

Introduction.

This primer will serve as a introduction to Maple 10 and to earlier editions of Maple. Maple 10 operates in two modes, namely, Document Mode and Worksheet Mode. Within worksheet mode, there are two options, 1D and 2D. The Document Mode is primarily for producing documents in which text is interlaced with "live" mathematical expressions that can be executed. The Worksheet Mode is for computation (with text options). Worksheet option 2D is a bit more user friendly. However, this Primer will introduce you to Option 1D. It uses the Maple language and syntax directly. **The Maple designers recommend that any programming be done in 1D mode.** Also, you will be able to read Maple worksheets written in earlier versions. And going over to the other modes is a snap!

Important: To make sure you are in Worksheet 1D mode, go to the Tools menu and select Options. Select the Display tab. In **Input display**, select **Maple Notation**. The **Output display** should be set on **Typeset Notation**. Click on the **Apply Globally** button to have Maple keep these options everytime you open a session.

For a more complete overview of Maple 10, go to the **Help** menu. Click on **Manuals, Dictionaries, and More**. Then click on **Manuals**. The on-line manual called **User Manual** is very useful.

Section 1. Maple Basics.

When you open a Maple worksheet, the MAPLE prompt $>$ will appear at the top of an almost blank screen. All MAPLE commands are entered at the $>$ prompt. For example, after the given $>$, type $3*4;$ (you must type in the semicolon), then press **Enter**. The display is as follows:

```
>3*4;
      12
```

$>$

MAPLE responds or "echoes" with the answer. It is ready to accept another command at the next prompt.

When in D1 mode, all MAPLE in commands must end with a semicolon or colon.

A semicolon instructs MAPLE to execute the statement **and** display its results on the screen. If a colon is used, the result is computed but will **NOT** be displayed. In what follows, commands that you input will appear after the $>$ symbol (you do *not* type $>$) and MAPLE responses will on the line below your input. If you make a syntax error so that MAPLE cannot execute your command, it responds with an error message. You can edit and correct your command and

press Enter again. **Important:** Maple does not automatically propagate its results forward. So if you are using the result of a computation later on in your worksheet, you must re-Enter all pertinent commands in between. You can do this by simply pressing Enter after each line.

Arithmetic is carried out much like on a calculator. The operations are:

+ and - add and subtract
* and / multiply and divide
^ or ** raise to a power
sqrt square root

For example, to compute $\frac{3+4}{\sqrt{6*.38}}$, type:

```
> (3+4)/sqrt(6*.38);
```

MAPLE will respond with

4.63586320

MAPLE uses the standard precedents in the order of arithmetic operations. If we enter sqrt(2.) --note the decimal point-- MAPLE responds with 1.414213562. However, MAPLE maintains symbolic representation as much as possible. Thus if we omit the decimal point and type

```
> sqrt(2);
```

MAPLE responds with

$\sqrt{2}$

The command **evalf(expr)** is used to obtain the numerical value of an expression that would otherwise be returned symbolically. (The "f" in evalf stands for floating point.) Thus, evalf(sqrt(2)) also returns 1.414213562.

MAPLE has many built in constants. For instance, type the right side (and remember to capitalize!) for:

π Pi
 $\sqrt{-1}$ I

For $e = 2.71\dots$, type exp(1). You will get back the symbol e . But exp(1.) returns 2.7182.. because of the decimal point after the number 1.

MAPLE also has many built in functions. For instance, to get the function on the right, type in the left:

sine, cosine, tangent	$\sin(x), \cos(x), \tan(x)$
e^x	$\exp(x)$
$\ln(x)$	$\ln(x)$
$ x $	$\text{abs}(x)$
inverse sine	$\arcsin(x)$

If we type

```
> sin (exp(3));
```

MAPLE returns

$\sin(e^3)$.

But if we type `evalf(sin(exp(3)))`, MAPLE returns .9444710079. MAPLE also retains the symbolic form for expressions involving the standard functions where possible.

NOTE: To evaluate $5\sin(.3)$, we **must** type

```
>5* sin(.3)
```

The multiplication symbol * must be present where ever multiplication is used or implied.

In MAPLE you can use the results of the previous line by using the percent sign % (or two previous lines by using %%). For example,

```
> 113/38;
```

113/38

```
> evalf(%);
```

2.973684211

```
> sin(%%);
```

$\sin\left(\frac{113}{38}\right)$

```
> evalf(%);
```

.1671205734

Exercises:

1. Use MAPLE to compute the numerical values of the following expressions:

i. $\frac{(13+53)*5}{\sqrt{5}}$

ii. $(-38.3 + 3^{1/4})*\pi$

iii. $\sin(3)$

iv. $e^{-5\ln(17)}$ (Remember the multiplication symbol.)

2. i. Write down a fairly complicated arithmetic expression: _____
and compute its numerical value:_____.

ii. Find the cosine of the value you computed in part (a):_____

iii. Divide the results of part (a) by 17 and compute the natural log of its absolute value_____.

To enter **comments or text** into a MAPLE document you can click on the **T** button in the menu bar. A new line appears with no > prompt. You enter text in that region. However, to return to a command line, you must click on the button labeled with [**>**]. (You can use this same button [**>**] whenever you wish to insert a new command line, even in between other command lines.) Entering text is easy. Do it often!

To **save** your MAPLE session to disc, , pull down SAVE from the file menu, and select an appropriate drive, disk or other space. What will be saved is a text document of your session.

Important:

When you reopen your document to edit or continue your work, the values that you computed **will not** be immediately available. To recompute them, pull down and click on **Execute Worksheet** from the **Edit** menu.

Getting Help!

MAPLE has an extensive, built in, reference manual. You can search for information in at least three ways: by table of contents, by topic, and by selected words. You can get help on getting help by pulling down **Maple Help** from the Help menu! (Try this first.) Click on the cross in front of the topic for explanations.

SHORT CUT If you have, say, forgotten the how to use the diff command and want the help page in a hurry, you can type **?diff** at the > prompt. (You enter the question mark from the keyboard followed by the command or topic about which you need information. Do **not** use a colon or a semicolon.) The Help page will pop up!

Exercises:

1. Find out how to compute the indefinite integral of $\sin^2(x)\cos^4(x)$.
2. Find out how to compute a definite integral.
3. Make up an appropriate MAPLE exercise to demonstrate your ability to differentiate and integrate using MAPLE.

Section 2. Using variables and creating functions.

