A Maple Tutorial

David Guichard

July 11, 1997

(modified by Albert Schueller, Jan 2011)

(modified by Douglas Hundley, Jan 2013)

Contents

---

1 Getting Started

Maple is a powerful mathematical calculator, often called a computer algebra system (CAS) or symbolic mathematics program, because unlike most calculators it can do algebra and calculus symbolically. For example, it can expand the product $(x+1)^{30}$ into the normal representation of this polynomial, and it can factor the result back into the original form. It can compute the derivative or antiderivative of $3x/\sqrt{x^2+2x+47}$. It can also serve as a numerical calculator, but again unlike most calculators it can do integer and rational number arithmetic exactly, no matter how many digits are involved. If it is undesirable or impossible to express a result exactly, Maple can do numerical approximation. Finally, Maple can do some quite sophisticated graphing in both 2 and 3 dimensions.

To learn how to use some of Maple's features, you should work through this tutorial step by step. What you should type will be shown in the left hand column in `typewriter style`; comments and explanations will appear at the right. You should do everything listed here, but you should feel free to experiment too. If you leave something out or do something extra it may change the results of some later computations. (If you want to experiment along the way, try using different names for functions and variables than the ones in the examples.)

To start Maple, log in on one of the workstations in the math lab. If the Maple icon is not on the launchpad on the left, then go to upper left icon (Dash Home) and search for Maple. Drag the icon over to your launchpad (NOTE: If Maple is running, you won't be able to drag it to the launchpad).

Once Maple has started, it is useful to set some options so that they are the default options:

- We will always work in `Worksheet` mode, NOT in `Document` mode. You can tell that you are in `Worksheet` mode because you will see the red cursor: `>`

  When the Maple program begins, select `Start with a Blank Worksheet`.

- To make some options default, go to: `Tools`, then `Options`
  - Go to the `Display` tab, and set `Input display:` to `Maple Notation`
  - Go to the `Interface` tab, and choose: `Default format for new worksheets:`, and select `Worksheet`

- Select the `Apply Globally` button at the bottom.

- You might close, then re-start Maple to see that the options took (that is, Maple should automatically start in `Worksheet` mode).

- Hint: You can shut off the somewhat annoying startup dialog box once you get the defaults set- Uncheck the little box in the lower left corner: Show this dialog on startup.

- Hint: You can close all the buttons on the left for the time being- We don't want to use the shortcut keys until we understand what is behind them. Use the small triangle key on the divider to shut them off temporarily.

Side Remark: The new `Document` interface is rather nice- It enables you to create documents with live, embedded mathematics. That is more than we need- We will use Maple to do symbolic and graphical tasks, then incorporate those answers into our LaTeX code for the lab write-ups. However, if you're feeling bored sometime, you might check out Maple's ability to create cool looking documents.

2 Algebra

2.1 Numbers

```
2+3;
123456789*987654321;
```

You must type the semicolon at the end of the line to signal the end of your expression.

```
2+3
34*98
-45;
```

Type this as is, without the semicolon on the first line. Maple will warn you about the missing semicolon, but will do the calculation anyway. Sometimes you may want to start a new line as part of the same expression; you can do this by holding the shift key while you press return. Do this after the "98".

```
2/5 + 1/3;
23/45-167/345*2/3;
```

All rational arithmetic is done exactly.

```
2^5;
2**5;
3^27;
ifactor(%);
```

Maple knows standard mathematical notation, including these two symbols for exponentiation. The percent sign (`%`) denotes the previous result (it's called the 'ditto' operator). The command `ifactor` does integer factoring; the command `factor` factors polynomials.

```
12345/98765;
evalf(%);
```

You can force Maple to approximate values as 'floating point numbers' (i.e., using decimal notation). Some of Maple's commands do this automatically, as we will see later.

```
I^2;
evalf(Pi);
evalf(exp(1));
```

The number $i = \sqrt{-1}$ is denoted by I; the case is significant—a lower case 'i' is different. Pi denotes π; again case is significant. It is very easy to mistakenly type 'pi' instead of 'Pi'. When you do, things will almost certainly not work properly, but you will probably get no error messages—you just have to watch out for this. There is no built-in symbol for $e \approx 2.71828$; use the function `exp(x)` whenever you need e^x.

