                               Equation Symbol
                               PositiveInteger

)clear properties f g h

-- Taylor series of nonscalar objects (noncommutative multiplication)
-- => (B A - A B) t^2/2 + O(t^3)   [Stanly Steinberg]
A : SquareMatrix(2, Integer);

                                                                Type: Void
                                                Time: 0.02 (IN) = 0.02 sec
B : SquareMatrix(2, Integer);

                                                                Type: Void
                                                               Time: 0 sec
%e**((A + B)*t) - %e**(A*t) * %e**(B*t)


   A is declared as being in SquareMatrix(2,Integer) but has not been
      given a value.
)clear properties A B

series(%e**((A + B)*t) - %e**(A*t) * %e**(B*t), t = 0, 4)


            15
   (17)  O(t  )
                        Type: UnivariatePuiseuxSeries(Expression Integer,t,0)
                           Time: 0.20 (IN) + 0.23 (EV) + 0.28 (OT) = 0.72 sec
-- Laurent series:
-- => sum( Bernoulli[k]/k! x^(k - 2), k = 1..infinity )
--    = 1/x^2 - 1/(2 x) + 1/12 - x^2/720 + x^4/30240 + O(x^6)
--    [Levinson and Redheffer, p. 173]
series(1/(x*(exp(x) - 1)), x = 0)
```

```
          - 2   1  - 1    1    1   2     1    4      6
   (18)  x    - - x    + -- - --- x  + ----- x  + O(x )
                2        12   720      30240
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                           Time: 0.07 (IN) + 0.05 (EV) + 0.05 (OT) = 0.17 sec
-- Puiseux series (terms with fractional degree):
-- => 1/sqrt(x - 3/2 pi) + (x - 3/2 pi)^(3/2) / 12 + O([x - 3/2 pi]^(7/2))
series(sqrt(sec(x)), x = 3/2*%pi, 2)


                     1                3               7
                   - -                -               -
              3%pi   2    1      3%pi 2          3%pi 2
   (19)  (x - ----)    + -- (x - ----)  + O((x - ----) )
                2        12        2               2
                 Type: UnivariatePuiseuxSeries(Expression Integer,x,(3*pi)/2)
               Time: 0.20 (IN) + 0.25 (EV) + 0.11 (OT) + 0.02 (GC) = 0.58 sec
-- Generalized Taylor series => sum( [x log x]^k/k!, k = 0..infinity )
series(x**x, x = 0)


   (20)
                         2            3            4            5
                   log(x)   2   log(x)   3   log(x)   4   log(x)   5
     1 + log(x)x + ------- x  + ------- x  + ------- x  + ------- x
                      2            6            24          120
   +
           6            7
     log(x)   6   log(x)   7      8
     ------- x  + ------- x  + O(x )
       720          5040
                   Type: GeneralUnivariatePowerSeries(Expression Integer,x,0)
               Time: 0.03 (IN) + 0.07 (EV) + 0.07 (OT) + 0.02 (GC) = 0.18 sec
-- Compare the generalized Taylor series of two different formulations of a
-- function => log(z) + log(cosh(w)) + tanh(w) z + O(z^2)
)set streams calculate 1

s1:= series(log(sinh(z)) + log(cosh(z + w)), z = 0)


                w 2                    w 2
             (%e )  + 1             (%e )  - 1        2
   (21)  log(----------) + log(z) + ---------- z + O(z )
                   w                   w 2
                2%e                 (%e )  + 1
                   Type: GeneralUnivariatePowerSeries(Expression Integer,z,0)
```

```
                           Time: 0.08 (IN) + 0.53 (EV) + 0.07 (OT) = 0.68 sec
s2:= series(log(sinh(z) * cosh(z + w)), z = 0)


                w 2                    w 2
             (%e )  + 1             (%e )  - 1        2
   (22)  log(----------) + log(z) + ---------- z + O(z )
                   w                   w 2
                2%e                 (%e )  + 1
                   Type: GeneralUnivariatePowerSeries(Expression Integer,z,0)
                                       Time: 0.53 (EV) + 0.07 (OT) = 0.60 sec
s1 - s2


            3
   (23)  O(z )
                   Type: GeneralUnivariatePowerSeries(Expression Integer,z,0)
                                       Time: 0.02 (IN) + 0.07 (OT) = 0.08 sec
)clear properties s1 s2

)set streams calculate 7

-- Look at the generalized Taylor series around x = 1
-- => (x - 1)^a/e^b [1 - (a + 2 b) (x - 1) / 2 + O((x - 1)^2)]
log(x)**a*exp(-b*x)


           - b x      a
   (24)  %e     log(x)
                                                     Type: Expression Integer
                           Time: 0.07 (IN) + 0.02 (EV) + 0.02 (OT) = 0.10 sec
series(%, x = 1, 1)


   >> Error detected within library code:
   No series expansion

initial (25) ->
real   29.4
user   9.7
sys    0.5

--------------------------------------------------------------------------------

Tue Aug 19 07:01:32 MET DST 1997
anne
```

```
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

)set streams calculate 7

-- ---------- Series ----------
-- Asymptotic expansions => sqrt(2) x + O(1/x)
series(sqrt(2*x**2 + 1), x = %plusInfinity, 0)

   There are 3 exposed and 0 unexposed library operations named series
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op series
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      series with argument type(s)
                            Expression Integer
                Equation OrderedCompletion Polynomial Integer
                            NonNegativeInteger

-- Wallis' product => 1/sqrt(pi n) + ...   [Knopp, p. 385]
series(1/2**(2*n) * binomial(2*n, n), n = %plusInfinity, 0)

   There are 3 exposed and 0 unexposed library operations named series
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op series
```

```
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      series with argument type(s)
                             Expression Integer
                Equation OrderedCompletion Polynomial Integer
                             NonNegativeInteger

-- => 0!/x - 1!/x^2 + 2!/x^3 - 3!/x^4 + O(1/x^5)   [Knopp, p. 544]
exp(x) * 'integrate(exp(-t)/t, t = x..%plusInfinity)


          x          exp(- t)
   (1)  %e integrate(--------,t= x,%plusInfinity    )
                         t
                                                             Type: OutputForm
               Time: 0.27 (IN) + 0.12 (EV) + 0.03 (OT) + 0.02 (GC) = 0.43 sec
series(%, x = %plusInfinity, 5)

   There are 3 exposed and 0 unexposed library operations named series
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op series
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      series with argument type(s)
                                 OutputForm
                Equation OrderedCompletion Polynomial Integer
                               PositiveInteger

-- Multivariate Taylor series expansion => 1 - (x^2 + 2 x y + y^2)/2 + O(x^4)
)set streams calculate 2

series(cos(x + y), y = 0)


                           cos(x)  2      3
   (2)  cos(x) - sin(x)y - ------ y  + O(y )
                                2
                        Type: UnivariatePuiseuxSeries(Expression Integer,y,0)
               Time: 0.32 (IN) + 0.22 (EV) + 0.10 (OT) + 0.10 (GC) = 0.73 sec
```

```
series(%, x = 0)

   There are 4 exposed and 0 unexposed library operations named series
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                             )display op series
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      series with argument type(s)
                                     Any
                         Equation Polynomial Integer

)set streams calculate 7

-- Power series (compute the general formula)
log(sin(x)/x)


            sin(x)
   (3)  log(------)
               x
                                                     Type: Expression Integer
                                       Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
exp(-x)*sin(x)


          - x
   (4)  %e   sin(x)
                                                     Type: Expression Integer
                                       Time: 0.03 (IN) + 0.02 (EV) = 0.05 sec
series(%, x = 0)


             2   1  3    1  5    1  6    1   7      9
   (5)  x - x  + - x  - -- x  + -- x  - --- x  + O(x )
                 3      30      90      630
                        Type: UnivariatePuiseuxSeries(Expression Integer,x,0)
                           Time: 0.03 (IN) + 0.08 (EV) + 0.05 (OT) = 0.17 sec
-- Derive an explicit Taylor series solution of y as a function of x from the
-- following implicit relation:
-- y = x - 1 + (x - 1)^2/2 + 2/3 (x - 1)^3 + (x - 1)^4 + 17/10 (x - 1)^5 + ...
y:= operator('y);
```

```
                                                       Type: BasicOperator
                                                Time: 0.08 (IN) = 0.08 sec
x = sin(y(x)) + cos(y(x))


   (7)  x= sin(y(x)) + cos(y(x))
                                           Type: Equation Expression Integer
                           Time: 0.47 (IN) + 0.02 (EV) + 0.14 (OT) = 0.62 sec
seriesSolve(%, y, x = 1, 0)


   >> Error detected within library code:
   Improper initial value

initial (8) ->
real   27.3
user   9.0
sys    0.6

--------------------------------------------------------------------------------

Tue Aug 19 07:21:02 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

)set streams calculate 7

-- ---------- Series ----------
```

```
-- Pade (rational function) approximation => (2 - x)/(2 + x)
pade(1, 1, series(exp(-x), x = 0))