MAPLE assigns a value or an expression to a variable (a letter or string) with the assignment operator `:=`. (Type a colon followed by an equal sign.) For example:

```
> b := 3;
      b := 3
> b*sqrt(2);
      3√2
> c := x^2 - x^5;
      c := x^2 - x^5
> f := c*sin(y);
      f := (x^2 - x^5) sin(y)
```

Important: If, after we have assigned a numerical value to a letter or string, we wish to return the string to variable status with no number associated to it, use the command `string := 'string'`. (Assign it to itself, enclosed in single quotes.) This allows you to use the variable again, without its previously assigned value.

```
> x := 3;
      x := 3
> x^3;
      27
> x := 'x';
      x := x
> x^3;
      x^3
```

The command `subs(variable = exp1, exp2)` allows us to substitute an expression (*exp1*) or numerical value in for a variable within another expression (*exp2*). (This is one way to create function.) For example, with $f = (x^2 - x^5)\sin(y)$ as above:

```
> subs(x=2, f);
      -28 sin(y)
```

We can substitute for both variables x and y by enclosing them in braces (curly braces always create a set in MAPLE) as follows:

```
> subs({x=7, y = 3}, f);
      -16758 sin(3)
```

```
>evalf(%);
```

```
-2364.889096
```

The value of x is **not** permanently changed with the command `subs`. To change it permanently, we must assign the modified expression to symbol f itself:

> `f:= subs(x=2, f);`

$f := -28 \sin(y)$
 > `f;`
 $-28 \sin(y)$

Creating functions:

If we want to create a function f so that we can evaluate $f(x)$, we type `f:= x -> expr`. (Form the arrow by a dash followed by the "greater than" symbol.) For example:

> `f:= x -> x^2 ;`

$$f := x \rightarrow x^2$$

> `f(2);`
 4

We can create a function of two or more variables as follows:

> `g:= (x, y) -> x*y/(x^2 + y^2);`

$$g := (x, y) \rightarrow \frac{x y}{x^2 + y^2}$$

> `g(3,5);`
 $\frac{15}{34}$

Exercises:

1. Assign the expression $x^3 + 3y^2$ to the variable z .
 - i. Substitute $x=3$ and $y = 4$ into z and evaluate.
 - ii. Substitute the expression $(a + b)$ in for x and $(c - d)$ in for y .
 - iii. Type **expand(%)**. (You will find that Maple has multiplied out your expression.)

2. i) Define the function $f(x) = \frac{\sin(2x)}{x}$. Evaluate it at a few values very near to zero. What

happens at $x = 0$?

ii. Define the function $g(x, y) = xe^{p^y}$. Evaluate it at $(x, y) = (2, 3)$ and obtain both the symbolic and numerical values.

More complicated functions like $f(x) = \begin{cases} x^2 & x > 3 \\ \sin(x) & 0 \leq x \leq 3 \\ x^4 & x < 0 \end{cases}$ can be defined via **procedures**.

Procedures are small programs that are used when your function cannot be expressed with a straightforward formula. (Please refer to the Help program or user manual for further details.) In our example we shall use a conditional "if statement". It must be terminated with **end if**. The procedure itself must be terminated by the words **end proc**. Note the punctuation and how the inequalities are expressed.

```
> f := proc(x)
> if x > 3 then x^2
> else if x <= 3 and x >= 0 then sin(x)
> else if x < 0 then x^4
> end if; end if; end if; end proc;
      proc(x)
          if 3 < x then x^2
              else if x <= 3 and 0 <= x then sin(x) else if x < 0 then x^4 end if end if
          end if
      end proc
```

```
> f(5);
      25
> f(2);
      sin(2)
> f(-2);
      16
```

Alternatively, we obtain a piecewise defined function with the **piecewise** command. The format is as follows:

```
piecewise(cond_1,f_1, cond_2,f_2, ..., cond_n,f_n, f_otherwise)
```

Parameters:

```
f_i      - an expression
cond_i   - a relation or a Boolean combination of inequalities
f_otherwise - (optional) default expression
```

We now redefine f with the **piecewise** command rather than with a procedure:

```
> f:=x-> piecewise(x>3, x^2, x>=0 and x<=3, sin(x), x^4);
```

Note the last expression is not preceded by a condition; it is the expression to be evaluated "otherwise" when the previous Boolean expressions do not hold.

The **piecewise** command will be very important for introducing non-continuous terms into differential equations to be solved by MAPLE.

Exercises:

1. Use a procedure to define the following function:

$$h(x) = \cos(x) \text{ if } -2\pi \leq x \leq 2\pi$$

$$h(x) = 1 \text{ if } x > 2\pi$$

$$h(x) = 0 \text{ if } x < -2\pi .$$

2. Define the same function using the piecewise command.

Section 3. Algebra

In this section we will investigate how to use MAPLE to perform algebraic operations. MAPLE can expand, factor and collect terms, etc. Before we proceed to examples, we will make sure that our variables are cleared of preassigned values. **DOING THIS REGULARLY IS A VERY GOOD HABIT!**

```
> x:= 'x': a:= 'a': b:= 'b': c:= 'c':
```

```
> expand ( (x^2 + y^3)^4);
```

$$x^8 + 4 x^6 y^3 + 6 x^4 y^6 + 4 x^2 y^9 + y^{12}$$

```
> factor ( x^3+ 3*x^2*y+3*x*y^2+y^3);
```

$$(x + y)^3$$

```
> expand((a + b)^2 +(c + 2*b)^2);
```

$$a^2 + 6 a + 45 + c^2 + 12 c$$

```
>collect (% , b) ;
```

$$5 b^2 + (2 a + 4 c) b + a^2 + c^2$$

With the **collect** (*expr*, *var*) command, we have collected on the variable b.

Exercises:

1. Enter the expression $\text{expr} = (3x^2 - axy + 2)^2 + (xy - 5)^3$ and use Maple to multiply it out. (Use **expand**.) Now use the command, `collect (expr, a)` to find the coefficient of a . Do the same for y .

2. Use MAPLE to factor the following. Then expand your results. Is the result always what you began with?

(i) $x^3 - 1$ (ii) $(x^2 - 1)(x^2 + 2x + 1)$ (iii) $(a^4 - 16b^8)$

MAPLE can solve equations and systems of equations symbolically and numerically in so far as they can be solved at all. (It can't solve a general quintic equation either!) The command **solve** (*expr*) sets the expression *expr* to zero and solves it for one of its variables. For example,

> solve (x^2+a*x+b);

$$\{ a = a, x = x, b = -x^2 - ax \}$$

Here MAPLE has chosen to solve for b . To make it solve for the variable a instead, we write:

> solve(x^2+a*x+b, a);

$$-\frac{x^2 + b}{x}$$

To solve several equations for a specific set of variables we enclose the set of equations in braces, and the set of variables in braces. In the following example, for clarity (and possible later editing!) we assign each equation to a variable name. We decided to solve for x and z . You might want to solve for b and y .

