Miscellaneous

Variables **must** be declared.

`var('y,z')` Declare variables

`x` is a complex variable by default.

`t=var('t',domain='positive')` $t \in (0, \infty)$

`assume(x>0)` $x \in (0, \infty)$

`assume(t,'integer')` $t \in \mathbb{Z}$

`=` Assignment, e.g., $x = 2$ or $f = x^2 + 1$

`==` Logical "is equal to"

`#` starts a comment line

`list=[1,2,3,4,5]` Define a list

`len(list)` Length of `list`

`list[2]` 3rd element of `list`

Constants

`pi` $\pi = 3.14159\ldots$

`e` $e = 2.71828\ldots$

`euler_gamma` Euler's constant, $\gamma = 0.5772\ldots$

`i` Imaginary unit, $i = \sqrt{-1}$

`golden_ratio` Golden ratio, $\phi = 1.618\ldots$

`infinity` Infinity, ∞

`-infinity` Negative infinity, $-\infty$

Numerics

`N(x)` Decimal form (53 bits or ≈ 16 digits)

`N(x,prec=128)` Decimal, 128-bit precision

`N(x,digits=32)` Decimal, about 32 digits

Simplifying

```
factor( x^2-3*x+2 )
expand( (x+3)*(2*x+1)^3 )
(1/(x^2-x)).partial_fraction()
(1/x+1/(x-1)).simplify_full()
```

Equations

```
solve(x^2-3*x+2==0,x)
solve([5*x+6*y==9,3*x-7*y==2],x,y)
find_root(cos(x)-x,0,1)
```

Function evaluation

```
f=x*sin(x); f(x=pi/2)
f=x*y*cos(pi*x*y); f(x=1,y=2)
f(x)=x*sin(x); f(pi/2)
```

Calculus

```
limit(sin(x)/x,x=0)
limit(abs(x)/x,x=0,dir='left')
diff(f,x)
diff(f,x,2)
integrate(g,x)
integrate(g,x,0,5)
numerical_integral(sin(x),0,1)
sum(1/x,x,1,25)
taylor(exp(x),x,0,6)
```

Differential equations

```
t=var('t')
x=function('x')(t)
de=2*diff(x,t,2)+3*diff(x,t)+4*x==cos(t)
desolve(de,x)
```

*With IC's use `desolve(de,x,ics=[0,1,1])`

`laplace(f,t,s)` Laplace transform

`inverse_laplace(f,t,s)` Inverse transform

`plot_slope_field(f,(x,-5,5),(y,-5,5))`

Vectors

```
u=vector([1,2,3]); v=vector([-1,0,3])
```

`u.norm()` Magnitude of u

`u.normalized()` Unit vector

`u.dot_product(v)` Dot product

`u.cross_product(v)` Cross product

```
plot(2*u-3*v,color='red')
```

Basic plotting

```
plot(sin(x),x,-5,5)
plot(sin(x),x,-5,5,axes_labels=['$x$','$y$'])
plot([sin(x),cos(x)],x,-5,5)
plot(3*sin(x),x,-5,5,ymin=-2,ymax=2)
parametric_plot([sin(t),cos(t)],(t,0,pi))
implicit_plot(x^3+4*x*y+y^3==0,(x,-5,5),(y,-5,5))
plot3d(x^2-3*sin(x)*y,(x,-5,5),(y,-5,5))
parametric_plot3d([sin(t),cos(t),t],(t,0,pi))
polar_plot(4*cot(t),(t,0,2*pi))
```

Matrices

```
A=matrix([[2,1],[-1,3]])
```

`A[i][j]` ij-element of A (starting with zeros)

`A.transpose()` Transpose

`A^-1` Matrix inverse

`det(A)` Determinant

`b=vector([1,2]); A\b` Solve $Ax = b$

`A.echelon_form()` Reduced row echelon form