Getting Started with Mathematica

Introduction

What is Mathematica?

Mathematica is a computer program for doing mathematics. It is often used for instruction, homework, research, and writing. Mathematica is extraordinary well-rounded. It is suitable for both numeric and symbolic work, and it has remarkable word-processing capabilities as well. Mathematicians can (and often do) search for a working model, do intensive calculation, and write a dissertation on the project (including complex graphics) -- all from within Mathematica.

Its weaknesses include a somewhat steep learning curve, an interface that is difficult to use from the command line, and rather complex installation procedures.

Philosophical issues

Mathematica is more than just a program. It is an attempt to redefine the interaction between mathematician and computer. *Every* command, function, option, button, menu, or object in Mathematica fits into this philosophy in a standard way. In the mid-to-long term, this makes Mathematica a very easy-to-use, enjoyable tool. For the new user, Mathematica can seem opaque and threatening.

Mathematica was created by a mathematician for other mathematicians. A user of Mathematica should be comfortable dealing with abstraction, generalization, and pattern discovery. If you feel you do not fit this profile, you might want to learn Maple first.

How to use this document

This document is intended for new users of Mathematica. No prior math software experience is assumed, though we do point out differences between the major packages along the way. We assume you are familiar with your operating system, so we will not cover clicking, dragging, etc. Since Mathematica is quite visually oriented, we assume you will be using it on a computer with a windowing system (such as Windows, Macintosh, or X-windows). If you intend to use Mathematica from the command line, it would be best to first learn it in a graphic environment, and then learn to Use Math Software under UNIX.

We will use a few conventions throughout this document.

Example	Explanation
File -> Open	Select *Open* from the *File* menu.
3,Ctl-6,5	Press 3, then hold the Control key and press 6, then press 5.
`2+2`	Input to be typed at the Mathematica prompt.
`4`	Output from Mathematica
	An important tip.
	A tip for Maple users.
	A tip for Matlab users.

Where to find Mathematica

Mathematica is available in all Macintosh, SGI, and Sun computer labs at IUB. It is also available on Steel and the SP (node sp09).

Mathematica is also available for sale in the IU Bookstore.

How to start and exit Mathematica

To start mathematica, follow these instructions.

- Unix: from a command line, type "mathematica &"
- Macintosh: Apple Menu->Stat/Math->Mathematica

	If you are using X windows and you get a stream of font errors, follow these instructions from the Knowledge Base.

To exit Mathematica, you can:

- Choose File->Exit
- Type `Quit[]`.

Orientation

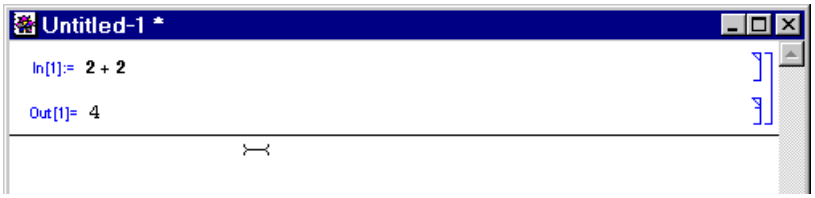
When you first start Mathematica, you should see a "splash" screen with the Mathematica logo, version, and license

information. When the program loads, you should see several objects on the screen. We will now describe what they are and what they do.

TIP	You may wish to turn on your speakers (or bring headphones if you are in an STC). Mathematica uses audio cues to notify the user of errors, finished calculations, etc.

Notebooks

A notebook is a collection of Mathematica statements, output, and graphics. The concept is like that of a "document" in a word processor. You enter information and commands into the notebook window, and the output (if any) is displayed there.



If the notebook has been modified since it was last saved, an asterisk (*) will appear in the title bar. To save your work, choose File->Save As... or File->Save.

If Mathematica is ready for new input, the cursor will flip sideways (see above). Just start typing to enter information. Try typing this (don't press return yet):

```
2 + 2
```

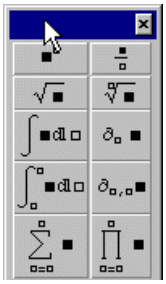
To tell Mathematica to evaluate this expression, hold down Shift and type Return. Since Mathematica is also a word processor, it needs to know if you want to evaluate the expression, or just insert a carriage return-linefeed. This can be quite confusing to the new user.