> eq1:= a*x + b*y + z = 2;

$$\text{eq1} := ax + by + z = 2$$

> eq2:= c*x + d*y - 3*z = 5;

$$\text{eq2} := cx + dy - 3z = 5$$

> solve({eq1,eq2}, {x, z})

$$\left\{ z = \frac{ady + 2c - 5a - cby}{3a + c}, x = -\frac{3by + dy - 11}{3a + c} \right\}$$

Most equations do not have tractable, closed form solutions. For instance, Maple will solve the cubic $x^3 - 3x + 1$ exactly, giving rather complicated expressions for its three real roots.

```
> solve(x^3 - 3*x + 1);
```

$$\begin{aligned} & \%1^{1/3} + \frac{1}{\%1^{1/3}}, \\ & -\frac{1}{2}\%1^{1/3} - \frac{1}{2}\frac{1}{\%1^{1/3}} + \frac{1}{2}I\sqrt{3}\left(\%1^{1/3} - \frac{1}{\%1^{1/3}}\right), \\ & -\frac{1}{2}\%1^{1/3} - \frac{1}{2}\frac{1}{\%1^{1/3}} - \frac{1}{2}I\sqrt{3}\left(\%1^{1/3} - \frac{1}{\%1^{1/3}}\right) \\ & \%1 := -\frac{1}{2} + \frac{1}{2}I\sqrt{3} \end{aligned}$$

"%1" indicates that you should substitute its value in where ever you that symbol in the expression. The complicated expression you see indicates that MAPLE is perhaps programmed to execute the historic Cardano's formula on the cubic and to maintain the results symbolically. But you want the answer! We use the command **fsolve** (f is for floating point) instead to get the decimal approximation to the answers found above:

```
> fsolve(x^3-3*x+1);
```

```
-1.879385242, .3472963553, 1.532088886
```

When MAPLE produces more than one solution, as above, we can use each of those solutions by making a vector of the solution set and accessing each component individually. We use square brackets to create vectors ($a := [expr]$) and access the components by calling for $a[i]$ (e.g. $a[2]$.) For example:

```
> a:= [ fsolve(x^3-3*x+1)];
```

```
a := [-1.879385242, .3472963553, 1.532088886]
```

```
> a[2];
```

```
.3472963553
```

When there is no solution to an equation or system of equations (or MAPLE can't find one) MAPLE simply fails to echo an output. **IMPORTANT:** Using the **solve** command and its variations is complicated and full of nuances. Please be prepared to use the Help menu frequently.

Exercises:

3. Solve for x in the following:

(i) $x^2 - 3x + 1 = 0$

(ii) $ax + 2by = 3$

(iii) $x^2 + 3x + 1 = x + 2$

iv) $\sin(x) = .4$

v) Use Help to find out how to find the solution to (iv) that lies between 10 and 14.

4. Solve the following systems of equations:

(i) $3x + 2y = 5, x - y = 3$

(ii) $x + 2y + z = 3, x - 2y + z = 7$

(iii) $y = x^2 - 4, y = x + 12$

5. Find all the numerical solutions to $x^3 - 2x + 1$. Assign them to a vector a and solve $a[1]x^2 - a[2]x + a[1] = 0$.

6. Find where the curves $y = x$ and $y = x\sin(x)$ intersect when x lies between 10 and 15.

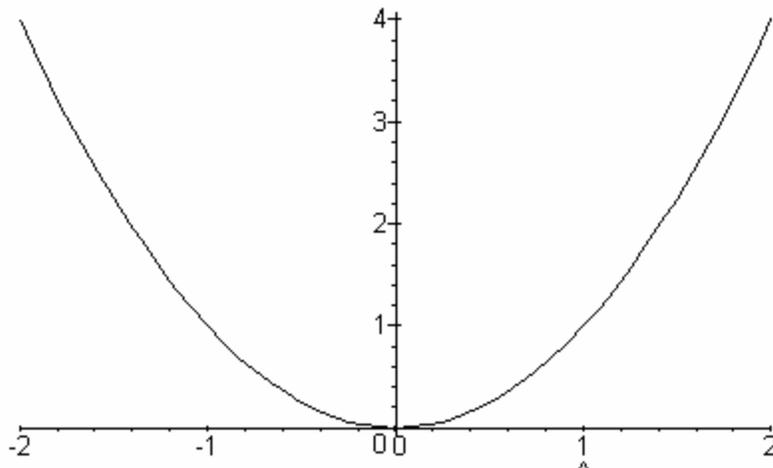
Section 4. Graphing

MAPLE has extensive graphing capabilities but they are not available unless we call them up from the MAPLE library. The graphing program is entitled, "plots." To do this, we use the command, **with(plots)**. (Other capabilities are invoked through other libraries. (Later, we will use **linalg** and **DEtools**, for linear algebra and differential equation commands respectively.) If you enter:

```
> with (plots);
```

MAPLE will echo with the list of commands available from this library. It is long and we will look at just a few. (Remember that you can suppress the list if you use a colon instead of a semicolon.) We begin with the simplest, **plot(expr, horizontal range)**. Here, *expr* can be the formula of the function you wish to plot or it can be the name of a previously defined function. The range of the independent variable (horizontal) must be set.

```
> plot( x^2, x = -2..2);
```



By clicking the arrow near the resulting graph, the options on the menu bar change. You can obtain axis and printing options with these. Note that the box on the upper left tells you the coordinates of the head of the arrow and acts something like a tracer.

In the following example, we define two functions, plot them on the same axis system and take advantage of some of MAPLE's graphing options. (These are numerous. Please consult the Help pages!) We enclose the set of functions to be graphed in braces. We set the horizontal and vertical ranges. We will set the line thickness, its style and provide a title. First, here is an excerpt from the Help page that describes the options available:

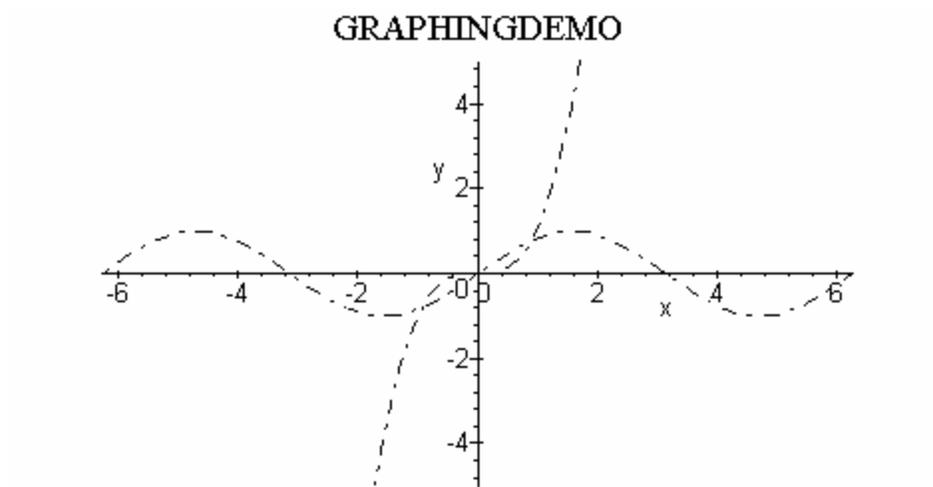
title = t The title for the plot. t must be a character string. The default is no title.

linestyle = n Controls the dash pattern used to render lines in the plot. Styles 0 and 1 give solid lines while greater values select varying dash patterns.

symbol=s Symbol for points in the plot, s is one of BOX, CROSS, CIRCLE, POINT, and DIAMOND.