   Compiling function G82130 with type Integer -> Boolean

        - x + 2
   (1)  -------
         x + 2
         Type: Union(Fraction UnivariatePolynomial(x,Expression Integer),...)
                Time: 0.69 (IN) + 0.67 (EV) + 0.28 (OT) + 0.37 (GC) = 2.0 sec
-- Fourier series of f(x) of period 2 p over the interval [-p, p]
-- => - (2 p / pi) sum( (-1)^n sin(n pi x / p) / n, n = 1..infinity )
x


   (2)  x
                                                              Type: Variable x
                                                                   Time: 0 sec
-- => p / 2
-- - (2 p / pi^2) sum( [1 - (-1)^n] cos(n pi x / p) / n^2, n = 1..infinity )
abs(x)


   (3)  abs(x)
                                                      Type: Expression Integer
                           Time: 0.12 (IN) + 0.03 (EV) + 0.02 (OT) = 0.17 sec
-- ---------- Quit ----------
)quit


real   7.4
user   2.6
sys    0.3
```

25 Transforms

```
-- ---------- Transforms ----------
-- Laplace and inverse Laplace transforms
-- => s/[s^2 + (w - 1)^2]   (Re s > |Im(w - 1)|)
--    [Gradshteyn and Ryzhik 17.13(33)]
laplace(cos((w - 1)*t), t, s)


                s
```

```
   (1)  ----------------
         2         2
        w  - 2w + s  + 1
                                                       Type: Expression Integer
              Time: 0.37 (IN) + 1.22 (EV) + 0.15 (OT) + 0.18 (GC) = 1.92 sec
inverseLaplace(%, s, t)


             +-----------+
             | 2
   (2)  cos(t\|w  - 2w + 1 )
                                            Type: Union(Expression Integer,...)
                        Time: 0.02 (IN) + 0.08 (EV) + 0.02 (OT) = 0.12 sec
-- => w/(s^2 - 4 w^2)   (Re s > |Re w|)   [Gradshteyn and Ryzhik 17.13(84)]
laplace(sinh(w*t)*cosh(w*t), t, s)


             w
   (3)  - --------
            2    2
          4w  - s
                                                       Type: Expression Integer
                        Time: 0.22 (IN) + 0.07 (EV) + 0.03 (OT) = 0.32 sec
-- e^(-6 sqrt(s))/s   (Re s > 0)   [Gradshteyn and Ryzhik 17.13(102)]
laplace(erf(3/sqrt(t)), t, s)


   >> Error detected within library code:
   Sorry - cannot handle that integrand yet

initial (4) ->
real   10.3
user   3.5
sys    0.3

-------------------------------------------------------------------------------

Fri Jun 13 03:43:05 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
```

```
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Transforms ----------
-- Solve y'' + y = 4 [H(t - 1) - H(t - 2)], y(0) = 1, y'(0) = 0 where H is the
-- Heaviside (unit step) function (the RHS describes a pulse of magnitude 4 and
-- duration 1).  See David A. Sanchez, Richard C. Allen, Jr. and Walter T.
-- Kyner, _Differential Equations: An Introduction_, Addison-Wesley Publishing
-- Company, 1983, p. 211.  First, take the Laplace transform of the ODE
-- => s^2 Y(s) - s + Y(s) = 4/s [e^(-s) - e^(-2 s)]
-- where Y(s) is the Laplace transform of y(t)
y:= operator('y);


                                                          Type: BasicOperator
                           Time: 0.08 (IN) + 0.10 (OT) + 0.02 (GC) = 0.20 sec
heaviside:= operator('heaviside);


                                                          Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
map(e +-> laplace(e, t, s), _
    D(y(t), t, 2) + y(t) = 4*(heaviside(t - 1) - heaviside(t - 2)))


   (3)
     2                          ,
   (s  + 1)laplace(y(t),t,s) - y (0) - y(0)s =

     4laplace(heaviside(t - 1),t,s) - 4laplace(heaviside(t - 2),t,s)
                                            Type: Equation Expression Integer
                Time: 1.0 (IN) + 1.17 (EV) + 0.31 (OT) + 0.27 (GC) = 2.75 sec
-- Now, solve for Y(s) and then take the inverse Laplace transform
-- => Y(s) = s/(s^2 + 1) + 4 [1/s - s/(s^2 + 1)] [e^(-s) - e^(-2 s)]
-- => y(t) = cos t + 4 {[1 - cos(t - 1)] H(t - 1) - [1 - cos(t - 2)] H(t - 2)}
-- What is the Laplace transform of an infinite square wave?
-- => 1/s + 2 sum( (-1)^n e^(- s n a)/s, n = 1..infinity )
```

```
--    [Sanchez, Allen and Kyner, p. 213]
laplace(1 + 2*limit(sum((-1)**n*heaviside(t - n*a), n = 1..N), _
                    N = %plusInfinity), t, s)

   There are 31 exposed and 17 unexposed library operations named *
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                               )display op *
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named *
      with argument type(s)
                              PositiveInteger
                                  failed

laplace(1 + 2*'limit(sum((-1)**n*heaviside(t - n*a), n = 1..N), _
                     N = %plusInfinity), t, s)

   There are 1 exposed and 0 unexposed library operations named laplace
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                            )display op laplace
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      laplace with argument type(s)
                                OutputForm
                                Variable t
                                Variable s

-- Fourier transforms => sqrt(2 pi) delta(z)   [Gradshteyn and Ryzhik 17.23(1)]
FourierTransform(f, x, z) == _
   [integrate(1/sqrt(2*%pi) * f * %e**(%i*z*x), _
              x = %minusInfinity..%plusInfinity), _
    integrate(1/sqrt(2*%pi) * f * %e**(%i*z*x), _
              x = %minusInfinity..%plusInfinity, "noPole")];

                                                                  Type: Void
                                                                 Time: 0 sec
FourierTransform(1, x, z)

   Compiling function FourierTransform with type (PositiveInteger,
```

```
      Variable x,Variable z) -> List Union(f1: OrderedCompletion
      Expression Complex Integer,f2: List OrderedCompletion Expression
      Complex Integer,fail: failed,pole: potentialPole)

   (5)  ["failed","failed"]
Type: List Union(f1: OrderedCompletion Expression Complex Integer,f2: List OrderedCompletio
               Time: 2.50 (IN) + 2.65 (EV) + 0.38 (OT) + 0.67 (GC) = 6.20 sec
-- => e^(-z^2/36) / [3 sqrt(2)]   [Gradshteyn and Ryzhik 17.23(13)]
FourierTransform(exp(-9*x**2), x, z)

   Compiling function FourierTransform with type (Expression Integer,
      Variable x,Variable z) -> List Union(f1: OrderedCompletion
      Expression Complex Integer,f2: List OrderedCompletion Expression
      Complex Integer,fail: failed,pole: potentialPole)

   (6)  ["failed","failed"]
Type: List Union(f1: OrderedCompletion Expression Complex Integer,f2: List OrderedCompletio
               Time: 0.37 (IN) + 1.42 (EV) + 0.18 (OT) + 0.02 (GC) = 1.99 sec
-- => sqrt(2 / pi) (9 - z^2)/(9 + z^2)^2   [Gradshteyn and Ryzhik 17.23(11)]
FourierTransform(abs(x)*exp(-3*abs(x)), x, z)


   >> Error detected within library code:
   Function not supported by Risch d.e.

initial (7) ->
real   44.4
user   14.3
sys    0.7

-------------------------------------------------------------------------------

Fri Jun 13 04:04:37 MET DST 1997
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
```

```
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Transforms ----------
-- Mellin transforms
-- => pi cot(pi s)   (0 < Re s < 1)   [Gradshteyn and Ryzhik 17.43(5)]
MellinTransform(f, x, s) == _
   [integrate(f * x**(s - 1), x = 0..%plusInfinity), _
    integrate(f * x**(s - 1), x = 0..%plusInfinity, "noPole")];

                                                                   Type: Void
                                                   Time: 0.02 (IN) = 0.02 sec
MellinTransform(1/(1 - x), x, s)

   Compiling function MellinTransform with type (Fraction Polynomial
      Integer,Variable x,Variable s) -> List Union(f1:
      OrderedCompletion Expression Integer,f2: List OrderedCompletion
      Expression Integer,fail: failed,pole: potentialPole)

   (2)  [potentialPole,"failed"]
Type: List Union(f1: OrderedCompletion Expression Integer,f2: List OrderedCompletion Express
               Time: 1.10 (IN) + 1.95 (EV) + 0.38 (OT) + 0.37 (GC) = 3.80 sec
-- => 2^(s - 4) gamma(s/2)/gamma(4 - s/2)   (0 < Re s < 1)
--    [Gradshteyn and Ryzhik 17.43(16)]
MellinTransform(besselJ(3, x)/x**3, x, s)

   Compiling function MellinTransform with type (Expression Integer,
      Variable x,Variable s) -> List Union(f1: OrderedCompletion
      Expression Integer,f2: List OrderedCompletion Expression Integer,
      fail: failed,pole: potentialPole)