System	Evaluate	Linefeed
Macintosh	Enter *or* Shift-Return	Return
Windows	Shift-Enter	Enter
X	Shift-Return	Return

Next, look at the blue symbols along the right side of the notebook. Each group of statements enclosed by the triangle-brackets ( ) is called a *cell*. The cell is the smallest unit of work in Mathematica. A cell may contain input or output, math or comments, text or graphics.

	Cells in Mathematica are like *execution groups* in Maple.

Palettes



You should see a box with a lot of symbols floating on the right side of your Mathematica window. This is called a *palette*. Palettes allow you to easily insert complicated mathematical notation. For example, to compute the cube root of 34, you could click on the  . Type `34`, which should appear under the root sign. Then click on the small square above the root, and type `3` in the box. Finally, evaluate the expression.

Using the default palette, you can enter fractions, integrals, summations, matrices, subscripts, and most greek letters. Of course, there are many other palettes available - choose File->Palettes to see a list.

Kernels

Mathematica is actually split into two conceptual pieces, the *front-end* and the *kernel*. When you start Mathematica, you are actually only starting the front-end.

The front-end handles input and output to the user, access to the file system, and creates graphics on your screen. Most users will deal primarily with the front-end.

The kernel does nearly all computation (excluding graphics rendering). When you evaluate any expression, the kernel does the hard work and sends the results back to the front-end, which then displays it in an attractive format for the user.

Most users will run the front-end and the kernel on the same computer. If your computer is connected to a network, you can run the kernel on a more powerful machine, while running the front-end on your favorite computer. To learn how to do this, choose Kernel->Kernel Configuration Options, click Add, and then click Help.

How to stop a runaway calculation:

- Macintosh - Command-Comma or Control-C
- Windows and X - Alt-Comma or Control-C

This will bring up a menu that allows you to view the state of the kernel, abort the calculation, etc.

Help system

Mathematica has an excellent help system. To get general help, choose Help->Help Browser. Browse among the topics listed.

Mathematica comes with an excellent resource, *The Mathematica Book*. It is available through the Help Browser. See also the Getting Started section, which contains several excellent tutorials.

Doing Math

Arithmetic

Basic arithmetic operations are easy to enter in Mathematica. You may use your choice of notation, and you may choose to enter symbols via the palettes or the keyboard.

Keystrokes	Input	Output	Comments
4, ,+, ,4 3,-,1,6	`4+4` `3-16`	`8` `-13`	Addition, subtraction. Spaces are optional.

4,*,4
4, ,4
4,4
4,Esc,*,Esc,4
	`4*4`
	`4 4`
	`44`
	`4×4`
		`16`
		`16`
		`44`
		`16`
			Multiplication.
			A space can mean multiplication as well...
			...but you should be careful.
			You can also use the symbol.

3,/,4
 	
3,Esc,d,i,v,Esc,4
 	
3,Ctl-/,4
	`3/4`
	
	`3÷4`
	
	$\frac{3}{4}$
		$\frac{3}{4}$
		
		$\frac{3}{4}$
		
		$\frac{3}{4}$
			Exact division.

3,^,5 3,Ctl-^,5	`3^5` 3^5	`243` `243`	The text version The symbolic version

3,^,(,1,/,2,)
Ctl-@,3
N,[,Ctl-@,3,Ctl-Space,]
	`3^(1/2)`
	$\sqrt{3}$
	`N[√3]`
		$\sqrt{3}$
		$\sqrt{3}$
		`1.73205`
			An approximation.

%,Ctl-^,2	$\%^2$	`3.`	Square last output.

Mathematica's data restrictions are as follows:

- Integers can be of any size, up to your computer's memory.
- Real and complex operations are carried out with arbitrary (limitless) precision, up to your computer's memory, unless `N` is used.
- See Help->Help Browser: Numerical Computation->Numerical Precision->N, and section 1.4.9 of *The Mathematica Book*.

	• Mathematica does not set limits on the size of matrices, as Maple does. • `N[...]` is just like Maple's *feval*.
	If you want to use double precision, you must explicitly say so. See above.

Algebra

Mathematica is more than a calculator, of course. We will now try out some of Mathematica's Computer Algebra System (CAS) features.