Example:

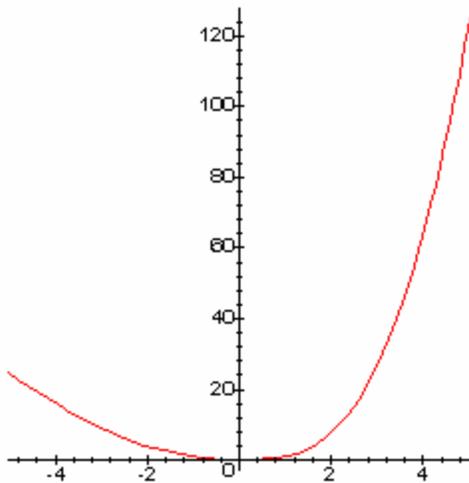
```
> f:=x->sin(x):  
> g:=x->x^3:  
> plot({f(x), g(x)}, x = -2*Pi..2*Pi, y = -5..5, title = GRAPHINGDEMO, linestyle= 4)
```



If the next example we use a procedure to define a function f . To graph it, we enter only the name of the function (not its variable), and we designate its range simply by x0..x1 (again without specifying the variable name). For example:

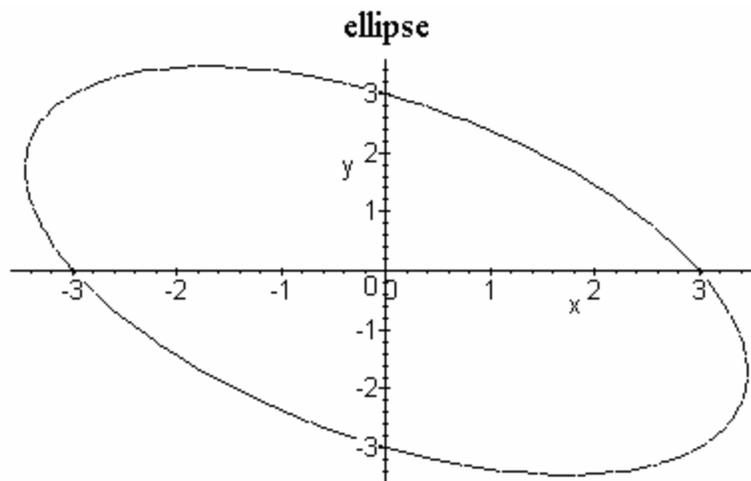
```
> f:=proc(x)  
> if x < 1 then x^2 else x^3  
> fi;end;
```

```
plot(f, -5..5);
```



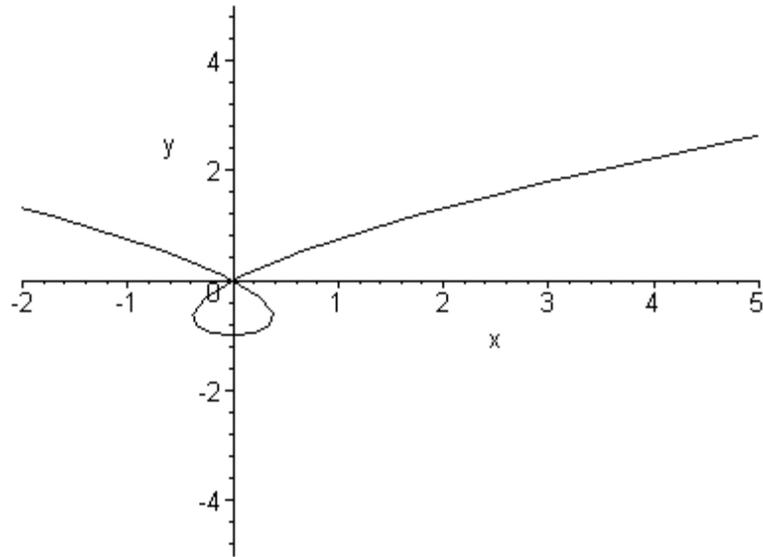
MAPLE can plot implicitly defined curves, curves given parametrically, and curves given in polar coordinates. For example:

```
> implicitplot(x^2+x*y+ y^2 =9 , x=-4..4, y = -4..4, title=ellipse, numpoints = 500);
```



To plot a function give parametrically, we enter the functions for the coordinates into a vector. Note where the range of the parameter is given. In the following example, we plot the function given parametrically by $x = t^3 - t$, $y = t^2 - 1$.

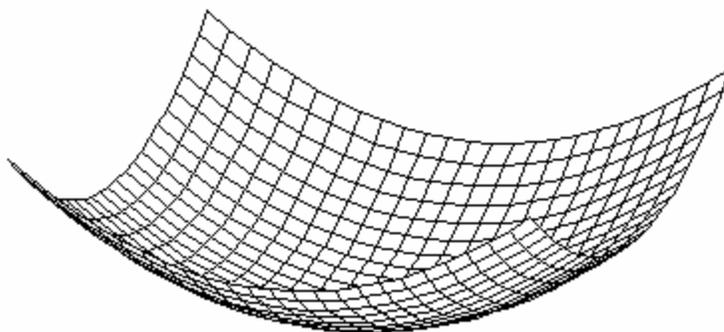
```
> plot([t^3-t,t^2-1,t=-5..5], x=-2..5, y = -5..5);
```



To plot the three dimensional graph of a function of two variables, we use the command **plot3d(expr, x-range, y-range, opts)**. For example:

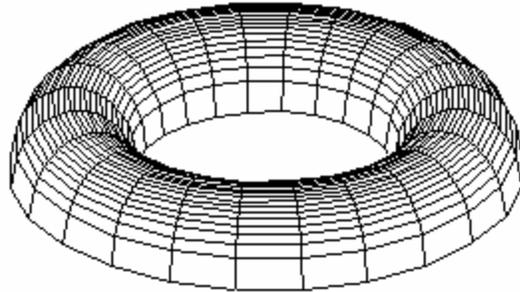
```
> h:=(x,y)-> (x^2+y^2):
```

```
> plot3d(h(x,y), x=-10..10, y=-10..10);
```



As our last example, we show how to plot a surface in 3-space parametrically:

```
> plot3d([s*cos(t), s*sin(t), sqrt(1-(s-3)^2)], s=2..4, t=0..2*Pi);
```



(Note: that the square brackets indicate that MAPLE has produced a vector of points in 3 space for each value (s, t) given.)

Exercises:

1. Graph the line $y = 3x+5$ and the parabola $y = x^2 - 2$ on the same axis system for x in the interval $[-3,3]$.
2. On a graph entitled "My Pet Graph", graph your favorite functions setting both the x and y ranges, and selecting at least two options.
3. Look up the parametric equations for the cycloid. Graph it in the plane.
4. Use Help to learn how to graph in polar coordinates. Graph a cycloid and a four petal rose on the same coordinate system.
5. Use plot3d to graph the function $f(x, y) = x^2 - y^2$. Rotate it (grab a corner with the mouse and pull! To redraw, click on R.) Print your graph. Investigate the options given on the menu bar at the top of your graph.
6. Graph the curve $x^2 - 2xy - y^2 + 2x - 4y = 0$.

Section 5. Calculus commands.

