Part I: playing with arithmetic on *Maple*

Start the *Maple* program. Under **File/New** please select Worksheet Mode.

There are two forms in which you can enter commands into *Maple*, **1D Input** and **2D Input**; some users like one, some the other. You can switch back and forth with a button the the top of the screen. In **1D Input** what you type appears directly on the screen, and all commands must end with a semicolon or colon. In **2D Input** what you type is rearranged to look like conventional mathematics. Here we will assume you are using **2D Input**, but not much would be changed if you used the other form.

The worksheet will open with a *Maple* command line and a blinking cursor that looks like this:

> /

Your *cursor position* is the blinking / on the input line, indicated by the > sign.
Now type the following.

3+2 **RET**

[*Note:* '**RET**" means: hit the "enter" key (new line) .]

"Things" should happen: you should get 5 and a new input line. By the way, if you looked carefully, you will see that the cursor changed from a slightly tilted / to | when the input line is not empty. You can move your cursor up and down with the arrow keys. Please do this. The phrase above the *Maple* worksheet changes from **2D Input** on the *Maple* screen to **2D Output** and then to **2D Input** when your cursor gets back to the original line.
Now move your cursor back to your new input line. Type

17∗3 **RET**

and see the result. The input line will display 17·3, using a centered dot for multiplication. But you must type ∗ to tell *Maple* about the multiplication. —At the next input line type

%+5 **RET**

and explain the result. In *Maple*, % is called the *ditto operator*.—Now type the following to learn what ^ means.

2^3 **RET**

Notice that *Maple*'s graphical interface interprets ^ as a typographical request for superscripts. But I wanted to calculate the 300^{th} power of 2. Move your cursor back to the input line with 2^3 and position it as shown: $> 2^{\boxed{3}}$ and type *00* and then **RET**. What happened?
You should learn to navigate superscripts, so please compute $2^{300} \cdot 300^2$ (After you type the "300", the right-arrow key will return you to the main line.) The first five digits of the answer should be 18333.

Please type the following (carefully! – I'm asking for a **colon** at the end of the expression!).

5+6: **RET**

You should immediately get another input line. Type (for example)

%+7

and deduce what *Maple* does when an input line ends with a : (that is, a colon). Note that computations might and do occur (such as 2^(2^(2^(2^2)))) which have results that are huge and silly to print out if you don't need them. Try that if you'd like, with no ":" and see how *Maple* handles the result.

(OVER)

Onward—please type

20 **RET**

Put a space between the 2 and the 0, then **RET**. What happens? — Next experiment:

2∗3+7 **RET**

and observe that *Maple* follows the usual rules of precedence. Can you put parentheses in so that *Maple* will compute two times the sum of three plus seven instead? Remember **RET** after you make the alterations. You should have gotten 20, of course. If you did **not** make an error inserting the parentheses, go back and take one out (create an intentional **error**!) and then hit **RET**. What happens? You haven't broken anything. Let's keep exploring ....

Please get a new input line and try the following commands in succession to learn how to do more arithmetic and to explore more features of *Maple*.

2/3; **RET**

The result on the screen is a more traditionally typeset fraction. Please go back and change this to $\frac{4}{3}$ and then change it to $\frac{4}{7}$. The arrow keys will help you. Now suppose you have entered the command *4/7* **RET**. *Maple* computes "exactly" and can do simple arithmetic:

%∗700 **RET**

Now try

sqrt(2) **RET**

followed by

%^2 **RET**

so *Maple* knows the "meaning" of fractions and square roots – or at least how to manipulate them. And now try (remember, if you mess up with a parenthesis or something else, just go back and do it again – nothing is broken!):