   >> Error detected within library code:
   Function not supported by Risch d.e.

initial (3) ->
real   14.4
user   4.9
sys    0.4

-------------------------------------------------------------------------------

Fri Jun 13 04:06:47 MET DST 1997
```

```
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Transforms ----------
-- Z transforms.  See _CRC Standard Mathematical Tables_, Twenty-first Edition,
-- The Chemical Rubber Company, 1973, p. 518.
-- Z[H(t - m T)] => z/[z^m (z - 1)]   (H is the Heaviside (unit step) function)
--heaviside(t - 3)
--heaviside(t - m)
-- ---------- Quit ----------
)quit


real   1.1
user   0.7
sys    0.1
```

26 Ordinary Difference and Differential Equations

```
-- ---------- Ordinary Difference and Differential Equations ----------
-- Second order linear recurrence equation: r(n) = (n - 1)^2 + m n
r:= operator('r);


                                                          Type: BasicOperator
```

```
                           Time: 0.07 (IN) + 0.05 (OT) + 0.02 (GC) = 0.13 sec
[r(n + 2) - 2 * r(n + 1) + r(n) = 2, r(0) = 1, r(1) = m]


   (2)  [r(n + 2) - 2r(n + 1) + r(n)= 2,r(0)= 1,r(1)= m]
                                        Type: List Equation Expression Integer
                Time: 1.17 (IN) + 0.05 (EV) + 0.24 (OT) + 0.07 (GC) = 1.52 sec
-- => r(n) = 3^n - 2^n   [Cohen, p. 67]
[r(n) = 5*r(n - 1) - 6*r(n - 2), r(0) = 0, r(1) = 1]


   (3)  [r(n)= 5r(n - 1) - 6r(n - 2),r(0)= 0,r(1)= 1]
                                        Type: List Equation Expression Integer
                                        Time: 0.10 (IN) + 0.05 (OT) = 0.15 sec
-- => r(n) = Fibonacci[n + 1]   [Cohen, p. 83]
[r(n) = r(n - 1) + r(n - 2), r(1) = 1, r(2) = 2]


   (4)  [r(n)= r(n - 1) + r(n - 2),r(1)= 1,r(2)= 2]
                                        Type: List Equation Expression Integer
                                        Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
-- => [c^(n+1) [c^(n+1) - 2 c - 2] + (n+1) c^2 + 2 c - n] / [(c-1)^3 (c+1)]
--    [Joan Z. Yu and Robert Israel in sci.math.symbolic]
[r(n) = (1 + c - c**(n-1) - c**(n+1))/(1 - c**n)*r(n - 1) _
        - c*(1 - c**(n-2))/(1 - c**(n-1))*r(n - 2) + 1, _
 r(1) = 1, r(2) = (2 + 2*c + c**2)/(1 + c)]


   (5)
   [
     r(n) =
                     n - 1              n + 1
           (r(n - 1)c      - r(n - 1))c
         +
             n - 1              n - 2                    n             n - 1 2
           (c      - c r(n - 2)c      + c r(n - 2) - 1)c  + r(n - 1)(c     )
         +
                                   n - 1              n - 2
           ((- c - 2)r(n - 1) - 1)c      + c r(n - 2)c      + (c + 1)r(n - 1)
         +
           - c r(n - 2) + 1
      /
           n - 1      n    n - 1
         (c      - 1)c  - c      + 1
     ,
                     2
```

```
                   c  + 2c + 2
    r(1)= 1, r(2)= -----------]
                      c + 1
                                       Type: List Equation Expression Integer
               Time: 0.48 (IN) + 0.13 (EV) + 0.13 (OT) + 0.03 (GC) = 0.78 sec
)clear properties r

-- Second order ODE with initial conditions---solve first using Laplace
-- transforms: f(t) = sin(2 t)/8 - t cos(2 t)/4
f:= operator('f);



                                                          Type: BasicOperator
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
ode:= D(f(t), t, 2) + 4*f(t) = sin(2*t)



         ,,
   (7)  f  (t) + 4f(t)= sin(2t)

                                            Type: Equation Expression Integer
               Time: 0.17 (IN) + 0.15 (EV) + 0.02 (OT) + 0.02 (GC) = 0.35 sec
map(e +-> laplace(e, t, s), %)


          2                          ,                2
   (8)  (s  + 4)laplace(f(t),t,s) - f (0) - f(0)s= ------
                                                     2
                                                    s  + 4
                                            Type: Equation Expression Integer
                            Time: 0.92 (EV) + 0.02 (OT) + 0.07 (GC) = 1.0 sec
subst(subst(%, f(0) = 0), subst(D(f(x), x), x = 0) = 0)


          2                         2
   (9)  (s  + 4)laplace(f(t),t,s)= ------
                                    2
                                   s  + 4
                                            Type: Equation Expression Integer
                           Time: 0.47 (IN) + 0.02 (EV) + 0.07 (OT) = 0.55 sec
map(e +-> e/(s**2 + 4), %)


                                  2
   (10)  laplace(f(t),t,s)= -------------
                             4     2
```

```
                            s  + 8s  + 16
                                            Type: Equation Expression Integer
                                       Time: 0.08 (IN) + 0.02 (OT) = 0.10 sec
map(e +-> inverseLaplace(e, s, t), %)


                   sin(2t) - 2t cos(2t)
   (11)  "failed"= --------------------
                             8
                            Type: Equation Union(Expression Integer,"failed")
                           Time: 0.02 (IN) + 0.10 (EV) + 0.03 (OT) = 0.15 sec
-- Now, solve the ODE directly
solve(ode, f, t = 0, [0, 0])


         sin(2t) - 2t cos(2t)
   (12)  --------------------
                   8
                                          Type: Union(Expression Integer,...)
                           Time: 0.30 (IN) + 0.87 (EV) + 0.07 (OT) = 1.23 sec
)clear properties f ode

-- Separable equation => y(x)^2 = 2 log(x + 1) + (4 x + 3)/(x + 1)^2 + 2 A
y:= operator('y);


                                                          Type: BasicOperator
                                                   Time: 0.03 (IN) = 0.03 sec
D(y(x), x) = x**2/(y(x)*(1 + x)**3)


                            2
          ,                x
   (14)  y (x)= -----------------------
                  3     2
                (x  + 3x  + 3x + 1)y(x)
                                            Type: Equation Expression Integer
                           Time: 0.08 (IN) + 0.02 (EV) + 0.07 (OT) = 0.17 sec
solve(%, y, x)


              2                         2              2
         (- 2x  - 4x - 2)log(x + 1) + (x  + 2x + 1)y(x)  - 4x - 3
   (15)  --------------------------------------------------------
                                 2
                               2x  + 4x + 2
```

```
                                           Type: Union(Expression Integer,...)
               Time: 0.02 (IN) + 0.82 (EV) + 0.02 (OT) + 0.42 (GC) = 1.27 sec
-- Homogeneous equation.  See Emilio O. Roxin, _Ordinary Differential
-- Equations_, Wadsworth Publishing Company, 1972, p. 11
-- => y(x)^2 = 2 x^2 log|A x|
D(y(x), x) = y(x)/x + x/y(x)


                    2    2
          ,     y(x)  + x
   (16)  y (x)= ----------
                  x y(x)
                                              Type: Equation Expression Integer
                           Time: 0.17 (IN) + 0.02 (EV) + 0.02 (OT) = 0.20 sec
solve(%, y, x)


             2             2
         - 2x log(x) + y(x)
   (17)  -------------------
                   2
                 2x
                                           Type: Union(Expression Integer,...)
                                                    Time: 0.15 (EV) = 0.15 sec
-- First order linear ODE: y(x) = [A - cos(x)]/x^3
y:= operator('y);


                                                           Type: BasicOperator
                                                                   Time: 0 sec
x**2 * D(y(x), x) + 3*x*y(x) = sin(x)/x


          2 ,                 sin(x)
   (19)  x y (x) + 3x y(x)= ------
                               x
                                              Type: Equation Expression Integer
                                                    Time: 0.08 (IN) = 0.08 sec
solve(%, y, x)


                        cos(x)          1
   (20)  [particular= - ------,basis= [--]]
                           3            3
                          x            x
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
```

```
                           Time: 0.33 (EV) + 0.02 (OT) + 0.02 (GC) = 0.37 sec
-- Exact equation => x + x^2 sin y(x) + y(x) = A   [Roxin, p. 15]
D(y(x), x) = -(1 + 2*x*sin(y(x)))/(1 + x**2*cos(y(x)))


          ,     - 2x sin(y(x)) - 1
   (21)  y (x)= ------------------
                   2
                  x cos(y(x)) + 1
                                             Type: Equation Expression Integer
                                        Time: 0.05 (IN) + 0.05 (OT) = 0.10 sec
solve(%, y, x)