MAPLE is the universal answer key to your calculus problems! It can differentiate, do definite and indefinite integrals, and take limits. We will use the commands **diff(expr, var)** to differentiate, **int(expr, var)** and **limit(expr, var = point)**. In all cases, we use **var** to tell MAPLE which symbol is the independent variable.

```
> f1:=diff(x^3,x);
```

```
>subs(x=5, f1);
```

$$3x^2$$
$$75$$

MAPLE can integrate and differentiate functions that contain symbolic constants. For example,

```
> diff( a*x^2 +sin(x) +3 +b*x, x);
```

$$2ax + \cos(x) + b$$

```
>diff( a*x^2 +sin(x) +3 +b*x,b);
```

$$x$$

In the first line, we told MAPLE to differentiate with respect to x . In the second we told it to differentiate with respect to b (so it assumed that b was the independent variable and x was a constant.) Thus we need no new commands to obtain partial derivatives. For example, in what follows we first obtain $\frac{\partial f}{\partial x}$ and then $\frac{\partial f}{\partial y}$ for the function

$$f(x, y) = x^3 - 3xy^5 + 17y^2.$$

```
> f:= (x, y) -> x^3 - 3*x*y^5 + 17*y^2;
```

```
> diff(f(x,y), x);
```

$$3x^2 - 3y^5$$

```
> diff(f(x,y), y);
```

$$-15xy^4 + 34y$$

To find the second derivative, we iterate `diff(diff(expr, var), var)` as shown in the next examples. First we find the second derivative of $x^3 - 3ax^4$. Then we find the mixed partial of x^3y^5 .

```
> diff(diff(x^3 - 3*a*x^4, x), x);
```

$$6x - 36ax^2$$

```
> diff(diff(x^3*y^5, x), y);
```

$$15x^2y^4$$

```
> subs({x=3, y=5}, %);
```

84375

We can also obtain higher derivatives (perhaps mixed) by listing the variables in the order we want them differentiated. Thus `diff(f(x,y), x, x)` is the second derivative with respect to x of the function f and `diff(f(x,y), x, y)` is the mixed partial.

In the preceding discussion we have differentiated expressions and MAPLE has returned expressions. If we start with a function $f(x)$ and want the function $f'(x)$ returned, we use the differential operator command **D**. The command `D(f)` returns the function f' and the command `D(f)(x0)` returns the function f' evaluated at x_0 . We use `(D@@2)(f)` or `(D@@3)(f)` for second and third (etc.) derivatives. (The symbol @@ denotes repeated composition in MAPLE.) For example:

```
> f:=x->x^3;
```

```
> D(f);
```

$$x \rightarrow 3x^2$$

```
> D(f)(2);
```

12

```
>(D@@2)(f);
```

$$x \rightarrow 6x$$

For functions of more than one variable, we indicate which partial we want **D** to obtain by giving the number of the variable enclosed in square brackets. In the following, because x is listed first in the definition of the function g , it is designated as [1], and similarly y is designated by [2].

```
> g:= (x,y) -> x^3*y^5;
```

$$g := (x, y) \rightarrow x^3y^5$$

```
> D[1](g);
```

$$(x, y) \rightarrow 3x^2y^5$$

The procedure for integrating is very similar. For an indefinite integral, we must simply indicate the variable. For an definite integral over the interval [a, b], we indicate the interval over we integrate by var = a..b.

> int(a*x^2, x);

$$\frac{1}{3} a x^3$$

> int(a*x^2, x=1..2) ;

$$\frac{7}{3} a$$

MAPLE can integrate all of your favorites, including expressions that involve integration by parts or partial fraction decomposition. (But you must add the constant c.) For example:

>int(3*x/(x^2-x-2),x);

$$\ln(x + 1) + 2 \ln(x - 2)$$

> int(x^2*exp(x),x);

$$x^2 e^x - 2 x e^x + 2 e^x$$

But some expressions do not have antiderivatives in terms of familiar functions. In this case MAPLE, returns an unevaluated integral. Even if we supply limits of integration, the answer is returned symbolically. To obtain a numerical answer, we apply **evalf** to the expression.

> int(sin(x^(4/3))^3, x=0..1) ;

$$\int_0^1 \sin(x^{(4/3)})^3 dx$$

> evalf(%)

$$.1442909931$$

MAPLE can perform improper integrals and double integrals over non-rectangular regions.

> int(1/x^2, x=1..infinity);

$$1$$

> int(int(x*y, y = 0..(x+1)), x=0..1);

$$\frac{17}{24}$$

We conclude this section by computing some familiar limits.

$$> \text{limit}(x^2, x = 4);$$
$$16$$

$$> \text{limit}(\sin(x)/x, x=0);$$
$$1$$

$$> \text{limit}(\ln(x)/x, x = \text{infinity});$$

$$0$$

$$> \text{limit}(\text{floor}(x), x = 1);$$

undefined

$$> \text{limit}(\text{floor}(x), x=1, \text{right});$$

$$1$$

$$> \text{limit}(\text{floor}(x), x = 1, \text{left});$$

$$0$$

In the last three examples, first we took the limit of the greatest integer function (which is called by "floor(x)" in MAPLE) which is undefined. Then we took it's limit from the right and left.

Exercises:

1. Use the **diff** command to find the expressions for the first and second derivatives of each of the following :

$$\text{i) } x e^{-x^2} \quad \text{ii) } \frac{\sqrt{x^3 - 5} - \sin(2x)}{x^5 + 3} \quad \text{iii) } x \arctan(x^2 - 3)$$

2. Use the differential operator command **D** to find $f'(x)$ for each of the following:

$$\text{i) } f(x) = \frac{\sqrt{x^3 - 5} - \sin(2x)}{x^5 + 3} \quad \text{ii) } f(x) = \frac{x^3 - 1}{x^2 + 3}$$

3. Find the critical points of each of the following. Use the second derivative test to determine which are local maximums or minimums. (Use the command **solve**.)

$$\text{(i) } f(x) = x e^{-x^2} \quad \text{(ii) } g(x) = x \ln(x) \quad \text{(iii) } \frac{x}{x^2 + 1}$$

4. Compute the following integrals (use evalf where necessary). Note for the improper integral, insert the word "infinity" for the symbol ∞ .)

$$\text{i) } \int x e^{-3x} dx \quad \text{ii) } \int \frac{x+1}{x^3-1} dx \quad \text{iii) } \int_0^1 \sqrt{x^3+3} dx \quad \text{iv) } \int_0^{\infty} \frac{1}{1+x^2} dx$$

5. Compute the following limits:

$$\text{i) } \lim_{x \rightarrow 0} \frac{\tan(3x)}{\sin(5x)} \quad \text{ii) } \lim_{x \rightarrow 0} \ln\left(1 + 3x\right)^{\frac{1}{x}} \quad \text{iii) } \lim_{x \rightarrow \infty} \frac{\ln(x)}{\sqrt{x}}$$

6. Remember those max-min problems that you could set up, but the algebra got you down? Find them in your old calc book and do a few.

Section 6. Differential Equations

Part 1. First order differential equations.

First, we demonstrate how to express an differential equation in a form that MAPLE can recognize and solve.

The mathematical notation $\frac{dy}{dx}$ tells us that we have a function $y(x)$ that is differentiated with respect to the independent variable x . (There may be other symbolic constants in our expression, but the notation makes it clear which are variables.) The MAPLE notation carries that same information even more explicitly. For $\frac{dy}{dx}$ we write, `diff(y(x), x)`. The differential equation $\frac{dy}{dx} = 3x^2 + y$ is expressed in MAPLE as

$$\text{diff}(y(x),x) = 3*x^2 + y(x).$$

Note that every time we use the dependent variable y , we must indicate its dependence on x . Of course, we can use any pair of coordinate variables, so that the differential equation $\frac{dx}{dt} = x$ becomes:

$$\text{diff}(x(t), t) = x(t)$$

Exercises:

1. What differential equation is expressed by the MAPLE code `x*diff(z(x), x) = z(x) - 3*sin(x)` ?
2. Express the differential equation $\frac{dx}{dt} = \frac{x^2}{1+t^2}$ in MAPLE code.