2.2 Variables and Expressions

Maple can manipulate expressions involving variables as well as numbers. Variables can be assigned values, either actual numeric values or entire expressions.

```
(x+5)^2;
f:=%;
x:=5;
f;
x^2;
```

Maple simply repeats the first line since x has no value. The second line saves the expression $(x+5)^2$ with name f; the ":=" symbol acts like an equal sign in normal mathematics or the ":=" in the Pascal programming language. The third line assigns x the value 5.

```
x:='x';
x;
f;
```

This clears the value of x—this is not the same as setting it to zero! Note that Maple remembers the 'real' definition of f—it is still an expression involving x, not the fixed value 100.

```
2x;
2*x;
```

You must always type the multiplication symbol—Maple will not accept the first line even though it is standard notation. If you make a simple typing mistake like this, you can fix it without retyping the command. In this case, Maple should correctly guess where the problem is and place the cursor between the 2 and the `x`; simply type the `*` and press return. In any case, you can always move the mouse pointer to the position of the mistake and click the left button, then make your corrections. You can adjust your position in the line using the arrow keys to the right of the main keyboard. When you press the return key, the command will be re-evaluated.

```
expand(f);
factor(%);
factor(x^2+4*x+5);
```

If a polynomial doesn't factor, Maple leaves it alone. (As a general rule, if Maple cannot perform a calculation it will simply restate the command.)

WARNING: You can go back anywhere in your Maple session and change a command. When you do this you change *only* the result of that command—you do not change any subsequent commands that depend on the result. If you want to re-execute subsequent commands, you must move the cursor to each command in turn and press the return key. Do not overuse this editing ability—it can make the chain of events very difficult to follow. It often is better to copy the command, paste it in at the end of the session, modify it there, and execute it as a new command.

To copy and paste a command, do this: Position the mouse pointer at the beginning of the command you want to copy, press the left mouse button and hold it down, move the mouse cursor to the end of the command, and release the mouse button. (If the command is on a single line, you may also select the whole command by positioning the cursor in the line, and clicking the left mouse button three times quickly—this is a "triple click".) The command should now be highlighted. Position the mouse pointer where you want to paste and click the middle button.

2.3 Evaluation and Substitution

It usually is not a good idea to assign values to variables (like: `x:=3;`)—they are more useful as variables. If you want Maple to evaluate an expression you can give the variables temporary values.

```
f:=A*(x+3)^3;
subs(A=3,f);
subs({x=5,A=2},f);
x; A; f;
```

Note here that you use a plain '=' inside the substitution command. Two or more substitutions are done by enclosing them in curly braces. The values of f, A and x *don't* change.

```
subs(x=z^2,f);
f:=subs(x=z^3,f);
f;
```

You can substitute whole expressions for variables, not just values. To force the value of f to change, just reassign it—but be careful. It usually is better to give the expression a new name in case you need the original definition of f later.

```
f:=x^2+3*x+4;
f(1);
g:=x -> x^2+3*x+4;
g(1);
h:=unapply(f,x);
h(1);
h(x);
h(x^3);
```

Maple makes a crucial distinction between an expression like x^2+3x+4 and the function which assigns to each value of x the value x^2+3x+4. In this example, f is an expression, g and h are functions. The same distinction exists in mathematics, but usually it is possible to ignore it, and indeed in many instances Maple will accept either a function or an expression. To employ the normal mathematical notation $f(1)$, however, f must be a function. This example shows two ways to define a function—directly, or using an existing expression. The last two examples show that you can evaluate a function on expressions, not just numbers.

```
map(g,[-3,-2,-1,0,1,2,3]);
map(g,{-3,-2,-1,0,1,2,3});
```

Using the 'map' operation and a function (not an expression) you can evaluate a function on a list of values. The list can be enclosed in square brackets or in set braces—but notice the difference! The set braces produce a set as a result, meaning that repeated values are shown only once, and the order does not necessarily correspond to the order of the input values. In general, the square bracket form is more useful.