          2
   (22)  x sin(y(x)) + y(x) + x
                                           Type: Union(Expression Integer,...)
                                        Time: 0.08 (EV) + 0.02 (OT) = 0.10 sec
-- Nonlinear ODE => y(x)^3/6 + A y(x) = x + B
ode:= D(y(x), x, 2) + y(x)*D(y(x), x)**3 = 0


          ,,           ,   3
   (23)  y  (x) + y(x)y (x) = 0

                                             Type: Equation Expression Integer
                                        Time: 0.05 (IN) + 0.03 (EV) = 0.08 sec
solve(%, y, x)


   >> Error detected within library code:
   getlincoeff: not an appropriate ordinary differential equation

initial (24) ->
real   20.4
user   9.1
sys    0.6

--------------------------------------------------------------------------------

Tue Mar 24 19:56:43 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha
```

```
(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Ordinary Difference and Differential Equations ----------
y:= operator('y);


                                                          Type: BasicOperator
                           Time: 0.05 (IN) + 0.08 (OT) + 0.02 (GC) = 0.15 sec
-- Nonlinear ODE => y(x)^3/6 + A y(x) = x + B
ode:= D(y(x), x, 2) + y(x)*D(y(x), x)**3 = 0


         ,,           ,   3
   (2)  y  (x) + y(x)y (x) = 0

                                            Type: Equation Expression Integer
               Time: 0.60 (IN) + 0.22 (EV) + 0.18 (OT) + 0.08 (GC) = 1.08 sec
-- => y(x) = [3 x + sqrt(1 + 9 x^2)]^(1/3) - 1/[3 x + sqrt(1 + 9 x^2)]^(1/3)
--    [Pos96]
solve(ode, y, x = 0, [0, 2])


   >> Error detected within library code:
   getlincoeff: not an appropriate ordinary differential equation

initial (3) ->
real   5.1
user   2.3
sys    0.2

-------------------------------------------------------------------------------

Tue Mar 24 19:56:49 MET 1998
```

```
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Ordinary Difference and Differential Equations ----------
y:= operator('y);


                                                         Type: BasicOperator
                           Time: 0.05 (IN) + 0.08 (OT) + 0.02 (GC) = 0.15 sec
-- A simple parametric ODE: y(x, a) = A e^(a x)
D(y(x, a), x) = a*y(x, a)


   (2)  y  (x,a)= a y(x,a)
         ,1
                                                 Type: Equation Expression Integer
               Time: 0.47 (IN) + 0.15 (EV) + 0.18 (OT) + 0.07 (GC) = 0.86 sec
solve(%, y, x);


   >> Error detected within library code:
   parseODE: equation has order 0

initial (3) ->
real   2.2
user   1.6
sys    0.2

-------------------------------------------------------------------------------
```

```
Tue Mar 24 20:03:01 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
------------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
------------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

-- ---------- Ordinary Difference and Differential Equations ----------
y:= operator('y);


                                                           Type: BasicOperator
                                        Time: 0.05 (IN) + 0.08 (OT) = 0.13 sec
-- ODE with boundary conditions.  This problem has nontrivial solutions
-- y(x) = A sin([pi/2 + n pi] x) for n an arbitrary integer
solve(D(y(x), x, 2) + k**2*y(x) = 0, y, x = 0, [0])


   (2)  0
                                           Type: Union(Expression Integer,...)
                Time: 1.15 (IN) + 0.97 (EV) + 0.17 (OT) + 0.17 (GC) = 2.45 sec
--bc(%, x = 0, y = 0, x = 1, D(y(x), x) = 0)
-- => y(x) = Z_v[sqrt(x)] where Z_v is an arbitrary Bessel function of order v
--    [Gradshteyn and Ryzhik 8.491(9)]
D(y(x), x, 2) + 1/x*D(y(x), x) + 1/(4*x)*(1 - v**2/x)*y(x) = 0


          2 ,,         ,            2
        4x y  (x) + 4xy (x) + (x - v )y(x)

   (3)  ------------------------------------= 0
```

```
                          2
                        4x
                                            Type: Equation Expression Integer
                           Time: 0.35 (IN) + 0.03 (EV) + 0.08 (OT) = 0.47 sec
solve(%, y, x)

                                                          2
                                                     2   v
   WARNING (genufact): No known algorithm to factor ?  - --, trying square-free.
                                                          4
   WARNING (genufact): No known algorithm to factor
      3     2       2          2
     ?  - 3?  + (- v  + 3)? + v  - 1, trying square-free.

   (4)  [particular= 0,basis= []]
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
            Time: 0.02 (IN) + 81.03 (EV) + 0.02 (OT) + 17.42 (GC) = 98.48 sec
-- Delay (or mixed differential-difference) equation.  See Daniel Zwillinger,
-- _Handbook of Differential Equations_, Second Edition, Academic Press, Inc.,
-- 1992, p. 210 => y(t) = y0 sum((-a)^n (t - n + 1)^n/n!, n = 0..floor(t) + 1)
D(y(t), t) + a*y(t - 1) = 0


         ,
   (5)  y (t) + a y(t - 1)= 0

                                            Type: Equation Expression Integer
                                       Time: 0.22 (IN) + 0.07 (OT) = 0.28 sec
solve(%, y, t)


           t
         ++
   (6)   |   a y(%V - 1)d%V  + y(t)
        ++
                                          Type: Union(Expression Integer,...)
               Time: 0.02 (IN) + 0.30 (EV) + 0.02 (OT) + 0.03 (GC) = 0.37 sec
-- Discontinuous ODE   [Zwillinger, p. 221]
-- => y(t) = cosh t   (0 <= t < T)
--           (sin T cosh T + cos T sinh T) sin t
--           + (cos T cosh T - sin T sinh T) cos t   (T <= t)
sgn(t) == if t < 0 then -1 else 1;

                                                                   Type: Void
                                                                  Time: 0 sec
solve(D(y(t), t, 2) + sgn(t - TT)*y(t) = 0, y, t = 0, [1, 0])
```

```
   Compiling function sgn with type Polynomial Integer -> Integer

   (8)  cos(t)
                                          Type: Union(Expression Integer,...)
                           Time: 0.15 (IN) + 0.13 (EV) + 0.03 (OT) = 0.32 sec
)clear properties sgn

   Compiled code for sgn has been cleared.
solve(D(y(t), t, 2) + sign(t - TT)*y(t) = 0, y, t = 0, [1, 0])

   There are 31 exposed and 17 unexposed library operations named *
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op *
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named *
      with argument type(s)
                                   failed
                             Expression Integer

-- Integro-differential equation.  See A. E. Fitzgerald, David E. Higginbotham
-- and Arvin Grabel, _Basic Electrical Engineering_, Fourth Edition,
-- McGraw-Hill Book Company, 1975, p. 117.
-- => i(t) = 5/13 [-8 e^(-4 t) + e^(-t) (8 cos 2 t + sin 2 t)]
i:= operator('i);



                                                          Type: BasicOperator
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
eqn:= D(i(t), t) + 2*i(t) + 5*'integrate(i(tau), tau = 0..t) = 10*%e**(-4*t)


          ,                                                   - 4t
   (10)  i (t) + 2i(t) + 5integrate(i(tau),tau= 0,t    )= 10%e

                                                    Type: Equation OutputForm
                           Time: 0.17 (IN) + 0.05 (EV) + 0.03 (OT) = 0.25 sec
solve(eqn, i, t)

   There are 6 exposed and 1 unexposed library operations named solve
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
```

```
                            )display op solve
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named solve
      with argument type(s)
                           Equation OutputForm
                              BasicOperator
                               Variable t

map(e +-> laplace(e, t, s), eqn)

   There are 1 exposed and 0 unexposed library operations named laplace
      having 3 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                           )display op laplace
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.
   Cannot find a definition or applicable library operation named
      laplace with argument type(s)
                               OutputForm
                               Variable t
                               Variable s

   AXIOM will attempt to step through and interpret the code.