The principle tool for solving differential equations in Maple is the command **dsolve**. With it, we can solve for the general solution to a differential equation or we can specify initial conditions to obtain a particular solution. It will return an explicit solution, with appropriate arbitrary constants where possible. If not, it will return an implicit solution (if possible!). Or it will solve an equation numerically if given initial conditions.

We begin by finding the general solution to a simple differential equation. MAPLE returns its arbitrary constants in the form `_Ci` (The initial dash is part of the constant name). For

convenience, we first write the differential equation and assign it to the name, eq1. We assign the solution to the equation to the name gsol.

```
> eq1:=diff(y(x),x)=x+y(x);
```

$$eq1 := \frac{\partial}{\partial x} y(x) = x + y$$

```
> gsol:= dsolve(eq1, y(x));
```

$$gsol := y(x) = -x - 1 + e^x _C1$$

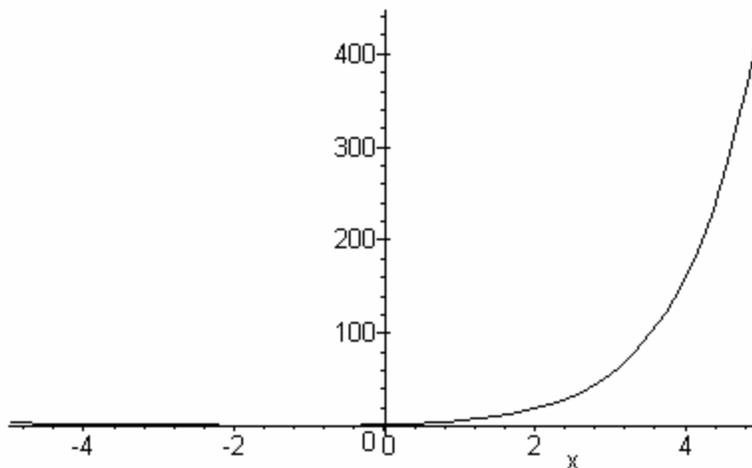
Note that the term e^x is multiplied by the constant $_C1$. Now we wish to substitute in for $_C1$ and plot a particular solution. We use the command **rhs** to extract the "right hand side" of the equation stored to gsol. We substitute 3 in for $_C1$ and we then assign the particular solution to the name psol. Finally we graph it.

```
> psol:=subs(\_C1=3,rhs(gsol));
```

$$psol := -x - 1 + 3 e^x$$

```
> with (plots):
```

```
>plot(psol,x=-5..5);
```



In the next example, we look at the equation $\frac{dx}{dt} = t^2 x^3$. It is simple enough to do by hand.

Separating and integrating, we get $\frac{1}{x^2} = \frac{-2t^3}{3} + c$. To solve for x explicitly, we have two

choices, giving us two branches to the solution: $x(t) = \pm \left(\frac{-2t^3}{3} + c \right)^{-1/2}$. Maple expresses the

solution as follows (varying our algebra slightly).

```
> eq1:=diff(x(t),t)=t^2*(x(t))^3
```

$$eq1 := \frac{\partial}{\partial t} x(t) = t^2 x^3$$

```
> sol := dsolve(eq1, x(t));
```

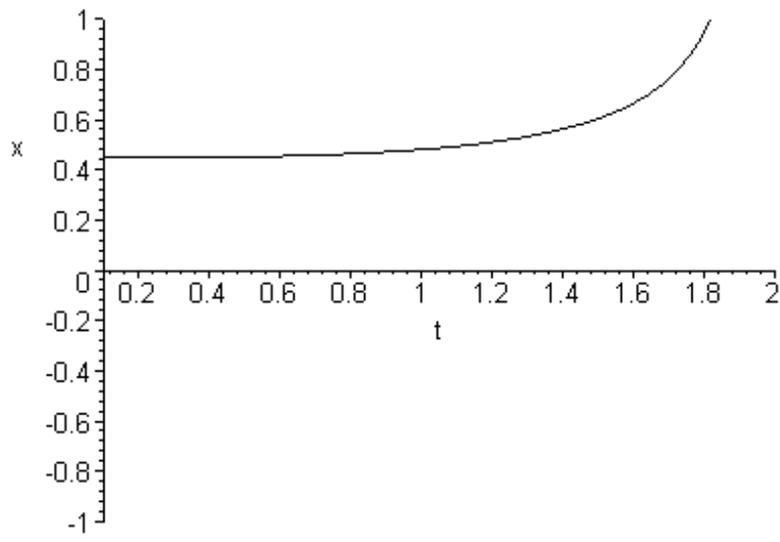
$$sol := x(t) = 3 \frac{1}{\sqrt{-6 t^3 + 9 _C1}}, x(t) = -3 \frac{1}{\sqrt{-6 t^3 + 9 _C1}}$$

The two branches are stored to the vector `sol`, the first in `sol[1]` and the second in `sol[2]`. (Note: the clue that `sol` is a vector is the comma in the solution.) In what follows we substitute 5 in for `C1` in the first branch, save it to `plot1`, and graph.

```
> plot1:=subs(_C1=5,rhs(sol[1]));
```

$$plot1 := 3 \frac{1}{\sqrt{-6 t^3 + 45}}$$

```
> plot(plot1, t=0.1..2, x=-1..1);
```



>

Exercises:

3. Use the command **dsolve** find the general solution to each of the following differential equations.

i) $\frac{dy}{dx} = \frac{-y}{x}$ (ii) $\frac{dx}{dt} = x + \sin(t)$ (iii) $x \frac{dy}{dx} + 3y = e^{-x}$ (iv) $\frac{dx}{dt} = t^4 x^4$

4. Substitute in for **_C1** in each of the solutions obtained in question 1 above, and plot several solution curves for each equation.

We can stipulate initial conditions within the **dsolve** command. Initial conditions are stipulated as set of equations grouped with the differential equation within curly braces.

```
> dsolve({diff(y(x),x)= y(x)+x, y(0)=2}, y(x));
```

$$y(x) = -x - 1 + 3 e^x$$

One of the most important aspects of using MAPLE to solve differential equations is that we can use it to produce numerical results when there is no convenient expression for a solution. It is MAPLE's default for cases when it cannot produce a closed form solution. Also, we can set the parameter, **type = numeric**, when we prefer a numerical solution. (**Note**: initial values must be provided and the differential equation cannot contain symbolic constants.)