2.4 Solving Equations

Maple will solve a limited class of equations exactly and a broader class approximately.

```
solve(x^2-x-10=10,x);
solutions:=solve(g(x),x);
solve(x^5+4*x^3-3*x^2+10,x);
solve(tan(x)=x+1,x);
allvalues(%);
solve(tan(x)=x,x);
allvalues(%);
solve(sin(x)=cos(x)-1,x);
```

The *solve* command will give exact answers if possible, including complex answers. If Maple can't solve an equation it may print nothing, or it may display the equation in a standard form, preceded by 'RootOf'. You can ask Maple to attempt to evaluate roots displayed as a RootOf using `allvalues`. Maple will attempt to find all solutions for an equation or a RootOf, but often will not succeed.

Do not rely on the solve command alone when looking for solutions to equations—at the very least, you should graph your functions so that you know where roots appear to be. Note that if you give an expression instead of an equation Maple assumes that you mean '$= 0$'.

```
map(g,[solutions]);
simplify(%);
```

In some cases, Maple will not automatically simplify as much as it can, so you need to nudge it a bit. In this case, of course, we knew that the values should be zero. If the result had not simplified to zero, careful scrutiny of the problem would be called for.

```
f:=x^3+3*x^2+x+5:
solve(f,x);
fsolve(f,x);
fsolve(f,x,complex);
```

You can end a line with a colon instead of a semicolon—it tells Maple not to display the result. This can be particularly handy when the result is very long, or simply when you either know what it is or do not need to know (as when you will be using the result to compute a further result). The *fsolve* operation tells Maple to approximate the roots. In this example, Maple can compute the roots exactly, but it may be more informative to see approximations. Note Maple only computes real roots unless the 'complex' option is present. If the function is not a polynomial, fsolve will find at most one solution, but for polynomials it will find all solutions.

```
g:=x -> exp(x)+x^2-10*x;
fsolve(g(x),x);
fsolve(g(x),x,-2..2);
g(%);
fsolve(g(x),x,2..4);
g(%);
```

You can specify a range in fsolve—you would normally do this when you know there are multiple roots and you want to approximate a particular one. In this case, a graph of the function clearly indicates a root near 0 and a root near 3. Because g is a function, and fsolve expects an expression, it is necessary to use $g(x)$ in fsolve. (I made g a function so that it would be easy to check the answers.)

```
solve({3*x+5*y=6,4*x-6*y=-8},
      {x,y});
h1:=3*x-6*y-5;
h2:=-4*x*y+7;
solve({h1,h2},{x,y});
evalf(%);
fsolve({h1,h2},{x,y});
fsolve({h1,h2},{x,y},{x=0..6,y=0..2});
```

Maple can solve equations simultaneously, with either solve or fsolve. As with single equations, fsolve may not find all solutions, but you can specify ranges for x and y to force fsolve to find particular solutions.

3 Plotting

Maple can do two and three dimensional graphing. There are many ways to adjust the plots that Maple produces; we'll cover just the basics here.

```
plot(f,x=-10..10);
plot(g(x),x=-5..5);
plot(g,-10..10);
plot(g,-10..10,y=-10..10);
```

Maple will accept either an expression or a function to plot, but notice that the form of the range is a bit different in the two cases. For an expression, use '`x=`' before the range, but for a function leave it out. You can also specify the range for y explicitly. This is often useful if the function gets very large on the interval.

You can change the appearance of the plot in a variety of ways. Click in the plot and you should see a border appear around it; then you can explore the items on the plot menu. If you prefer, you can have plots come up in separate windows; use the **Options** item on the **Tools** menu in the main Maple window, then open the Display tab and change the plot display.

```
plot([1+2*sin(t),t,t=-Pi..Pi],
     coords=polar);
```

You can do plots in polar coordinates. Inside the square brackets, the first expression is the radius, the second is the angle, and the third is the range of the variable. In this case, the angle and the variable are the same, but in general the angle could be a more complicated function of the variable.

```
plot({x^2,x^3},x=-5..5);
plot({
  [1+2*sin(t),t,t=-Pi..Pi],
  [cos(4*t),t,t=-Pi..Pi]},
  coords=polar);
```