Keystrokes	Input	Output	Comments
a	a	a	All names are symbols.
a,=,4	a=4	4	Defining a constant.
Esc,a,l,p,h,a,Esc,=,3	α=3	3	Greek letters (see palettes).
(,x,+,1,),Ctl-^,2	(x+1)^2	$(x+1)^2$	An expression.
E,x,p,a,n,d,[,%,]	Expand[%]	$1+2x+x^2$	Algebraic expansion....
S,i,m,p,l,i,f,y,[,%,]	Simplify[%]	$(x+1)^2$	...and simplification.

From this point onward, we will assume that you know how to enter keystrokes.

Input	Output	Comments
p=x^3+3x^2+3x+1	x^3+3x^2+3x+1	p is now an alias for the expression. This is not mathematical equality.
Solve[p==0]	{{x->-1}, {x->-1}, {x->-1}}	Solve the *equation* p==0. Mathematica returns a set of *transformation rules*. The system has three identical solutions.
p /. x->2	27	Substitute 2 for x (temporarily).
p==0 /. x->2	False	27 does not equal 0.

	Transformation rules are like the *subs* command in Maple.
	Go exploring in the Algebra Palette: File->Palettes->AlgebraicManipulation.

Functions

Using them

A *function* in Mathematica is a formalized, named transformation rule. A function may return a symbol, a real

number, a complex matrix, another function.... Any mathematical object is an acceptable value for a function. We have seen several built-in functions so far: **Solve**, **Expand** , and **N** are a few examples. A function may be written using mathematical notation, it may be created using programming constructs (like for, if/then, etc.), or it could even be written in another language entirely (Fortran, C, etc.).

All built-in Mathematica functions are named with InitialCapitalLetters (**FullSimplify**) and some use abbreviations (**NDSolve** - Numerical Differential Equation Solver).

Creating them

Input	Output	Comments
`f[x_]:=x^2`	(none)	A simple function definition. Note the underscore (_) that follows the dependent variable name.
`f[2]` `f[y+z]`	`4` $(y+z)^2$	The function works with numbers and expressions.
`f[f[f[2]]]`	`16348`	You can nest functions.
`?f`	`Global'f` $f[x_] := x^2$	What is f?
`g[x_,y_] := x*y`	(none)	You can have as many arguments as you like.
`Clear[f]`	(none)	Deletes the definition of f.

Procedural Functions

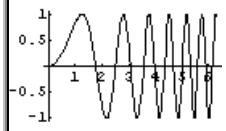
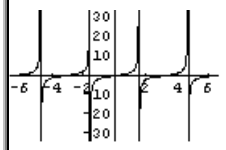
You may wish to create more complex functions. Suppose that you wanted to count to 100, and print out a list of each number that is divisible by the argument. Here is one way that you could do it:

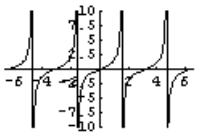
```
sillyCount[n_] := ( Do[ If[Mod[i,n]==0, Print[i]] ,{i,100}]; )
```

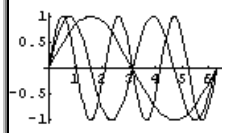
For more information on programming in Mathematica, see the online help.

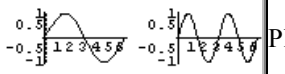
Graphics

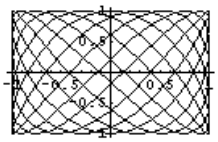
2-D Graphics

Input	Output	Comments
`Plot[Sin[x^2],{x,0,2Pi}]`		A simple plot.
`Plot[Tan[x], {x, -2Pi, 2Pi}]`		If the plot has singularities, Mathematica will try to choose a suitable scale....

`Plot[`
`  Tan[x],`
`  {x, -2Pi, 2Pi},`
`  PlotRange -> {-10,10}]`
		...or you can set it manually.

`Plot[`
`  {Sin[x],Sin[2x],Sin[3x]},`
`  {x, 0, 2Pi}]`
		You can plot multiple functions by
		including them in curly brackets.

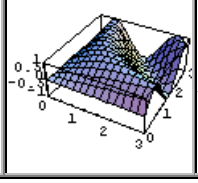
`Show[GraphicsArray[`
`  {Plot[Sin[x],{x,0,Pi}],`
`  Plot[Sin[2x],{x,0,2Pi}]`
`  }]]`
		Plot two plots, side-by-side.