MAPLE cannot respond with an "echo" that contains the numeric data. It responds with a procedure that we can use like a function to evaluate the solution at designated points. In the next example we solve a logistic equation numerically. **IMPORTANT**: To access results, when using **type = numeric**, you must assign the results of **dsolve** to a variable name. In what follows we save the results to the name **LG**. Also, you must stipulate initial conditions.

```
> LG:=dsolve({diff(y(t), t)= .0025*(200-y(t))*y(t), y(0)= 10}, y(t), type = numeric);
```

```
LG := proc(rkf45_x) ... end
```

```
> LG(0);
```

$$[t = 0, y(t) = 10.]$$

```
> LG(10);
```

$$[t = 10, y(t) = 177.3016557149294]$$

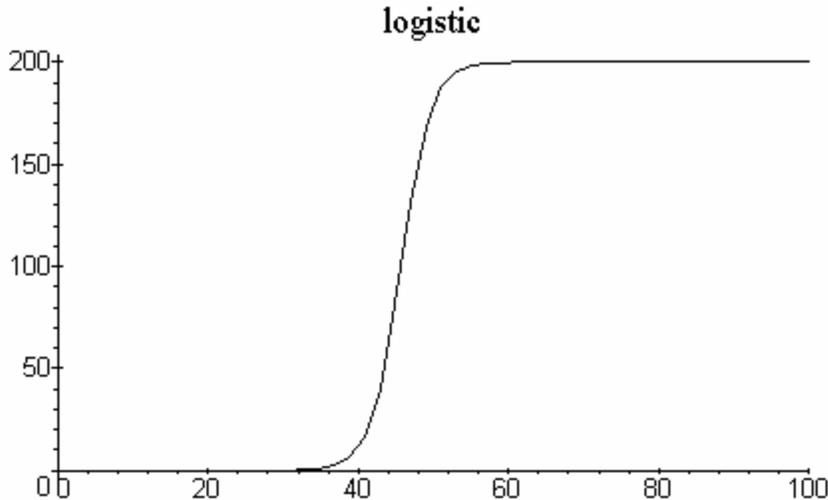
```
> LG(100);
```

$[t = 100, y(t) = 199.9999999811010]$

We have asked MAPLE for the value of the solution at several values of t . Note, that as t increases, the numeric values approach 200 as expected. To **graph**, the results we use the command **odeplot**(*name*, [*vars*], *range*, *options*).

> with (plots):

>odeplot(LG, [t, y(t)], 0..100, title= logistic);



Exercises.

5. Solve the following initial value problems and graph your solution.

$$t \frac{dx}{dt} + 2x = e^t \text{ when } x(1) = 5.$$

6. Solve the following numerically. Use **odeplot** to graph your solution.

$$\frac{dy}{dx} - \frac{y}{x} = x^{\frac{1}{2}} \text{ through the point } [1, 2].$$

Linear Algebra

When we do linear algebra, we use matrices and vectors and we “operate on them” in various ways such as multiplying them, transposing them, finding related matrices and vectors, etc. The

following is a brief overview of how to do Linear Algebra with Maple. The first thing to do is load the Linear Algebra package.

```
> with(LinearAlgebra);
```

A. Constructing Matrices and Vectors.

We show how to construct the matrix $A = \begin{bmatrix} 1 & -2 & 5 & 1 \\ 2 & 3 & 4 & 7 \\ 1 & 2 & 6 & 9 \end{bmatrix}$ in several ways.

First, you can use the **Palette** on the left of the worksheet. Enter `A:=` at the prompt. Click on **Matrix** to open the dialog box. Enter 3 in **rows** and 4 in **columns**. Initially, the other buttons should be set at Type: **Custom Values** (meaning you enter them), Shape:**Any** (meaning you determined the dimensions) and Data Type: **Any** (meaning you enter values). Then click on **Insert Matrix**. A matrix appears at the cursor with entries of the form $m_{i,j}$. Highlight each of these entries and replace them with the appropriate values. Hit Enter to check that you have entered you values correctly. Now, explore the buttons to see the Palette can help you construct other matrices. For instance, if the dimensions are set to 3 by 3 and Type is set to **Identity**, the 3 by 3 identity matrix is returned.

In 1D mode, the Palette construction displays the Maple notation for entering a matrix directly, which we now discuss. The Matrix constructor command is **Matrix** (*options*).

Here are some examples to try.

- i. `Matrix(3)` yields a 3×3 matrix filled in with 0s.
- ii. `Matrix(2,3)` yields a 2×3 matrix filled in with 0s.
- iii. `Matrix(2, 3, 5)` yields a 2×3 matrix filled in with 5s.
- iv. `Matrix(2,3,m)` yields a 2×3 matrix filled in with $m(i, j)$.

To fill in the values in a matrix directly, list its rows within square brackets, separated by commas. Note the set of brackets that encloses the list of rows.

```
>Matrix( [[1,2,3], [4,6,7]]);
```

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 6 & 7 \end{bmatrix}$$

The instructions `Matrix(3,4,[[1,2,3],[4,5,6]])` fills the extra spaces in with 0s.

```
>A:= Matrix( 3,4,[[1,2,3],[4,5,6]]);
```

```

      é1  2  3  0 ù
      è   ú
A := è4  5  6  0 ú
      ê   ú
      ë0  0  0  0 ù

```

We can also fill in a matrix with entries that are a function of its position names. For example:

```

> f := (i,j) -> 3^(i+j-1):
> Matrix(2,f);

```

```

      é3  9ù
      ê   ú
      ë9 27ù

```

On the other hand, we can fill a matrix in with random values:

```

      é27  99  92ù
      è   ú
      ê 8  29 31ù
      ë   ú
      ë69 44  67ù

```

Note: Matrices bigger than 10 by 10 are not displayed. Instead a place holder is displayed.

Click on the placeholder to browse the entries the matrix. The placeholder is not interactive.

You can not enter values through it.

```

      é 12 x 12 Matrix ù
      è   ú
      ê Data Type: anything ú
A := è   ú
      ê Storage: rectangular ú
      ê   ú
      è Order: Fortran_order ù

```

A column vector is an n by 1 matrix and a row vector is a 1 by n matrix. So we can enter vectors with the Matrix command. We can also use the **Vector** command. The default is a column vector.

```

> v := Vector([1,2,0]);

```

$$v := \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$$

```
> v:=Vector[row]([1,2,0]);
      v := [1 2 0]
```

There are some handy short cuts **Short Cuts.**

$\langle a \mid b \mid c \rangle$ constructs an object by columns.

Examples:

Construct a column Vector

```
> < 1, 2, 3 >;
```

$$\begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

Construct a row Vector

```
> < 1 | 2 | 3 >;
```

$$[1 \ 2 \ 3]$$

Construct a Matrix by columns

```
> A:=< < 1, 2, 3 > | < 4, 5, 6 > >;
```

$$A := \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix}$$

Augment Matrix A with a column Vector $\langle x,y,z \rangle$

```
> < A | < x, y, z > >;
```

$$\begin{pmatrix} 1 & 4 & x \\ 2 & 5 & y \\ 3 & 6 & z \end{pmatrix}$$

Construct a Matrix by rows

```
> < < 1 | 2 | 3 >, < 4 | 5 | 6 > >;
      é1 2 3 ù
      ê   ú
      ë4 5 6 ù
```

Construct a 1 x n Matrix from 2 row Vectors

```
> < < 1 | 2 | 3 > | < 4 | 5 | 6 > >;
      [1 2 3 4 5 6 ]
```

B. Operations on Matrices and Vectors

Most of the operations are intuitive. Use +, -, and exponentiation as usual. But multiplication is not done as you would expect. **Scalar** multiplication use * and **matrix –matrix** or **matrix-**

vector multiplication uses .(a period).

```
> a:=<<1,2,3>|<-3,1,0>>; b:=<1,-1>;
Example:
```

```
      é1 K 3 ù
      ê   ú
a := ê2  1 ú
      ê   ú
      ë3  0 ù
```

```
      é  1 ù
b := ê   ú
      ëK 1 ù
```

```
> 3*a; 3*b;
```

```
      é3 K 9 ù
      ê   ú
      ê6  3 ú
      ê   ú
      ë9  0 ù
```

```
      é  3 ù
      ê   ú
      ëK 3 ù
```

```
> a.b;
```

```
      é4 ù
      ê  ú
      ê1 ú
      ê  ú
      ë3 ù
```

The Linear Algebra Package includes many commands. Here are a few that are most commonly used. There are many more!!