You can put more than one curve in the same plot by enclosing a list of expressions in curly braces.

```
r:=sqrt(x^2+y^2);
f:=sin(r)/r;
plot3d(f,x=-10..10,
         y=-10..10);
```

Try changing the appearance of a 3D plot by using the **Style**, **Color**, **Axes** and **Projection** menus. If you position the cursor in the drawing area and press *and hold* the left mouse button, you can move the mouse to change the orientation of the plot—try it.

```
with(plots):
polarplot(1+sin(t));
polarplot(sin(t/3));
polarplot(sin(t/3),
     t=0..3*Pi);
```

When you start Maple, only the most basic plotting commands are available. By typing "`with(plots)`" you tell Maple to load a number of special purpose plotting functions. One of the most useful is the polarplot routine shown here. By default, Maple uses the range $[-\pi, \pi]$; the last command shows how to specify a different range.

4 Calculus

4.1 Limits

```
f:=x^2+4*x+30:
limit(f,x=3);
subs(x=3,f);
g:=sin(x)/x;
limit(g,x=0);
limit(g,x=infinity);
g:=x/abs(x);
limit(g,x=0);
limit(g,x=0,left);
limit(g,x=0,right);
```

The *limit* command `limit(f,x=a)` computes the limit $\lim_{x\to a} f(x)$. Maple can handle many non-trivial limit problems and limits at infinity.

4.2 Differentiation

```
f:=x->x^3+4*x^2+3*x+4:
fp:=diff(f(x),x);
critpts:=solve(%,x);
map(f,[critpts]);
evalf(%);
evalf(critpts);
plot({f(x),fp},x=-5..5,
                y=-5..10);
```

The *diff* command does differentiation. Here I have found the derivative with respect to x, then the x coordinates of the critical points of f, then the value of f at these points, then floating point representations for the x and y coordinates. Because Maple can do calculations exactly, you should do approximations at the end; it would not be a good idea to approximate *critpts* before substitution into f (though in this simple case the results are almost identical). Note that if you click in the plot, the coordinates of the point under the cursor appear at the upper left of the Maple window. Youe can use this to approximate critical points and double check the computed values.

4.3 Implicit Differentiation

```
eqn:=x^2+y^2=1;
xp:=implicitdiff(eqn,x,y);
yp:=implicitdiff(eqn,y,x);
subs({x=1/2,y=sqrt(3/4)},yp);
```

Notice that the order of the second two arguments to the `implicitdiff` function is important—`xp` is dx/dy and `yp` is dy/dx. The last line computes y' at a specific point, $(1/2, \sqrt{3}/2)$.

4.4 Integration

```
f:=(2*z-3)*sqrt(z-3);
g:=int(f,z);
diff(%,z);
factor(%);
```

The *int* command shown here will compute an antiderivative. The computer is not infallible, and you may not notice a typing mistake, so it's not a bad idea to use Maple to check your work. Here, I've taken the derivative to check the antiderivative.

```
int(f,z=5..10);
simplify(%);
evalf(%);
subs(z=10,g)-subs(z=5,g);
```

To do definite integration give the limits after the variable. Of course, you can also do it yourself using the antiderivative. Notice that the answer can look quite different before simplification.

```
f:=1/sqrt(x):
int(f,x=0..1);
f:=1/x^2:
int(f,x=0..1);
int(f,x=1..infinity);
f:=exp(-x^2):
int(f,x=-infinity..infinity);
int(sin(x),x=0..infinity);
```

Maple can do many improper integrals exactly. If the integral does not converge, Maple may answer with '∞' if appropriate, or it may give an error message.

```
f:=tan(x^2):
int(f,x);
int(f,x=0..1);
evalf(%);
Int(f,x=0..1);
evalf(%);
```

Maple can't find antiderivatives of some functions. In such cases you can use *evalf* to approximate the value of a definite integral. If you don't want Maple to even try to find an antiderivative, use the capital I form of the integration operator.