   Anonymous user functions created with +-> that are processed in
      interpret-code mode must have result target information
      available. This information is not present so AXIOM cannot
      proceed any further. This may be remedied by declaring the
      function.
--subst(%, [t = 0, D(i(t), t) = 10])
)clear properties i eqn

-- System of two linear, constant coefficient ODEs:
-- x(t) = e^t [A cos(t) - B sin(t)], y(t) = e^t [A sin(t) + B cos(t)]
x:= operator('x);


                                                      Type: BasicOperator
                                                              Time: 0 sec
system:= [D(x(t), t) = x(t) - y(t), D(y(t), t) = x(t) + y(t)]
```

```
           ,                    ,
   (12)  [x (t)= - y(t) + x(t),y (t)= y(t) + x(t)]

                                       Type: List Equation Expression Integer
                           Time: 0.05 (IN) + 0.03 (EV) + 0.02 (OT) = 0.10 sec
-- Check the answer
ans:= [x(t) = %e^t*(A*cos(t) - B*sin(t)), y(t) = %e^t*(A*sin(t) + B*cos(t))]


                      t                   t           t                   t
   (13)  [x(t)= - B %e sin(t) + A cos(t)%e ,y(t)= A %e sin(t) + B cos(t)%e ]
                                       Type: List Equation Expression Integer
                           Time: 0.25 (IN) + 0.07 (EV) + 0.02 (OT) = 0.33 sec
dans:= map(e +-> D(lhs(e), t) = D(rhs(e), t), ans)


   (14)
     ,                t                          t
   [x (t)= (- B - A)%e sin(t) + (- B + A)cos(t)%e ,

     ,                t                        t
    y (t)= (- B + A)%e sin(t) + (B + A)cos(t)%e ]

                                       Type: List Equation Expression Integer
                           Time: 0.02 (IN) + 0.07 (EV) + 0.02 (OT) = 0.10 sec
map(e +-> subst(lhs(e), dans) = subst(rhs(e), ans), system)


   (15)
   [
                t                          t
     (- B - A)%e sin(t) + (- B + A)cos(t)%e  =
                  t                          t
       (- B - A)%e sin(t) + (- B + A)cos(t)%e
     ,
               t                        t             t                        t
    (- B + A)%e sin(t) + (B + A)cos(t)%e = (- B + A)%e sin(t) + (B + A)cos(t)%e
     ]
                                       Type: List Equation Expression Integer
                           Time: 0.02 (IN) + 0.03 (EV) + 0.03 (OT) = 0.08 sec
)clear properties ans dans

-- Triangular system of two ODEs: x(t) = A e^t [sin(t) + 2],
--    y(t) = A e^t [5 - cos(t) + 2 sin(t)]/5 + B e^(-t)
-- See 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.
system:= [D(x(t), t) = x(t) * (1 + cos(t)/(2 + sin(t))), _
          D(y(t), t) = x(t) - y(t)]


           ,     x(t)sin(t) + x(t)cos(t) + 2x(t)  ,
   (16)  [x (t)= --------------------------------,y (t)= - y(t) + x(t)]
                            sin(t) + 2
                                       Type: List Equation Expression Integer
                           Time: 0.08 (IN) + 0.03 (EV) + 0.03 (OT) = 0.15 sec
-- Try solving this system one equation at a time
solve(system.1, x, t)


                                  t            t
   (17)  [particular= 0,basis= [%e sin(t) + 2%e ]]
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
                           Time: 0.70 (EV) + 0.02 (OT) + 0.03 (GC) = 0.75 sec
x(t) = C1 * %.basis.1


                    t               t
   (18)  x(t)= C1 %e sin(t) + 2C1 %e
                                            Type: Equation Expression Integer
                                       Time: 0.17 (IN) + 0.02 (OT) = 0.18 sec
solve(subst(system.2, %), y, t)


   (19)
                      - t   t 2                              - t   t 2
                2C1 %e   (%e ) sin(t) + (- C1 cos(t) + 5C1)%e   (%e )
   [particular= ------------------------------------------------------,
                                           5
              - t
    basis= [%e   ]]
Type: Union(Record(particular: Expression Integer,basis: List Expression Integer),...)
                Time: 0.03 (IN) + 2.0 (EV) + 0.02 (OT) + 0.05 (GC) = 2.10 sec
y(t) = simplify(%.particular) + C2 * %.basis.1


                     t                              t          - t
               2C1 %e sin(t) + (- C1 cos(t) + 5C1)%e  + 5C2 %e
   (20)  y(t)= ---------------------------------------------------
                                        5
                                            Type: Equation Expression Integer
                           Time: 0.08 (IN) + 0.23 (EV) + 0.03 (OT) = 0.35 sec
```

```
)clear properties x y

x:= operator('x);


                                                       Type: BasicOperator
                                                               Time: 0 sec
y:= operator('y);


                                                       Type: BasicOperator
                                                Time: 0.02 (IN) = 0.02 sec
z:= operator('z);


                                                       Type: BasicOperator
                                                Time: 0.03 (IN) = 0.03 sec
-- 3 x 3 linear system with constant coefficients:
-- (1) real distinct characteristic roots (= 2, 1, 3)   [Roxin, p. 109]
--     => x(t) = A e^(2 t),   y(t) = B e^t + C e^(3 t),
--        z(t) = -A e^(2 t) - C e^(3 t)
system:= [D(x(t), t) =  2*x(t), _
          D(y(t), t) = -2*x(t) + y(t) - 2*z(t), _
          D(z(t), t) =    x(t)        + 3*z(t)]



          ,            ,                              ,
   (24)  [x (t)= 2x(t),y (t)= - 2z(t) + y(t) - 2x(t),z (t)= 3z(t) + x(t)]

                                       Type: List Equation Expression Integer
                                       Time: 0.13 (IN) + 0.03 (OT) = 0.17 sec
-- (2) complex characteristic roots (= 0, -1 +- sqrt(2) i)   [Roxin, p. 111]
--     => x(t) = A + e^(-t)/3 [-(B + sqrt(2) C) cos(sqrt(2) t) +
--                             (sqrt(2) B - C) sin(sqrt(2) t)],
--        y(t) = e^(-t) [B cos(sqrt(2) t) + C sin(sqrt(2) t)],
--        z(t) = e^(-t) [(-B + sqrt(2) C) cos(sqrt(2) t)
--                      -(sqrt(2) B + C) sin(sqrt(2) t)]
system:= [D(x(t), t) = y(t), D(y(t), t) = z(t), _
          D(z(t), t) = -3*y(t) - 2*z(t)]



          ,            ,            ,
   (25)  [x (t)= y(t),y (t)= z(t),z (t)= - 2z(t) - 3y(t)]

                                       Type: List Equation Expression Integer
                         Time: 0.08 (IN) + 0.02 (EV) + 0.03 (OT) = 0.13 sec
```

```
-- (3) multiple characteristic roots (= 2, 2, 2)   [Roxin, p. 113]
--     => x(t) = e^(2 t) [A + C (1 + t)],   y(t) = B e^(2 t),
--        z(t) = e^(2 t) [A + C t]
system:= [D(x(t), t) = 3*x(t) - z(t), D(y(t), t) = 2*y(t), _
          D(z(t), t) = x(t) + z(t)]


           ,                     ,            ,
   (26)  [x (t)= - z(t) + 3x(t),y (t)= 2y(t),z (t)= z(t) + x(t)]

                                       Type: List Equation Expression Integer
               Time: 0.57 (IN) + 0.03 (EV) + 0.03 (OT) + 0.47 (GC) = 1.10 sec
-- x(t) = x0 + [4 sin(w t)/w - 3 t] x0'   [Rick Niles]
--        + 6 [w t - sin(w t)] y0 + 2/w [1 - cos(w t)] y0',
-- y(t) = -2/w [1 - cos(w t)] x0' + [4 - 3 cos(w t)] y0 + sin(w t)/w y0'
system:= [D(x(t), t, 2) = 2*w*D(y(t), t), _
          D(y(t), t, 2) = -2*w*D(x(t), t) + 3*w**2*y(t)]


           ,,        ,     ,,          ,       2
   (27)  [x  (t)= 2wy (t),y  (t)= - 2wx (t) + 3w y(t)]

                                       Type: List Equation Expression Integer
                           Time: 0.13 (IN) + 0.03 (EV) + 0.02 (OT) = 0.18 sec
)clear properties x y z system

-- ---------- Quit ----------
)quit


real   267.4
user   92.4
sys    1.8
```

27 Partial Differential Equations

```
-- ---------- Partial Differential Equations ----------
-- A very simple PDE => g(x) + h(y) for arbitrary functions g and h
f:= operator('f);


                                                          Type: BasicOperator
                           Time: 0.07 (IN) + 0.10 (OT) + 0.05 (GC) = 0.22 sec
D(f(x, y), [x, y]) = 0
```

```
   (2)  f    (x,y)= 0
         ,1,2
                                                 Type: Equation Expression Integer
              Time: 0.65 (IN) + 0.27 (EV) + 0.27 (OT) + 0.15 (GC) = 1.33 sec
)clear properties f

-- Heat equation: the fundamental solution is 1/sqrt(4 pi t) exp(-x^2/[4 t]).
-- If f(x, t) and a(x, t) are solutions, the most general solution obtainable
-- from f(x, t) by group transformations is of the form u(x, t) = a(x, t)
-- + 1/sqrt(1 + 4 e6 t) exp(e3 - [e5 x + e6 x^2 -  e5^2 t]/[1 + 4 e6 t])
-- f([e^(-e4) (x - 2 e5 t)]/[1 + 4 e6 t] - e1, [e^(-2 e4) t]/[1 + 4 e6 t] - e2)
-- See Peter J. Olver, _Applications of Lie Groups to Differential Equations_,
-- Second Edition, Springer Verlag, 1993, p. 120 (an excellent book).  See also
-- Heat.input
u:= operator('u);



                                                              Type: BasicOperator
                                          Time: 0.03 (IN) + 0.02 (OT) = 0.05 sec
D(u(x, t), t) = D(u(x, t), x, 2)