(sqrt(2)-1)^5 **RET**

This result is puzzling. Sometimes *Maple* is lazy. Let's urge it to work by writing

expand(%) **RET**

That's better. But what if we want or need decimal approximations? Try

evalf(sqrt(2)) **RET**

Parentheses must match – always a source of anxiety as more complex expressions and commands are typed. We can coax *evalf* to get more digits of $\sqrt{2}$. To see how, type

help(evalf); **RET**

Another screen should pop up. When I use *Maple* I need a lot of *help*! Read the evalf screen (I usually skip down to the examples on any help screen first!) until you figure out how to get the first 100 digits of $\sqrt{2}$ (see the entry link on the left entitled evalf,details). What is the one-hundredth digit after the decimal point of $\sqrt{2}$? (The answer is 7: be careful with the specification of the number of digits.) Can you tell me the three-hundredth digit after the decimal point of $17^{1/3}$? (I think the answer is 5.) If we type

1400/24; **RET**

we learn that *Maple* knows how to factor integers. Please get *Maple* to factor your social security number. How would you find a factoring command in *Maple*? If the first thing you try with the *help* command doesn't work, look at the references below the **See Also** line at the end or look to the left and check them out. My social security number has three distinct prime factors. How many does yours have?

We can go on to try some algebra. But you can stop your *Maple* session at any time in several ways. One way: select **Quit** from the menu (under File or under Maple, depending on the version used). You do not need to save this *Maple* session!

Part II: playing with algebra on *Maple*

The most attractive feature of *Maple* may be its ability to do intricate symbolic computations. Just try this:

(x+2y)^5 **RET**

Darn it: lazy *Maple*! Tell *Maple* to work harder:

expand(%) **RET**

and see what happens. Now type

x=2 **RET**

and see what happens. Then type

x **RET**

Now try

x:=2 **RET**

followed by

x **RET**

and appreciate the difference. The character string *:=* assigns the **value** on the right to the **name** on the left. Now type

x^3 **RET**

Did you expect that? Try to evaluate $(x+1)^8$ when x has value 2. After doing that, try:

restart **RET**

followed by

(x+1)^8 **RET**

again and "expand" it. Is the result unexpected? What do you think *restart* does, and (more importantly!) how could you check that restart has that function? Hint: just try *help(almost any word!)* when you're curious or confused.

Please assign x the value 17 and then type the following character string **exactly** as written here **but** don't hit **RET** yet!

x;2x;x2;x2;x^2;2x;2^x

The semicolon **;** is used to separate several *Maple* statements on the same input line. Think about what's here and try to predict what will happen. Now hit **RET** and fix up any problems and hit **RET** again. Were you correct?

Warning! In older *Maple* implementations and even in *Maple*'s current command line version, the character string *2x* will be rejected (with the message **syntax error, missing operator or `;`:**). The graphical interface to *Maple* `10` will accept *2x* as an implied `2x`. People are sometimes reluctant to use long variable names, but I think this can be a very useful *Maple* feature. Long names can help you remember what entries represent during complicated computations. For example, try

sumsqrts:=sqrt(x)+sqrt(y)+sqrt(x) **RET**

followed by

expand(sumsqrts^3) **RET**

Certainly more letters take more time to type (and increase the chance for error), but remember this freedom exists: you can call something by a character string close to its real name or with some important attribute recognized. This can reduce confusion.

OVER

Here are some other algebraic manipulations:

factor(y^4–16) **RET**

We can tell *Maple* to use imaginary numbers. See the *help* information about *factor*. Specifically, try the command *factor(y^4–16,I)* **RET**. *Maple* is born knowing some constants. *I* is a number whose square is -1. You may be able to guess what *Pi* is. And the constant *infinity*. In older *Maple* systems, E was the number whose decimal approximation begins 2.71828..., but the latest releases of *Maple* don't "know" this. If you do need E, you can define it with the command *E:=exp(1)* **RET**.

Work with a typical expression occurring at the beginning of calculus: $\dfrac{W+\triangle W)^5 - W^5}{\triangle W}$.

I used the variables W and *deltaW* in my *Maple* analysis and (after *expand*ing!) got the usual calculus mess involving both W and $\triangle W$. Notice now (if you haven't already) that *Maple* is "case-sensitive". Therefore x and X need not be the same. Be careful!

