

Basic Operations

```
? foo; /* exact search  $\pi$  */
?? foo; /* inexact search */
%; /* most recent calculated result */
%%; /* '%' in compound statements */
%th(i); /* the  $i$ -th previous output */
load("SomePackage");
demo("SomePackage");
example("SomeFunction");
values; /* a list of all bound user variables */
kill(all); /* removes all bindings */
reset(); /* reset many global variables and options */
to_lisp(); /* from Maxima to Lisp */
(to-maxima) /* from Lisp to Maxima */

display(a, b, c);
disp("a = ", a); /* similar to 'display' but no equations */
print(a, b, c); /* similar to 'display' but all in one line */
```

Data Structures

```
r: [x, y, z];
r[3]; /* z */
r[1] : 2; /* [2, y, z] */
first(r); /* x */
length(r);
member(x, [x, y, z]); /* true */
create_list(x^i, i, [2, 3, 5]); /* [x^2, x^3, x^5] */
make_list(x=y, y, [a, b, c]); /* [x=a, x=b, x=c] */
copylist(r);
append(r1, r2, r3);
join([x, y, z], [a, b, c]); /* [x, a, y, b, z, c] */
cons(w, [x, y, z]); /* [w, x, y, z] */
endcons(a, [b, c, d]); /* [b, c, d, a] */
endcons(a, f(b, c, d)); /* f(b, c, d, a) */
delete(y, [w, x, y, z, y, x, w]); /* [w, x, z, x, w] */
delete(sin(x), x + sin(x) + y); /* x + y */
r2 = push(w, r); /* r2 = [w, x, y, z] is a copy */
pop(r); /* r = [x, y, z], r2 = [w, x, y, z] */

array(aa, 2, 3); /* 2x3 array aa */
arrays; /* all allocated arrays */

a["foo"] : 1; a["foo"]; /* 1 */
a[bar] : 2; a[bar]; /* 2 */

s1: "Hello";
s2: " World!";
concat(s1, s2); /* "Hello World!" */
concat(1, 2) + 3; /* "12" + 3, not a number! */
(a: 5, b: 40);
concat(a, b/2); /* 520 */
c: concat('a, b/2); /* c: a20 */
c:: sqrt(3);
a20; /*  $\sqrt{3}$  */
```

Program Flow

```
# if-else
if sqrt(99) - sqrt(98) > sqrt(98) - sqrt(97) then
  print("No way.")
elseif 99 / 98 > 98 / 97 then
  print("Stupid.")
else
  print("Oh man.");

# for loop
for i : 1 thru 10 step 3 do disp(sqrt(i));
for i : 1 step 3 while i <= 10 do disp(sqrt(i));
for i : 1 step 3 unless i > 10 do disp(sqrt(i));
for i in [1, 4, 7, 10] do disp(sqrt(i));
```

Fundamental Mathematics

```
%pi;
%e;
%i;
%phi; /* (1 +  $\sqrt{5}$ )/2 */
%gamma; /* 0.57721566... */
infinity;

a + b; /* a + b */
a - b; /* a - b */
a * b; /* a * b */
a / b; /* a / b */

a = b; /* a = b */
a # b; /* a  $\neq$  b */
a > b; /* a > b */
a >= b; /* a  $\geq$  b */
a < b; /* a < b */
a <= b; /* a  $\leq$  b */

abs([-4, 1+%i]); /* [-4,  $\sqrt{2}$ ] */
floor(x); /*  $\lfloor x \rfloor$  */
ceiling(x); /*  $\lceil x \rceil$  */
max(x, y, z);
min(x, y, z);

realpart(z);
imagpart(z);
polarform(z); /*  $\|z\|e^{i\theta}$  */
rectform(%e^(%i*t)); /*  $i \sin \theta + \cos \theta$  */

n!;
n!!;
binomial(x, y);
log(x);
exp(x);

cabs(a + b * %i); /*  $\sqrt{a^2 + b^2}$  */
carg(1 + %i); /*  $\pi/4$  */
conjugate(z); /*  $\bar{z}$  */

sin(x); asin(x); sinh(x); asinh(x);
csc(x); acsc(x); csch(x); acsch(x);
cos(x); acos(x); cosh(x); acosh(x);
sec(x); asec(x); sech(x); asech(x);
tan(x); atan(x); tanh(x); atanh(x);
cot(x); acot(x); coth(x); acoth(x);

atan2(y, x); /* arctan(y/x) */
trigexpand(sin(x + y));
trigreduce(-sin(x)^2 + cos(x)^2);
trigsimp(sin(x)^3 + cos(x)^3);

random(100); /* 12 */
random(5.0); /* 3.643515335 */
```

Simplification

```
declare(n, integer); /*  $\sin(n\pi) = 0$  */
opproperties; /* op prop */
features; /* var prop */
remove(n, integer);
assume(x > 0);
facts();
forget(x > 0);

combine(a/2+b/3); /* (2b + 3a)/6 */
expand((x+1)^2); /*  $x^2 + 2x + 1$  */
factor(x^2+2*x+1); /* (x+1)^2 */
radcan(log(x+x^2)-log(x));
coeff(expand((x+a)^6), x, 3);
subst([x=cos(t), y=sin(t)], x*y);
```

Functions

```
# Ordinary function
f(x, y) := exp(x^2 + y^2);

# Memoizing Functions
f[a, b] := b! / (a! * (b-a)!);

# Anonymous Functions
map(lambda([x], %e^(x*pi%i)), [1, 2, 3]);
```

Equations

```
solve(sin(x)^2=1, x);
solve([x^2 + y = 1, x + y^2 = 2], [x, y]);
```

Linear Algebra

```
M0 : matrix([0,0],[0,0]);
I3 : ident(3);
D3 : diagmatrix(3, z);
M3 : genmatrix(lambda([i, j], j - i), 3, 3);
col(M0, 2);
row(M3, 1);

load("eigen");
uvect(x); /*  $x/\|x\|$  */

T : matrix([30, -18, 9],[8, 4, 7],[16, -16, 26]);
eivals(T);
eivects(T);

gramschmidt(T);
ip(f, g) := integrate(f * g, u, a, b);
gramschmidt([1, sin(u), cos(u)], ip), a = -%pi/2, b = %pi/2;

transpose(T);
invert(T);
mattrace(T);
rank(M);
```

Calculus

```
/* Limits */
limit(sin(x)/x, x, 0, plus);
limit(f(n)/f(n+1), n, inf);

/* Differentiation */
diff(log(x));
diff(exp(x*y), x, 2, y, 2);
laplace(exp(x), x, s);

/* Integration */
integrate(sin(x), x);
```

Differential Equations

```
eqn_1: 'diff(f(x),x,2) = sin(x) + 'diff(g(x),x,2);
eqn_2: 'diff(f(x),x) - f(x) = 2*'diff(g(x),x);
desolve([eqn_1, eqn_2], [f(x),g(x)]);

'diff(y,x,2) = exp(x)*x^2;
ode2(%y,x);
```

Sums Products Series

```
sum(1/2^n, n, 1, inf), simpsum;

product(a[i], i, 1, inf);

powerseries(log(x), x, 1);
taylor(log(1-x), x, 0, 6);
taylor(sin(y + x), x, 0, 3, y, 0, 3);
```