Jumpstarting Technology in the Calculus Curriculum

Dr. Patrice Geary Tiffany
Dr. Rosemary Carroll Farley
Manhattan College
Bronx, NY 10471
email:patrice.tiffany@manhattan.edu
email:rosemary.farley@manhattan.edu

Introduction:

We have spent the last decade incorporating Maple into our mathematics courses at Manhattan College. This has been an extremely time consuming process filled with both successes and failures. We have learned that each student comes to a lab session with his or her own style. It is important that this style be respected. The students must be encouraged to try to solve problems in their individual ways. Mathematics must be the driving force in any laboratory setting. The teacher must be ready to answer questions about individual attempts. This only works if the teacher is able to recognize syntax mistakes quickly and help students resolve their problems expeditiously. The fact that each year different students make the same mistakes led us to write this paper.

This paper is intended for those faculty members who want to begin to incorporate Maple into their calculus classes. It is intended for the novice Maple user. We will stress what we consider to be the most important Maple commands. You will learn enough code to be able to start using Maple immediately. This paper contains examples of the most important Maple commands. It also contains examples of the most common errors made by students. Comments referring to errors are italicized.

Common Maple commands

1. **Load in the appropriate libraries.** In calculus classes the student library is usually all you need. For more advanced graphing you will need the plots package.

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

There must be either a colon or a semicolon at the end of each Maple command. If you type a colon, Maple performs the command but does not provide any output. Output is given when the semicolon is at the end of the command. *The most common mistake that students make at the beginning of their Maple experience is that they forget to include the semicolon or the colon at the end of each line.* At an ICTCM meeting many years ago, a presenter said that when the students don't put the semicolon at the end of the line, he said, "You forgot to say 'please'." It works!

2. **Define a function.** *Students often leave out one or more of the symbols between the name of the function and the function itself.* Remind the students that there are **five** symbols between the name of the function and the function itself.

```
> f:=x->x^2-9;
> g:=x->x^3+7*x^2-8*x+9;
```

Note that the symbol * is needed for multiplication. *One of the most common mistakes made by students is to type 7 x rather than 7 * x. Another common mistake is to forget parentheses. The students make the mistake of typing cosx rather than cos(x).* Now f (x) can be used as usual. For example:

```
> f(2);
> f(x+h);
```

3. **Plot a function.** Maple will automatically plot between -10 and 10.

```
> plot(f(x),x);
```

If you want to plot over a different domain, list the values between which you want to plot.

```
> plot(f(x),x=0..3);
```

If in addition you want to limit the y values, list both the x values and y values.

```
> plot(f(x),x=0..3,y=-10..10);
```

You can plot more than one function. Note that you need the braces { } in the following code.

```
> plot({f(x),g(x)},x);
```

You do not have to declare the function before you plot it.

```
> plot(x^2,x=0..3);
```

4. **Create implicit plots.** Suppose that we want to plot something that is not a function. For example, plot $x^2+y^2=25$. Note that to use this command, you need the plots package.

```
> implicitplot(x^2+y^2=25,x=-5..5,y=-5..5);
```

You can combine these types of graphs. This is more advanced. You can create the graphs and give them names. Then use the display command to put them together. Note the use of the colons and the semicolon. You don't want output on the first two lines, but you do want it on the last command.

```
> p1:=plot(x^2,x=-2..2):
> p2:=implicitplot(x^2+y^2=4,x=-2..2,y=-2..2):
> display({p1,p2});
```

5. **Solve equations.**

```
> solve(f(x)=g(x));
```

You can find a decimal approximation. The symbol % is used to represent the **last entered** line. *Sometimes the students use the arrow keys to move around the worksheet and then get an incorrect answer when they include a line containing a % sign.*

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

6. **Find a limit.** For example, find $\lim_{x \to 2} \frac{x^2 - 4}{x - 2}$.

```
> limit((x^2-4)/(x-2),x=2);
```

You can also use the declared function f(x).

```
> limit(f(x),x=1);
> limit((f(x+h)-f(x))/h,h=0);
```

7. **Differentiate a function.**

```
> D(g)(x);
```

*The **mistake** the students make here is that type the following.*

```
> D(g(x));
```

8. **Find a second derivative.** We will differentiate the derivative.

```
> D(D(g))(x);
```

*The **mistake** the students make is this.*

```
> D(D(g(x)));
```

If g is the name of the function, then g(x) is the actual function. D(g) is the name of the derivative, and D(g)(x) is the actual derivative. D(D(g)) is the name of the second derivative, D(D(g))(x) is the actual second derivative. It helps to remind the students that (x) will be the last part of the above lines.

