First Introduction to *Mathematica*

If you don't have *Mathematica* yet, go to: http://www.engr.washington.edu/mycoe/computing/software/install_mathematica
and click the link under "**Order:**" Follow the directions to obtain your copy of *Mathematica.* The current version of *Mathematica* is 11 but it appears only 10 appears on the website. I believe after you get your license you should be able to download 11 from Wolfram. Regardless, either *Mathematica* 10 or 11 will be sufficient for all topics we will be covering. If you have trouble getting *Mathematica*, feel free to contact me at smithm16@uw.edu

Creating functions and variables

The basic format of a function in *Mathematica* is NameOfFunction[variable1_,variable2_,...]:= function involving variables. The underscore lets *Mathematica* know that the preceeding letter is a placeholder for whatever input you want to put into that function. The colon tells *Mathematica* not to evaluate the function until you have given it an argument.

To run whatever you have written in a line (defining a function in this case), click somewhere in the line and then push "Shift+Enter"

```
f[x_] := x^2
```

To use the function, the basic format is NameOfFunction[argument]

```
f[2]
```

```
4
```

To define a variable, the format is just Variable= argument. This stores the argument as whatever letter or string of letters you chose for your variable

```
a = 2
```

```
2
```

Now when I type just the variable, it will return whatever is stored in it.

```
a
```

```
2
```

Anywhere you could use that number, you can use the variable instead.

```
f[a]
```

```
4
```

If you want to free the variable, you can use the Clear function.

```
Clear[a]
```

```
a
```

```
a
```

```
f[a]
```

a^2

Similarly, functions can be cleared by using Clear on the name of the function.

```
Clear[f]
```

```
f[a]
```

```
f[a]
```

When you give *Mathematica* something that isn't defined or it doesn't know what to do with it, it will just return the input.

A useful function for clearing all variables and functions is the following.

```
ClearAll["Global`*"]
```

If you want to define multiple variables at once it can be useful to define them in a "list". A list in *Mathematica* is just a string of numbers or variables separated my commas, enclosed by curly { and } braces.

```
{a, b, c} = {1, 2, 3}
```

```
{1, 2, 3}
```

```
c
```

```
3
```

Similarly, you can save a list as a variable.

```
list = {1, 2, 3}
```

```
{1, 2, 3}
```

```
list
```

```
{1, 2, 3}
```

If you use a function on a list, it will apply the function to each term in the list and return the results as a list.

```
f[x_] := x^2
```

```
f[list]
```

```
{1, 4, 9}
```

```
ClearAll["Global`*"]
```

It can be tiresome to come up with new names for variables which you have defined as constants so it is sometimes easier to label them by an index.

```
{c[1], c[2], c[3]} = {1, 5, 7}
```

```
{1, 5, 7}
```

```
c[2]
```

```
5
```

This is especially defining a function by telling *Mathematica* its exact value for various arguments. If you try to use it one something you haven't defined, as we saw before it will give back your input.

```
c[4]
```

```
c[4]
```

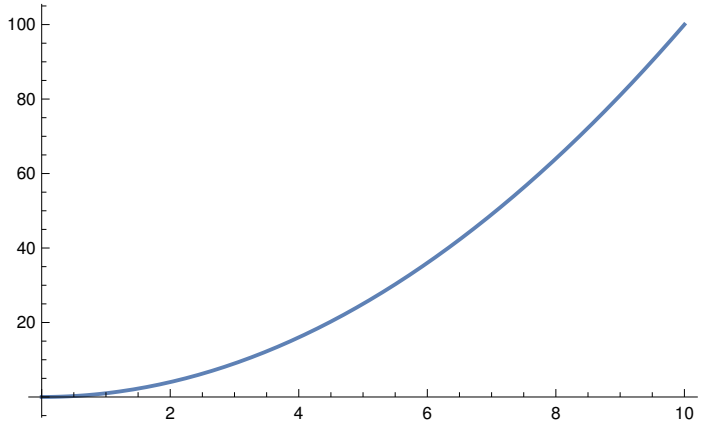
Plotting basics

The basic format of plotting is Plot[function,{variable, lower bound, upper bound},Options]

Here I am using the option PlotStyle -> Thick to make the line easier to see. The arrow is made out of a minus sign followed by a greater than symbol .

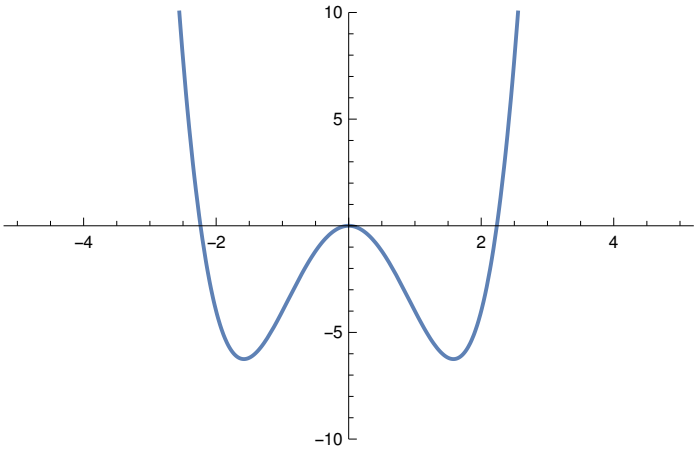
```
f[x_] := x^2
```

```
Plot[f[x], {x, 0, 10}, PlotStyle → Thick]
```



You can also directly enter the function instead of defining one before. Here I am using the Option PlotRange -> {lower bound in y axis, upper bound in y axis}. *Mathematica* is usually pretty good at guessing a good range for the y axis but it is often nice to be able to specify exactly.

```
Plot[-5 * x^2 + x^4, {x, -5, 5}, PlotRange → {-10, 10}, PlotStyle → Thick]
```



You can also define a function of more than one variable.