   (4)  u  (x,t)= u    (x,t)
         ,2        ,1,1
                                                 Type: Equation Expression Integer
                         Time: 0.07 (IN) + 0.05 (EV) + 0.03 (OT) = 0.15 sec
)clear properties u

-- Potential equation on a circular disk---a separable PDE
-- => v(r, theta) = a[0] + sum(a[n] r^n cos(n theta), n = 1..infinity)
--                       + sum(b[n] r^n sin(n theta), n = 1..infinity)
v:= operator('v);



                                                              Type: BasicOperator
                                                      Time: 0.03 (IN) = 0.03 sec
1/r * D(r * D(v(r, theta), r), r) _
        + 1/r**2 * D(v(r, theta), theta, 2) = 0



                      2
        v    (r,theta) + r v    (r,theta) + rv  (r,theta)
         ,2,2               ,1,1              ,1
   (6)  ---------------------------------------------------= 0
```

```
                                 2
                                r
                                            Type: Equation Expression Integer
                           Time: 0.63 (IN) + 0.15 (EV) + 0.10 (OT) = 0.88 sec
)clear properties v

-- ---------- Quit ----------
)quit


real   9.5
user   3.3
sys    0.3
```

28 Operators

```
-- ---------- Operators ----------
f(x) == exp(x)

                                                                   Type: Void
                                                   Time: 0.02 (IN) = 0.02 sec
g(x) == x**2

                                                                   Type: Void
                                                   Time: 0.02 (IN) = 0.02 sec
-- (f + 2 g)(y) => e^y + 2 y^2
(f + 2*g)(y)

   There are 31 exposed and 17 unexposed library operations named *
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                                )display op *
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named *
      with argument type(s)
                               PositiveInteger
                              FunctionCalled g

-- (f o g)(y) => e^(y^2)
(f * g)(y)
```

```
   There are 31 exposed and 17 unexposed library operations named *
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op *
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named *
      with argument type(s)
                            FunctionCalled f
                            FunctionCalled g

)clear properties f g

-- Linear differential operator
DD : LODO(Expression Integer, e +-> D(e, x)) := D()


   (3)  D
 Type: LinearOrdinaryDifferentialOperator(Expression Integer,theMap("NIL",1))
               Time: 0.08 (IN) + 0.02 (EV) + 0.15 (OT) + 0.07 (GC) = 0.32 sec
L:= (DD - 1) * (DD + 2)


         2
   (4)  D  + D - 2
 Type: LinearOrdinaryDifferentialOperator(Expression Integer,theMap("NIL",1))
                           Time: 0.13 (IN) + 0.10 (EV) + 0.03 (GC) = 0.27 sec
f:= operator('f);


                                                         Type: BasicOperator
                                       Time: 0.07 (IN) + 0.02 (OT) = 0.08 sec
g:= operator('g);


                                                         Type: BasicOperator
                                                   Time: 0.05 (IN) = 0.05 sec
-- => f'' + f' - 2 f
L(f(x))


         ,,        ,
   (7)  f  (x) + f (x) - 2f(x)
```

```
                                                     Type: Expression Integer
               Time: 0.53 (IN) + 0.22 (EV) + 0.13 (OT) + 0.07 (GC) = 0.95 sec
-- => g’’(y) + g’(y) - 2 g(y)
subst(L(subst(g(y), y = x)), x = y)


         ,,       ,
   (8)  g  (y) + g (y) - 2g(y)

                                                     Type: Expression Integer
                           Time: 0.54 (IN) + 0.08 (EV) + 0.10 (OT) = 0.72 sec
-- => 2 A [(1 + z) cos(z^2) - (1 + 2 z^2) sin(z^2)]
subst(L(subst(A * sin(z**2), z = x)), x = z)


               2           2                    2
   (9)  (- 4A z  - 2A)sin(z ) + (2A z + 2A)cos(z )
                                                     Type: Expression Integer
               Time: 0.13 (IN) + 0.20 (EV) + 0.07 (OT) + 0.02 (GC) = 0.42 sec
-- Truncated Taylor series operator
T:= (f, xx, a) +-> subst((DD**0)(f(x)), x = a)/factorial(0) * (xx - a)**0 + _
                   subst((DD**1)(f(x)), x = a)/factorial(1) * (xx - a)**1 + _
                   subst((DD**2)(f(x)), x = a)/factorial(2) * (xx - a)**2


   (10)
     (f,xx,a)
   +->
               0                                 1
       subst(DD (f(x)),x= a)         0   subst(DD (f(x)),x= a)         1
       --------------------- (xx - a)  + --------------------- (xx - a)
            factorial(0)                      factorial(1)
     +
               2
       subst(DD (f(x)),x= a)         2
       --------------------- (xx - a)
            factorial(2)
                                                      Type: AnonymousFunction
                                                   Time: 0.02 (OT) = 0.02 sec
-- => f(a) + f’(a) (x - a) + f’’(a) (x - a)^2/2
T(f, x, a)


           2           2  ,,                ,
         (x  - 2a x + a )f  (a) + (2x - 2a)f (a) + 2f(a)
```

```
   (11)  ------------------------------------------------
                                2
                                                     Type: Expression Integer
                           Time: 0.50 (IN) + 0.13 (EV) + 0.10 (OT) = 0.73 sec
-- => g(b) + g'(b) (y - b) + g''(b) (y - b)^2/2
T(g, y, b)


           2           2  ,,                ,
         (y  - 2b y + b )g  (b) + (2y - 2b)g (b) + 2g(b)

   (12)  ------------------------------------------------
                                2
                                                     Type: Expression Integer
                           Time: 0.18 (IN) + 0.12 (EV) + 0.07 (OT) = 0.37 sec
-- => sin(c) + cos(c) (z - c) - sin(c) (z - c)^2/2
T(sin, z, c)


             2           2
         (- z  + 2c z - c  + 2)sin(c) + (2z - 2c)cos(c)
   (13)  -----------------------------------------------
                                2
                                                     Type: Expression Integer
                           Time: 0.18 (IN) + 0.07 (EV) + 0.05 (OT) = 0.30 sec
)clear properties DD L f g T

-- Define the binary infix operator ~ so that x ~ y => sqrt(x^2 + y^2)
-- Make it associative: 3 ~ 4 ~ 12 => 13
-- Define the matchfix pair of operators | and | so that | x | => abs(x)
-- ---------- Quit ----------
)quit


real   6.8
user   5.8
sys    0.4
```

29 Programming and Miscellaneous

```
-- ---------- Programming and Miscellaneous ----------
-- How easy is it to substitute x for a + b in the following expression?
-- => (x + c)^2 + (d - x)^2
expr:= (a + b + c)**2 + (d - a - b)**2
```

```
         2                   2                  2            2
   (1)  d  + (- 2b - 2a)d + c  + (2b + 2a)c + 2b  + 4a b + 2a
                                                     Type: Polynomial Integer
               Time: 0.23 (IN) + 0.02 (EV) + 0.18 (OT) + 0.10 (GC) = 0.53 sec
subst(expr, a + b = x)


   >> Error detected within library code:
   left hand side must be a single kernel

initial (2) ->
real   3.2
user   1.8
sys    0.2

-------------------------------------------------------------------------------

Mon Jan  5 00:26:00 MET 1998
anne
% axiom
Axiom Computer Algebra System (Release 2.1)
Digital Unix on DEC Alpha

(AXIOM Sockets) The AXIOM server number is undefined.
-----------------------------------------------------------------------------
   Issue )copyright to view copyright notices.
   Issue )summary for a summary of useful system commands.
   Issue )quit to leave AXIOM and return to shell.
-----------------------------------------------------------------------------

initial (1) -> -- ----------[ A x i o m ]----------
-- ---------- Initialization ----------
)set messages autoload off

)set messages time on

)set quit unprotected

)set streams calculate 7

-- ---------- Programming and Miscellaneous ----------
-- How easy is it to substitute x for a + b in the following expression?
-- => (x + c)^2 + (d - x)^2
expr:= (a + b + c)**2 + (d - a - b)**2
```

```
         2                   2                  2            2
   (1)  d  + (- 2b - 2a)d + c  + (2b + 2a)c + 2b  + 4a b + 2a
                                                     Type: Polynomial Integer
               Time: 0.25 (IN) + 0.03 (EV) + 0.15 (OT) + 0.07 (GC) = 0.50 sec
subst(expr, b = x - a)


          2                   2    2
   (2)  2x  + (- 2d + 2c)x + d  + c
                                                     Type: Expression Integer
               Time: 0.38 (IN) + 0.05 (EV) + 0.17 (OT) + 0.12 (GC) = 0.72 sec
)clear properties expr