5 Adding text to a Maple session

You can add plain text to a *Maple* session. This is especially useful if you are going to give a printed or electronic version of a session to someone else, since it lets you explain what is going on. It is also a good idea to add text to remind yourself at a later date what you were doing and why.

To add text above a command, position the cursor anywhere in the command. Then open the Insert menu, and then the Paragraph submenu, and choose Before Cursor. When you type you will get ordinary text that will not be interpreted as part of the command.

6 Printing

BEFORE YOU PRINT: You should clean up your session before you print, so that you do not waste paper and so that the printed copy is attractive and easy to read. If you want to print only part of the session, first save the session (see the section *Saving your work*) and then cut out the portions of the session that you do not want to print. You might want to save the trimmed version using a different name.

To print your worksheet, choose **Print** from the **File** menu, or click on the print button (the fourth little button from the left in the menu bar).

If you would like to print just a plot, you need to display it in a window of its own. You can do this by changing the plot display option, as described in the *Plotting* section. Then position the cursor in the plot command and press return. The plot should open in a new window, and you can use the Print menu in that window to print the plot.

7 Working with Data

Let's begin with an example from the `help` file for `LinearFit`. In this case, we have some model function that we want to fit to a given set of data points.

In this example, we will use the data from the table:

x	1	2	3	4	5	6
y	2	3	4	3.5	5.8	7

And we will find the values of a, b, and c so that the error between these points and the parabola:

$$y = a + bt + ct^2$$

is minimized. We won't go into the numerical routines, but here are the Maple commands:

```
with(Statistics):
with(plots):
```

We will work with commands that require these two packages.

Load the data, then find the function. In this case, the variable A stores the function (as a function of t).

```
X := Vector([1, 2, 3, 4, 5, 6], datatype = float);
Y := Vector([2, 3, 4, 3.5, 5.8, 7], datatype = float);
A := LinearFit([1, t, t^2], X, Y, t);
```

```
B:=pointplot(X,Y);
C:=plot(A,t=0..7);
display({B,C});
```

Plot the data points, plot the curve, then overlay the two plots on one graph.

8 Saving your work

You can save your Maple worksheet so that you can come back to it at a later time. Choose the **Save** or **Save As...** item from the **File** menu in Maple; a new window will pop up. (Use **Save As** if you want to save your worksheet under a new name.) You may also save a worksheet using the third button from the left in the menu bar.

When you reload a worksheet (with the **Open** item on the **File** menu or with the second button from the left) Maple will not automatically recreate all the values created in the worksheet. You may selectively re-execute commands in the worksheet, or you may use the **Execute** item on the **Edit** menu to execute the entire worksheet.

HINT: The size of the file is a lot smaller if you save only your Maple commands and leave out plots, animations, and Maple output. To remove the output before saving your file, go to: **Edit**, then **Remove Output**, then **From Worksheet**. Once this is saved, if you want all the commands to be re-executed after opening the file, select the triple exclamation point button: `!!!`.

9 Getting Help

There are (somewhat out of date) manuals for Maple in the lab. You can also get help 'on line' from the Maple program. The help system is very good and always available as you work in Maple.

`?int`	If you know the command you need help on, just type its name after a question mark.

Of particular use are the examples at the bottom of most help pages. Often you can just copy an example from the help page into your worksheet and modify it to suit your needs. Also useful are the links at the bottom that direct you to related help pages.

You can browse through the entire help system by choosing **Maple Help** from the **Help** menu.

10 Packages

Maple is a large software package. For the sake of efficiency, only a core set of commands is loaded on start up. Many useful commands are in "packages" that must be loaded manually by the user before use. To see a summary of the available packages choose the menu item `Help`→`Manuals, Resources, and more`→`List of Packages`.

To load a package:

`with(plots);` `?animate;`	Notice that a list of commands included in the `plots` package is listed. Information about particular commands is available by typing a `?` followed by the command name.
`with(plottools);` `?polygon;`	The `plottools` is another package that is useful for constructing advanced figures for inclusion in write-ups. You can load as many packages as are necessary to accomplish the task at hand.

Though you can load packages from any position in the worksheet, for complicated worksheets with lots of loaded packages, it is useful to group all of the commands for loading the packages near the top of the worksheet.