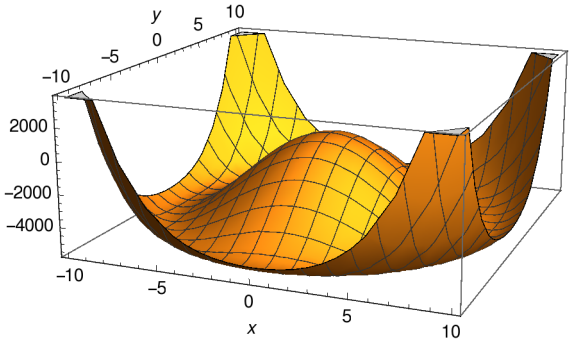
```
g[x_, y_] := -150 (x^2 + y^2) + (x^2 + y^2)^2
```

```
g[5, 5]
```

```
-5000
```

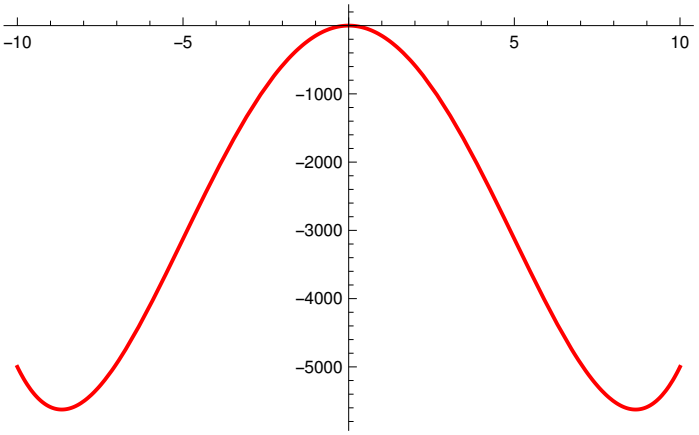
Similarly, you can plot it using Plot3D. I'm also using the AxesLabel option here, as knowing your orientation can be tougher in a 3D plot.

```
Plot3D[g[x, y], {x, -10, 10}, {y, -10, 10}, AxesLabel → {x, y}]
```



If you just want a cross-section of the function you can enter a specific value in for one of the variables and plot it in 2D using Plot

```
Plot[g[x, 0], {x, -10, 10}, PlotStyle → {Thick, Red}]
```



There are **MANY** options you can use with Plot. If you want to see them, you can go to the documentation either by going to “Help -> Documentation Center” or simply by clicking somewhere in the function you are interested in and clicking F1.

Derivatives and Integrals

```
f[x_] := Exp[-x^2]
```

The basic format of a derivative is D[function, variable]

```
D[f[x], x]
```

```
-2 ℯ^(-x^2) x
```

or as a shortcut you can use ‘ to take a derivative of a function

```
f'[x]
```

```
-2 ℯ^(-x^2) x
```

This can make it quicker to take higher order dervatives.

```
f''[x]
```

```
-2 ℯ^(-x^2) + 4 ℯ^(-x^2) x^2
```

But careful! You can’t use ‘ to take a derivative when it is a function of more than one variable as it doesn’t specify which variable you want to take the derivative with respect to

```
h[x_, y_] := x^2 y + y^2 x
```

```
h'[x, y]
```

```
h′[x, y]
```

```
D[h[x, y], x]
```

```
2 x y + y^2
```

```
f[x_] := x^2
```

The basic format of an indefinite integral is the same as a derivative Integrate[function,variable]

```
Integrate[f[x], x]
```

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

If you want to do a definite integral, the format is Integrate[function,{variable, lower bound, upper bound}]

```
Integrate[f[x], {x, 0, 10}]
```

$$\frac{1000}{3}$$

When you get to more complicated functions, it is possible *Mathematica* won't be able to solve them in an exact form. For these you can use NIntegrate to numerically integrate the function.

```
NIntegrate[f[x], {x, 0, 10}]
```

```
333.333
```

Similarly, if you simply include a number that includes a decimal somewhere, *Mathematica* will assume you want a numerical result

```
Integrate[f[x], {x, 0, 10.}]
```

```
333.333
```

Limits

The basic format of a limit is Limit[function,variable -> value

```
Limit[Sin[x]/x, x → 0]
```

```
1
```

Here you see you can even take a derivative using the definition of a derivative.

```
Limit[((x + h)^3 - x^3)/h, h → 0]
```

$$3\,x^2$$