-- How easy is it to substitute r for sqrt(x^2 + y^2) in the following
-- expression? => x/r
x/sqrt(x**2 + y**2)


             x
   (3)  ----------
         +-------+
         | 2    2
        \|y   + x
                                                     Type: Expression Integer
                           Time: 0.30 (IN) + 0.07 (EV) + 0.12 (OT) = 0.48 sec
subst(%, sqrt(x**2 + y**2) = r)


        x
   (4)  -
        r
                                                     Type: Expression Integer
                                       Time: 0.15 (IN) + 0.02 (EV) = 0.17 sec
-- Change variables so that the following transcendental expression is
-- converted into a rational expression   [Vernor Vinge]
-- => (r - 1)^4 (u^4 - r u^3 - r^3 u + r u + r^4)/[u^4 (2 r - 1)^2]
q:= (1/r**4 + 1/(r**2 - 2*r*cos(t) + 1)**2 _
           - 2*(r - cos(t))/(r**2 * (r**2 - 2*r*cos(t) + 1)**(3/2))) / _
    (1/r**4 + 1/(r - 1)**4 - 2*(r - 1)/(r**2 * (r**2 - 2*r + 1)**(3/2)))


   (5)
              6      5      4      3     2       2
           (4r  - 16r  + 24r  - 16r  + 4r )cos(t)
```

```
       +
              7      6      5      4      3      2                 8     7
         (- 4r  + 16r  - 28r  + 32r  - 28r  + 16r  - 4r)cos(t) + 2r  - 8r
       +
            6      5      4      3     2
         14r  - 16r  + 15r  - 12r  + 8r  - 4r + 1
    *
        +-----------+ +--------------------+
        | 2           |               2
       \|r  - 2r + 1 \|- 2r cos(t) + r  + 1
   +
              7      6      5      4     3       2
         (- 4r  + 16r  - 24r  + 16r  - 4r )cos(t)
       +
            8      7      6      5      4     3     2            9     8
         (6r  - 24r  + 38r  - 32r  + 18r  - 8r  + 2r )cos(t) - 2r  + 8r
       +
              7      6      5     4     3
         - 14r  + 16r  - 14r  + 8r  - 2r
    *
        +-----------+
        | 2
       \|r  - 2r + 1
/
              6      5      4      3     2       2
           (8r  - 16r  + 24r  - 16r  + 4r )cos(t)
         +
                7      6      5      4      3      2                 8     7
           (- 8r  + 16r  - 32r  + 32r  - 28r  + 16r  - 4r)cos(t) + 2r  - 4r
         +
              6      5      4      3     2
           10r  - 12r  + 15r  - 12r  + 8r  - 4r + 1
      *
          +-----------+
          | 2
         \|r  - 2r + 1
     +
            7      6      5     4       2
       (- 8r  + 24r  - 24r  + 8r )cos(t)
     +
          8      7      6      5      4     3            9     8      7      6
       (8r  - 24r  + 32r  - 32r  + 24r  - 8r )cos(t) - 2r  + 6r  - 10r  + 14r
     +
            5      4     3     2
       - 14r  + 10r  - 6r  + 2r
  *
```

```
        +--------------------+
        |               2
       \|- 2r cos(t) + r  + 1
                                                     Type: Expression Integer
               Time: 0.61 (IN) + 0.28 (EV) + 0.28 (OT) + 0.07 (GC) = 1.25 sec
subst(q, cos(t) = (r**2 - u**2 + 1)/(2*r))


   (6)
                                                                   +-----------+
            4     3     2           4    8     7     6     5    4  | 2
         ((r  - 4r  + 6r  - 4r + 1)u  + r  - 4r  + 6r  - 4r  + r )\|r  - 2r + 1
      *
          +--+
          | 2
         \|u
     +
              5     4     3     2      4       7     6     5     3     2      2
         ((- r  + 4r  - 6r  + 4r  - r)u  + (- r  + 4r  - 5r  + 5r  - 4r  + r)u )
      *
          +-----------+
          | 2
         \|r  - 2r + 1
  /
                                        +-----------+
              4     3     2           4 | 2
           (2r  - 4r  + 6r  - 4r + 1)u \|r  - 2r + 1
         +
                5     4     3     2  4
           (- 2r  + 6r  - 6r  + 2r )u
    *
        +--+
        | 2
       \|u
                                                     Type: Expression Integer
                           Time: 0.15 (IN) + 0.17 (EV) + 0.12 (OT) = 0.43 sec
(rule sqrt(x**2) == x)(%)


   (7)
           4     3     2           4       5     4     3     2      3
         (r  - 4r  + 6r  - 4r + 1)u  + (- r  + 4r  - 6r  + 4r  - r)u
       +
             7     6     5     3     2          8     7     6     5    4
         (- r  + 4r  - 5r  + 5r  - 4r  + r)u + r  - 4r  + 6r  - 4r  + r
    *
```

```
        +-----------+
        | 2
       \|r  - 2r + 1
  /
                                  +-----------+
        4     3     2           4 | 2                  5     4     3     2  4
     (2r  - 4r  + 6r  - 4r + 1)u \|r  - 2r + 1  + (- 2r  + 6r  - 6r  + 2r )u
                                                     Type: Expression Integer
                          Time: 0.07 (IN) + 0.93 (EV) + 0.12 (OT) = 1.12 sec
(rule sqrt(r**2 - 2*r + 1) == r - 1)(%)


   (8)
         4     3     2           4       5     4     3     2      3
       (r  - 4r  + 6r  - 4r + 1)u  + (- r  + 4r  - 6r  + 4r  - r)u
     +
           7     6     5     3     2          8     7     6     5    4
       (- r  + 4r  - 5r  + 5r  - 4r  + r)u + r  - 4r  + 6r  - 4r  + r
  /
        2           4
     (4r  - 4r + 1)u
                                                     Type: Expression Integer
                          Time: 0.03 (IN) + 0.47 (EV) + 0.07 (OT) = 0.57 sec
map(factor, % :: Fraction Polynomial Integer)


               4  4      3       3          4
        (r - 1) (u  - r u  + (- r  + r)u + r )
   (9)  --------------------------------------
                              2 4
                      (2r - 1) u
                                   Type: Fraction Factored Polynomial Integer
              Time: 0.17 (IN) + 0.25 (EV) + 0.03 (OT) + 0.02 (GC) = 0.47 sec
-- Establish a rule to symmetrize a differential operator:   [Stanly Steinberg]
-- f g'' + f' g' -> (f g')'
f:= operator('f);



                                                          Type: BasicOperator
                                       Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
g:= operator('g);



                                                          Type: BasicOperator
                                                   Time: 0.03 (IN) = 0.03 sec

symmetrize:= rule _
```

```
   f(x)*D(g(x), x, 2) + D(f(x), x)*D(g(x), x) == D(f(x)*D(g(x), x), x)


              ,,       ,    ,
   (12)  f(x)g  (x) + f (x)g (x) + %B == 'D('f(x)'D('g(x),x),x) + %B

                        Type: RewriteRule(Integer,Integer,Expression Integer)
                           Time: 0.62 (IN) + 0.05 (EV) + 0.07 (OT) = 0.73 sec
q:= f(x)*D(g(x), x, 2) + D(f(x), x)*D(g(x), x)


              ,,       ,    ,
   (13)  f(x)g  (x) + f (x)g (x)

                                                     Type: Expression Integer
                           Time: 0.03 (IN) + 0.03 (EV) + 0.05 (OT) = 0.12 sec
symmetrize q

   There are no library operations named %diff
      Use HyperDoc Browse or issue
                               )what op %diff
      to learn if there is any operation containing " %diff " in its
      name.

   Cannot find a definition or applicable library operation named %diff
      with argument type(s)
                             Expression Integer
                                Variable %%01
                                 Variable x

-- => 2 (f g')' + f g
symmetrize 2*q + f(x)*g(x)


               ,,        ,    ,
   (14)  2f(x)g  (x) + 2f (x)g (x) + f(x)g(x)

                                                     Type: Expression Integer
                                       Time: 0.05 (IN) + 0.02 (OT) = 0.07 sec
)clear properties f g q

-- Infinite lists: [1 2 3 4 5 ...] * [1 3 5 7 9 ...]
-- => [1 6 15 28 45 66 91 ...]
l1:= [i for i in 1..];
```

```
                                                 Type: Stream PositiveInteger
                                       Time: 0.02 (IN) + 0.03 (OT) = 0.05 sec
l2:= [2*i-1 for i in 1..];



                                                         Type: Stream Integer
                                       Time: 0.03 (IN) + 0.02 (EV) = 0.05 sec
[l1.i*l2.i for i in 1..]