9. **Integrate a function.** First we will find the indefinite integral.

```
> int(g(x),x);
```

Now find the definite integral.

```
> int(g(x),x=0..2);
```

10. Define a constant. Use the assignment statement.

```
> k:=7;
```

You can also use the assignment statement as follows.

```
> v:=int(g(x),x=0..2);
```

11. Define functions of two variables.

```
> f:=(x,y)-> 2*x*(x-y);
```

This can be used in the same way as functions of one variable could be used.

```
> f(1,2);
> limit(((f(x+h,y)-f(x,y))/h, h=0));
```

The most common ***mistake*** *the students make is that they type:*

```
> f:=x->2*x*(x-y);
```

Note: In the above command, the function f has been defined as a function of x. Maple will now treat y as a constant. The problem is that you will not get an error message if you really want to treat f as a function of x and y. It is only when subsequent lines of code do not produce the expected results that an error like this is found. Note that if you entered the line containing the mistake, you either have to correct it or scroll up and reenter the correct line. *The students make the same mistakes they did when they declared functions of one variable. They forget one or more of the* ***five*** *symbols they need between the name of the function and the function itself. They also mix up the order of these symbols.*

12. Create a three dimensional plot of a function.

Note that to use this command, you need the plots package. Suppose that $f(x, y) = 2\,x\,(x - y)$ for $0 < x < 1$, $-x < y < x$; 0 elsewhere. Create a three dimensional plot of this function.

```
> plot3d(f(x,y),y=-x..x,x=0..1);
```

If you click and drag the mouse anywhere on the graph, you can look at the graph from different perspectives. There are several features that you can add to the graph. Click on the graph and new graph icons appear on the toolbar. Experiment with them.

13. Create a three dimensional graph of a surface.

Suppose you want to graph the sphere $x^2 + y^2 + z^2 = 25$.

```
> implicitplot3d(x^2+y^2+z^2 = 25,x=-5..5,y=-5..5,z=-5..5);
```

Note that every function can be plotted using the above command. However, using the above command you have to declare not only the x and y values but the z values. The plot3d command allows you to define only the x and y values. You can also combine these types of graphs as we did before.

14. Find a double integral.

Let $f(x, y) = 2\,x\,(x - y)$ for 0 < x < 1, -x < y < x; 0 elsewhere. Double integrate the function over the defined region. Note that we are performing two integrations here.

```
> int(int(f(x,y),y=-x..x),x=0..1);
```

*There are several **mistakes** that students make here. The most common mistake is that students omit one of the int commands and they also omit the close parenthesis after the statement y = - x .. x. They use the same kind of syntax as in the plot3d command. That is they type:*

```
> int(f(x,y),y=-x..x,x=0..1);
```

It helps to remind the students that they are performing two different integrations and so they need two int commands.

*Realize that students will also make the **mathematical mistake** of switching the x and y values.*

```
> int(int(f(x,y),x=0..1),y=-x..x);
```

15. Find a partial derivative.

The first partial derivative of f (x , y) with respect to x is:

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

The first partial derivative of f (x , y) with respect to y is:

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

Note that each of the above is the name of the function. The actual function is:

```
> D[1](f)(x,y);
```

16. Find a second partial derivative. The four second partials are:

```
> D[1,1](f);
> D[1,2](f);
> D[2,1](f);
> D[2,2](f);
```

Remember that f was defined as f : = (x , y) - >. Because of this, the 1 in the above code refers to the first variable x and the 2 refers to the second variable y.

17. Solving simultaneous equations. Consider the following example. Set the two first partials equal to 0 and solve.

```
> solve({D[1](f)(x,y)=0,D[2](f)(x,y)=0});
```

In order to understand the need for the symbols { } you have to understand the different possibilities in the solve command. The following command will set the first partial derivative of f (x , y) with respect to y equal to 0 and solve it for the variable x.

```
> solve(D[2](f)(x,y)=0,x);
```

To distinguish the comma in the above line from the comma needed in simultaneous equations, we need to include the symbols { } around the simultaneous equations.

Conclusion: These are the commands we have used most often in our calculus courses. Master them and have fun using Maple in the calculus classroom. Keep it simple. Make mathematics the driving force in everything you do.