GaussianElimination -- Reduces a matrix to row echelon form.

MatrixInverse – Computes the inverse of a square matrix or the pseudo-inverse of a non-square matrix.

Eigenvectors – Computes the eigenvalues and eigenvectors of a matrix.

ReducedRowEchelonForm – finds the reduced row echelon form of a matrix.

LinearSolve –

Solves the system $A \cdot x = b$.

Examples:

```
> a := <<1, 2, 3> | <-3, 1, 0>>; b := <1, -1>; c := <-2, 3, 3>;
```

```

      é1 K 3ù
      è      ú
a := è2      1ú
      ê      ú
      ë3      0ù

```

```

      é 1ù
      è      ú
      ëK 1ù

```

```

      éK 2ù
      è      ú
c := è 3ú
      ê      ú
      ë 3ù

```

```
> 3*a; 3*b;
```

```

      é3 K 9ù
      è      ú
      è6      3ú
      ê      ú
      ë9      0ù

```

```

      é 3ù
      è      ú
      ëK 3ù

```

```
> a.b;
```

$$\begin{pmatrix} 4 \\ 1 \\ 3 \\ 3 \end{pmatrix}$$

> a^%T;

$$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 0 \end{pmatrix}$$

> GaussianElimination(a);

$$\begin{pmatrix} 1 & 3 \\ 0 & 7 \\ 0 & 0 \end{pmatrix}$$

> ap:=MatrixInverse(a);

$$ap := \begin{pmatrix} \frac{7}{139} & \frac{21}{139} & \frac{30}{139} \\ \frac{41}{139} & \frac{16}{139} & \frac{3}{139} \end{pmatrix} \text{ pseudo-inverse is computed.}$$

> ap.a;

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

> m:=<<1,1>|<2,1>>;

$$m := \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix}$$

> Eigenvectors(m);

$$\begin{pmatrix} 1 & \sqrt{2} \\ 1 & \sqrt{2} \end{pmatrix}, \begin{pmatrix} \sqrt{2} & 1 \\ 1 & 1 \end{pmatrix}$$

The eigenvalues are returned in the first vector. The eigenvectors are the columns of the second

> ReducedRowEchelonForm(m);
matrix.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

> LinearSolve(a,c);

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Exercise

Construct a 3 by 3 matrix A with random entries. Construct a vector b of length 3 with random entries.

1. Compute $3A$, $-5b$ and Ab and $b^T A$.
2. Solve the system $Ax = b$ in the following ways:
 - i. Augment A by b and use ReducedRowEchelonForm
 - ii. Use Linear Solve.
3. Repeat the exercise 2 in the case that A is a 2 by 3 matrix.
4. Find the Eigenvectors of $A := \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}$ using the Eigenvector command.
5. Remember that any scalar multiple of an eigenvector is also an eigenvector. Let $b = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$, which is not an eigenvector of A. Use b to compute $A^n \times b$ for large powers of n. Which eigenvector of A do you approximate? Notice this is a way to obtain an eigenvector without using a determinant.

A Look at Programming

Much basic programming is done with very few statements: the “for”, the “if”, and the “while” statements. What follows uses all three. It’s just a glimpse. The help pages are where you will learn this.

Here is the simple problem we will solve with programming.

.We construct a vector named *vec* of length *n* and we fill it with random integer entries. (The entries and the results will change every time you re-execute the steps.) Using the basic programming commands, we

1. Change the values of *vec*, dividing each by $n - i + 1$ using a “for” statement.
2. Sum the entries of *vec* up until the they just exceed 20 using a “while” statement.
3. Print out different results depending on whether the sum did indeed exceed 20 using an “if” statement.

First we assign a value to *n*, the length of the vector “*vec*”. Then we construct a vector with random integer values. It will return a different vector every time you re-enter the command.

```
> n:=100;vec:=RandomVector(n);
```

```
n := 100
```

```
vec := é 1 .. 100 Vectorcolumn ú  
ê  
ê Data Type: anything ú  
ê  
ê Storage: rectangular ú  
ê  
ê Order: Fortran_order ú
```

Next, we divide the *i*th component of *vec* by the $n - i + 1$. We do this with a “for” loop. Notice that there is no punctuation after the word “do.” The sequence of steps is then separated by colons or semicolons. The sequence must end with the words “end for” so as to indicate the end of the program. Ending the sequence of expressions with a colon suppresses the output. Try it with a semicolon.

```
> for i from 1 to n do  
> vec[i]:=vec[i]/(n-i+1); end do:
```

>

Now we will sum up the components of `vec` up to when the sum exceeds 10. This is done here with a "while" statement. Again, no punctuation after "do." The sequence of statements ends with "end do". The sum is assigned to the variable "sumvect". Notice that we initialize `i` and `sumvect` before we start the while statement.

```
> sumvect:=evalf(vec[1]):i:=1:
> while sumvect < 10 and i<= n-1 do i:=i+1:
sumvect:=evalf(sumvect+vec[i]):end do:
```

Finally, we want to print out the sum and the position it at which the summing is terminated. We also want to know if the sum fails to exceed 20. We do this with an "if" statement that determines what to print (using `printf`) depending on the output. Note: what is printed here will be different each time the sequence is executed because we have used random entries.

```
> if sumvect > 10 then printf("At position %a the sum
exceeds 10 and the value of the sum is %a",i,sumvect);
else printf("The sum of all components is less than
10."); end if;
```

>

>

```
At position 56 the sum exceeds 10 and the value of the
sum is 10.08088985
```

Note: in the `printf` statement, the symbol "%a" is a place holder for the variables which are listed at the end.

Exercises

1. Chose a large integer and assign it to the variable n . Write a program that finds the number of prime numbers less than or equal to n and then lists them in order. Use the `isprime(exp)` command to determine if a given integer is prime or not. The command `isprime(exp)` returns *true* or *false*. Your program should use a **for** statement and an **if** statement. Print out the number of primes with a **printf** command.
2. The polynomial $x^5 + 2x - 1$ has no rational roots. But the Mean Value Theorem guarantees that it has a root between 0 and 1. Use a **while** statement and the interval bisection method to construct a program to find a rational approximation to the root that is within 1/1000 of the actual root. (Do not use the `fsolve` command!)