   (17)  [1,6,15,28,45,66,91,...]
                                                         Type: Stream Integer
                                       Time: 0.02 (IN) + 0.02 (OT) = 0.03 sec
)clear properties l1 l2

-- Write a simple program to compute Legendre polynomials
p(n, x) == 1/(2**n*factorial(n)) * D((x**2 - 1)**n, x, n)

                                                                   Type: Void
                                                                  Time: 0 sec
-- p[0](x) = 1,   p[1](x) = x,   p[2](x) = (3 x^2 - 1)/2,
-- p[3](x) = (5 x^3 - 3 x)/2,   p[4](x) = (35 x^4 - 30 x^2 + 3)/8
for i in 0..4 repeat _
  (output(""); _
   output(concat(["p(", string(i), ", x) = "])); _
   output(p(i, x)) )

   Compiling function p with type (NonNegativeInteger,Variable x) ->
      Polynomial Fraction Integer

   p(0, x) =
   1

   p(1, x) =
   x

   p(2, x) =
   3  2   1
   - x  - -
   2      2

   p(3, x) =
   5  3   3
   - x  - - x
   2      2
```

```
   p(4, x) =
   35  4   15  2   3
   -- x  - -- x  + -
    8       4      8
                                                                  Type: Void
                              Time: 0.20 (IN) + 0.08 (EV) + 0.12 (OT) = 0.40 sec
-- p[4](1) = 1
eval(p(4, x), x = 1)

   Compiling function p with type (PositiveInteger,Variable x) ->
      Polynomial Fraction Integer

   (20)  1
                                              Type: Polynomial Fraction Integer
                              Time: 0.63 (IN) + 0.02 (EV) + 0.12 (OT) = 0.77 sec
-- Now, perform the same computation using a recursive definition
pp(0, x) == 1

                                                                  Type: Void
                                                    Time: 0.02 (OT) = 0.02 sec
pp(1, x) == x

                                                                  Type: Void
                                                                 Time: 0 sec
pp(n, x) == ((2*n - 1)*x*pp(n - 1, x) - (n - 1)*pp(n - 2, x))/n

                                                                  Type: Void
                                                                 Time: 0 sec
for i in 0..4 repeat _
  (output(""); _
   output(concat(["pp(", string(i), ", x) = "])); _
   output(pp(i, x)) )

   Compiling function pp with type (Integer,Variable x) -> Polynomial
      Fraction Integer

   pp(0, x) =
   1

   pp(1, x) =
   x

   pp(2, x) =
   3  2   1
   - x  - -
   2      2
```

```
   pp(3, x) =
   5  3   3
   - x  - - x
   2      2

   pp(4, x) =
   35  4   15  2   3
   -- x  - -- x  + -
    8       4      8
                                                                   Type: Void
                          Time: 0.33 (IN) + 0.03 (EV) + 0.05 (OT) = 0.42 sec
pp(4, 1)

   Compiling function pp with type (Integer,Integer) -> Fraction
      Integer

+++ |*2;pp;1;initial| redefined

   (25)  1
                                                       Type: Fraction Integer
                                      Time: 0.08 (IN) + 0.05 (OT) = 0.13 sec
)clear properties p pp

   Compiled code for p has been cleared.
   Compiled code for pp has been cleared.
-- Iterative computation of Fibonacci numbers
myfib(n) == ( _
            local i, j, k, f; _
            if n < 0 then _
               error("undefined") _
            else if n < 2 then _
               n _
            else _
              (j:= 0,   k:= 1, _
               for i in 2..n repeat _
                 (f:= j + k,   j:= k,   k:= f), _
               return(f)));

                                                                   Type: Void
                                                  Time: 0.02 (IN) = 0.02 sec
-- Convert the function into FORTRAN syntax
outputAsFortran(myfib)

   There are 4 exposed and 0 unexposed library operations named
      outputAsFortran having 1 argument(s) but none was determined to
```

```
      be applicable. Use HyperDoc Browse, or issue
                        )display op outputAsFortran
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.

   Cannot find a definition or applicable library operation named
      outputAsFortran with argument type(s)
                          FunctionCalled myfib

-- Create a list of the first 11 values of the function.
[myfib(i) for i in 0..10]

   Compiling function myfib with type NonNegativeInteger -> Any

   (27)  [0,1,1,2,3,5,8,13,21,34,55]
                                                              Type: List Any
                                    Time: 0.22 (IN) + 0.07 (OT) = 0.28 sec
)clear properties myfib

   Compiled code for myfib has been cleared.
-- Define the function p(x) = x^2 - 4 x + 7 such that p(lambda) = 0 for
-- lambda = 2 +- i sqrt(3) and p(A) = [[0 0], [0 0]] for A = [[1 -2], [2 3]]
-- (the lambda are the eigenvalues and p(x) is the characteristic polynomial of
-- A)   [Johnson and Reiss, p. 184]
p(x) == x**2 - 4*x + 7

                                                                  Type: Void
                                                                 Time: 0 sec
p(2 + %i*sqrt(3))

   Compiling function p with type Expression Complex Integer ->
      Expression Complex Integer

   (29)  0
                                          Type: Expression Complex Integer
               Time: 0.43 (IN) + 0.08 (EV) + 0.12 (OT) + 0.03 (GC) = 0.67 sec
p(matrix([[1, -2], [2, 3]]))

   There are 11 exposed and 5 unexposed library operations named +
      having 2 argument(s) but none was determined to be applicable.
      Use HyperDoc Browse, or issue
                              )display op +
      to learn more about the available operations. Perhaps
      package-calling the operation or using coercions on the arguments
      will allow you to apply the operation.
```

```
   Cannot find a definition or applicable library operation named +
      with argument type(s)
                               Matrix Integer
                               PositiveInteger

   AXIOM will attempt to step through and interpret the code.

         +0  0+
   (30)  |    |
         +0  0+
                                                Type: SquareMatrix(2,Integer)
               Time: 0.22 (IN) + 0.03 (EV) + 0.03 (OT) + 0.05 (GC) = 0.33 sec
)clear properties p

   Compiled code for p has been cleared.
-- Define a function to be the result of a calculation
-log(x**2 - 2**(1/3)*x + 2**(2/3))/(6 * 2**(2/3)) _
   + atan((2*x - 2**(1/3))/(2**(1/3) * sqrt(3))) / (2**(2/3) * sqrt(3)) _
   + log(x + 2**(1/3))/(3 * 2**(2/3))


   (31)
                                                                 3+-+2      +-+
      +-+    3+-+2    3+-+    2      +-+    3+-+              (x \|2   - 1)\|3
   - \|3 log(\|2   - x\|2  + x ) + 2\|3 log(\|2  + x) + 6atan(------------------)
                                                                      3
   ------------------------------------------------------------------------------
                                      3+-+2 +-+
                                     6\|2  \|3
                                                     Type: Expression Integer
                           Time: 0.65 (IN) + 0.55 (EV) + 0.13 (OT) = 1.33 sec
function(%, f, x)


   (32)  f
                                                                 Type: Symbol
                                       Time: 0.08 (IN) + 0.03 (EV) = 0.12 sec
expr:= f(y)

   Compiling function f with type Variable y -> Expression Integer

   (33)
                                                                 3+-+2      +-+
      +-+    3+-+2    3+-+    2      +-+    3+-+              (y \|2   - 1)\|3
   - \|3 log(\|2   - y\|2  + y ) + 2\|3 log(\|2  + y) + 6atan(------------------)
                                                                      3
```

```
   -----------------------------------------------------------------------------
                                      3+-+2 +-+
                                     6\|2  \|3
                                                     Type: Expression Integer
                           Time: 0.23 (IN) + 0.22 (EV) + 0.17 (OT) = 0.62 sec
-- Display the top-level structure of a nasty expression, hiding the
-- lower-level details.
name(mainKernel(expr))


   (34)  log
                                                                 Type: Symbol
                                                                  Time: 0 sec
)clear properties expr f

   Compiled code for f has been cleared.
-- Convert the following expression into TeX or LaTeX
y = sqrt((exp(x**2) + exp(-x**2))/(sqrt(3)*x - sqrt(2)))


             +-------------+
             |   2        2
             |  x      - x
             |%e   + %e
   (35)  y=  |-------------
             |   +-+    +-+
            \| x\|3  - \|2
                                            Type: Equation Expression Integer
                           Time: 0.20 (IN) + 0.07 (EV) + 0.05 (OT) = 0.32 sec
outputAsTex(%)

$$
y={\sqrt {{{{e \sp {x \sp 2}}+{e \sp {\left( -{x \sp 2}
\right)}}}
\over {{x \  {\sqrt {3}}} -{\sqrt {2}}}}}}
\leqno(36)
$$

                                                                   Type: Void
                                       Time: 0.07 (IN) + 0.03 (EV) = 0.10 sec
-- ---------- Quit ----------
)quit


real   29.9
user   14.9
```

```
sys    0.7
```

30 Makefile

⟨*⟩≡

```
TANGLE=/usr/local/bin/NOTANGLE
WEAVE=/usr/local/bin/NOWEAVE
LATEX=/usr/bin/latex
LISP=/sei/lisp
MAKEINDEX=/usr/bin/makeindex

all: code doc run

code: wester.pamphlet
${TANGLE} -Rcode wester.pamphlet >wester.lisp

doc:
${WEAVE} -t8 -delay wester.pamphlet >wester.tex
${LATEX} wester.tex 2>/dev/null 1>/dev/null
@{MAKEINDEX} wester.idx
${LATEX} wester.tex 2>/dev/null 1>/dev/null

run:


remake:
${TANGLE} -t8 wester.pamphlet >Makefile.wester
```

References

[1] Wester, Michael J. (ed) ”Computer Algebra Systems” Wiley 1999 ISBN 0-471-98353-5