You can now have fun doing algebraic things which no sane human being would ever think of doing "by hand". For example, what is the coefficient of r^7 in $(r^2+3r+4)^{10}$? Please remember all the necessary parentheses and ^'s. (I think the answer is 175173120.)

Maple can also substitute in algebraic expressions. Try

subs(a=t,5a^3+3ta+2sqrt(a)) **RET**

This command changes a to t. It isn't equality. Try the following command with no between t and a:

subs(a=t,5a^3+3ta+2sqrt(a)) **RET**

Maple will compute exactly what you ask! In *3ta* the initial *3* will multiply the product of the variables t and a. *3ta* is interpreted as a request to multiply the variable *ta* by 3.

Braces or curly brackets, { and }, are used to create a list of variables for the *subs* command. Please try to predict what the result of the following will be **before** hitting **RET**:

subs({a=t,b=t^2,c=t^3},ab^2c^3) **RET**

Maple can solve some equations. Try

solve(x^3=7x+1,x) **RET**

and

solve({ab+3=2,a+b=0}) **RET**

followed by

solve({ab+3=2,a+b=1}) **RET**

and I don't know why there's such a difference in the answer (hey, *Maple* will tell you the roots of quadratics using the quadratic formula – just ask it). You could always try *help(solve)* which brings up a huge amount of information, and probably the reason for the difference is explained there somewhere!

Maple will also find approximate numerical solutions. You could explore the difference in the answers to the command

solve(x^7–x^2+1); **RET**

(*Maple* assumes you mean to ask for a root of the equation obtained by setting the expression $x^7 - x + 1$ equal to 0) and the command

fsolve(x^7–x^2+1); **RET**

which gives an approximate numerical solution.

Now, let's go on to calculus.

Part III: playing with calculus on *Maple*

We will no longer indicate the return **RET** *at the end of each command line in Maple*

The basic calculus commands do differentiation and integration. Let's try them.

diff(3x^7 – 22.1x^2,x) **RET**

int(xsqrt(x+2),x) **RET**

This integration is not easy: *Maple* either substituted or it integrated by parts (it can do both). Typing the explicitly forces *Maple* to recognize variables. I've done some "experiments" and the results suggest that writing 's rather than having *Maple* guess leads to fewer misunderstandings. *Maple* likely knows all the functions you do, and many others. The function $\sin\left(x^2\sqrt{x+1}\right)$ and the calculus rules are known. Let's assign this expression a name and then play.

Q:=sin(x^2 sqrt(x+1)) **RET**

diff(Q,x) **RET**

This should get the first derivative. How many ways can you think of to get the second derivative? First, immediately after the command and response above, type

diff(% ,x) **RET**

This will do it. For an independent computation, try the command line

diff(Q,x,x) **RET**

How about the tenth derivative? First type

x$10 **RET**

to see what *$* means. Now you can find the tenth derivative of the function $\sin\left(x^2\sqrt{x+1}\right)$. Realize (even if the tenth derivative of this function is needed) why people end commands with *:* (which turns off the output) rather than displaying the output. You may want the results of a computation, but you may not have the need or desire to actually look at it!

What is the coefficient of x^3 in the seventh derivative of $\left(x^2+\frac{1}{x^2}\right)^5$? First compute the indicated derivative. You'll get a mess. Then have *Maple* massage the result algebraically so you can read off the answer. I'm an amateur and first tried *expand(%)* and I also tried *simplify(%)* and the results were different. Another way is to first *expand* $\left(x^2+\frac{1}{x^2}\right)^5$ and then differentiate the result seven times. The answer should be the same!

Let's look at integration a bit more closely. Define V to be $e^{\sin x}$:

V:=exp(sin(x)) **RET**

Now let's integrate it. First (try this carefully!) type

int(V,w) **RET**

and explain the result to yourself. Remember, a program will do what you tell it to do! Now try

int(V,x) **RET**

and you may wait a bit and then have something else to explain. *Maple* has the usual integration algorithms and many, many other antidifferentiation tricks. An answer like this is a fairly good hint that it "can't be done": that is, the antiderivative can't be expressed in terms of familiar functions with familiar ways of combining them, including sum, product, composition, ... even using the rather large collection of functions known to *Maple*. Such results can be proved!

OVER

We can also compute definite integrals. For example,

int(x^3,x=1/7..b) **RET**

computes $\int_{\frac{1}{7}}^{b} x^3 dx$ (if you want it!). *Maple* indicates a range (for integration and for other purposes) by the notation *variable=lower_limit..upper_limit*.

Remember V? Be sure *Maple* does (check by typing V) and compute $\int_0^1 e^{\sin x}\, dx$ by typing

int(V,x=0..1) **RET**

and consider the result. Disappointment is decreased if we follow that answer with

evalf(%); **RET**

You can evaluate V itself with a command like

subs({x=3},V) **RET**

followed by

evalf(%) **RET**

If you're not scared, you could have done this all together by typing

evalf(subs({x=3},V)) **RET**

but sometimes I get confused by the matching required (in count and type) of all the parentheses.

We could similarly evaluate a derivative of V by differentiating with *diff*, *subs*tituting, and then *evalf*uating. Or we could define our own functions. Initially the syntax may seem burdensome, but here is a simple example.

N:=x->arctan(x^3) **RET**

So: call N the function which takes the input x and assigns to it $\arctan(x^3)$. Try this

N(2);N(5z) **RET**

which shows that *Maple* believes N is a function. Now type

diff(N(x),x); **RET**

That's an **expression** and not a function. There's no way to "plug in" $K+3$ easily in it (yes, we could use the *subs* command, but that's cumbersome). So *Maple* has another way to differentiate functions (such as N) rather than expressions (such as V). Try

D(N); **RET**

and view the result. Indeed: call it by a new name, say, M:

M:=D(N); **RET**

Now evaluate $M(3)$ and $M(K+3)$.

Please check the difference between

int(N(x),x); **RET**

and

int(N,x); **RET**

One of these has an answer that's fun to me (because I didn't have to compute it and can appreciate the work involved!) and the other seems silly: the function N hasn't been told what to evaluate, so it can't be antidifferentiated.

The difference between "expressions" like V and "functions" like N can be subtle. Expressions seem more static, while functions have a more dynamic aspect – substituting is built into their structure.

Maple also knows about such calculus topics as limits, sums, and series. If you need to work with any of these, look at *help(limit)* and *help(sum)* and *help(series)*. I almost always look at the examples first. Don't be afraid to experiment, and look at the *help* pages if the command you are investigating doesn't do exactly what you want.

It's time to *graph.*

Part IV: playing with graphs on *Maple*

We no longer indicate the return **RET** *at the end of each command line in Maple*

The graphing capabilities of *Maple* explored here will probably seem rather familiar to you after your experience with graphing calculators and possibly other programs or devices. I won't discuss polar plots or parametric plots – but even those are generally available on hand-held calculators now. *Maple* draws lovely pictures with three-dimensional plots. This capacity will be used extensively later in this course.

Let's begin with the simple command

plot(x^3–6x+1) **RET**

After the program responds, move back and alter your command to read

plot(x^3–6x+1,x) **RET**

Now a graph should appear.

There are some things to notice. For this kind of plot, the "default" interval for x (what *Maple* assumes unless advised otherwise) is $[-10, 10]$. Maybe this is o.k. for you (suggestion: try graphing $\frac{1}{\sqrt{x^2-100}}$ and see what happens) but you may want more control. The *plot* command has many options. Read about them during the first free week you have by typing *help(plot)* and following all the references!

Let's look at the graph we have. By the way, when I am graphing, I tend to foul things up a great deal and I frequently end up with 5 or 10 graphs sitting around at the same time. Try to be neat. You can get rid of those graphs you don't want by (mouse) clicking on the plot and then invoking **Cut**. Of course, if they hang around until you *quit Maple* entirely, the graphs will then disappear also. But you can try to decrease your own confusion.

Again, look at the graph. Note that the vertical and horizontal axes are very differently scaled. But *Maple* is trained to "autoscale" so it will distort the picture to fill the rectangular screen. I wrote that word in a large and bold font because it's something that has repeatedly caused me confusion. Now go to the graph, right click, and click on **Scaling Constrained** and that's the way it really looks, darn it: a very, very thin graph.

Let's draw another graph.

plot(sin(100/x),x=.1..1) **RET**

This graph is attempting to illustrate some well-known misbehavior since $\lim_{x\to 0^+} \sin(\frac{100}{x})$ doesn't exist. The graph should bounce around a lot when x is close to 0. It certainly seems to, but notice (by pure thought: the Intermediate Value Theorem) that between each root or x-intercept of the function, there must alternately be a place where the function is $+1$ and where the function is -1. The graph doesn't show that! (Look carefully, please.) *Maple* doesn't think. It plots points and connects the dots to produce the graph. The default performance is fairly simple-minded (more sophisticated alternatives can be specified) and the plotted points can be seen by clicking on **Style** and then on **Point** – you'll see just the computed points on the curve with no connections drawn. If you increase the number of points to be plotted, you will probably get a better picture, but computation time for the graph's creation increases.

Here's another type of bad picture. Please look closely at the graph resulting from

plot(1/(10^3x),x=–1..1) **RET**

and you can try to determine the range of the function based only on the appearance of the graph. The range, deduced from the displayed graph, seems to be approximately the interval

$[-1, 2.5]$. Of course this is incorrect. I know that $\frac{1}{10^{3x}}$ on $[-1, 1]$ is not defined at 0 and is also unbounded both positively and negatively. So *Maple* may have difficulty graphing functions with discontinuities. Some of the options for *plot* may be useful. The *help* screen for *plot* is long and has many entries.

By the way, any of the programs (`Mathematica`, `Maple`, graphing calculators, etc.) can be "spoofed" and made to draw bad graphs. Devices with finite memories can hardly imitate all aspects of the real numbers accurately.

Try the command (but you may want to guess the result before completing the input):

plot({x^2,x^3},x=4..5) **RET**

Therefore you can plot collections of functions, certainly useful sometimes.

Define your own function and plot it: try to define $P(w) = \frac{e^w}{1+w^2}$ and then remember that *Maple* finds the derivative function of P with the character string *D(P)*. Try this:

plot({P(t),D(P)(t)},t=–2..5) **RET**

As long as the variable (here t) is used consistently, *Maple* will have no difficulty. Compare the two graphs. Where *D(P)* is 0 then P must have a horizontal tangent. What happens to the graph of P when the graph of *D(P)* has a maximum?

We can graph curves implicitly defined by equations. For example, try

implicitplot(x^3–5xy^2=7,x=–5..5,y=–5..5)

and notice the result: nothing happens! Well, *Maple* is such a huge program that most of it is stored "asleep", and these parts must be specifically recalled to active memory. In this case, we need to load the routines in the *plots* package. This is done with the command

with(plots)

Now after you have loaded the *plots* package, please try again

implicitplot(x^3–5xy^2=7,x=–5..5,y=–5..5)

the result should be a nice picture. Implicit curve plotting is very difficult and due to problems with discontinuities frequently may give pictures which are untrue. Look at

implicitplot(x^3-5xy^2=7,x=–50..50,y=–50..50)

which asks for the same algebraic curve in a 100×100 window. The result seems to have some corners and wiggles which I doubt are correct. Here also some of the options which can be seen with *help(implicitplot)* may be useful.

Let's conclude by attempting to locate a root.

plot(xln(x)–sin(x),x=0..3) **RET**

This function seems to have a root. Right click on this graph and then investigate the **Manipulator**. The capabilities of **Point probe**, **Scale**, and **Pan** work very much like "zooming" and "tracing" with a graphing calculator. I can easily find an approximate location of the root (about 1.77, I think).

We can check the picture with a numerical computation:

fsolve(xln(x)–sin(x),x) **RET**

reports 1.75267781, fairly close to where the root can be located graphically. You can, of course, read about *fsolve* with the *help* command.

Continue to explore, please. And thank you for working through this.