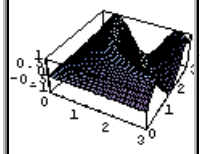
`ParametricPlot[`
`  {Cos[7t],`
`Sin[11t]},`
`  {t, 0, 2Pi}]`
		Parametric plot.

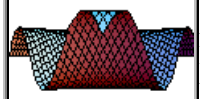
For more information, see section 1.9.1 in *The Mathematica Book*.

3-D Graphics

Input	Output	Comments

`Plot3D[`
`  Sin[x y],`
`  {x,0, 3},`
`  {y, 0, 3}]`
		A basic surface plot.

`Plot3D[`
`  Sin[x y],`
`  {x, 0, 3},`
`  {y, 0, 3},`
`  PlotPoints -> 40]`
		You can get a finer plot if you increase the number of sample points.

`Plot3D[ Sin[x y],`
`  {x, 0, 3},`
`  {y, 0, 3},`
`  ViewPoint -> {1, 1, 0},`
`  PlotPoints -> 40,`
`  Boxed -> False,`
`  Axes -> False]`
		A variety of options.
		Most important is the ViewPoint option, which tells where to put the "camera".

For more information, see section 1.9.7 of *The Mathematica Book*.

Further Reading

If you have enjoyed working with Mathematica, you will almost certainly enjoy *The Mathematica Book*. It is available on the web, through Stat/Math Center Sales, at your local bookstore, and in the online help.

In addition, be sure to wander through the Wolfram Research Documentation Center, where you will find reference works on any topic imaginable.

You can find other people who use Mathematica in the following usenet newsgroups:

- comp.soft-sys.math.mathematica
- sci.math.symbolic (Mostly Maple discussion, but some Mathematica)

In addition, your local bookstore or library probably has several books about Mathematica or one of its applications. WRI keeps a list of current books on their website.

Class Assignments

1. calculate $2(4+\frac{3}{4})^5$
2. using % calculate the sqare-root of the result.
3. solve the equation $\frac{1}{x+1}=4$
4. write (EXACTLY)
   Sove [1/(x+1)==4]
   and press SHIFT-ENTER. You should get an error message from Mathematica indicating that it does not recognize the command "Sove" and giving possible similar command names. Now, place the cursor on the last output line and add the missing "l". The command re-appears in a new line and you can execute it by pressing SHIFT - ENTER . Next, click on the] sign on the right side of the screen, in the region where you have entered the wrong command name. This is how you mark a region. To delete it hit the delete key. In this manner you can keep your notebook organized.
5. Plot the polynomial $x^3+3x^2-50x+2$ in the interval [-20,20] using the Plot command.
6. Plot sin(xy) in the interval 0<x,y<3 using the Plot3D command.
7. plot Bessel's function $J_2(x)$ in the interval [-30,30]. Find the required command name in Mathematica's online help.
8. calculate the integral $\int \frac{dx}{a^2-x^2}$
9. calculate the integral $\int_0^\infty e^{-\frac{x^2}{2\sigma^2}}dx$. What is the meaning of the result?

You can learn a lot more about mathematica using its help. There are examples for almost everything you may think of!

Common Errors and Their Solutions

Idea: These are common mishaps that arise when using *Mathematica* and something is typed in wrong. So before you get upset and throw the computer across the room, read through this and see if a simple solution can be found.

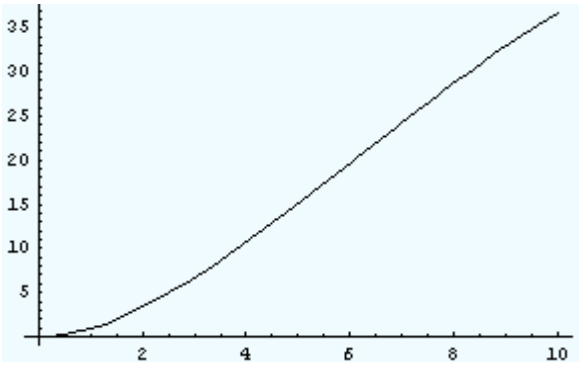
The most common errors that I have seen throughout the course occur because of capitalization mix ups. When we use letters to define functions and variables, we should always use lower case letters. And the reason for that is that *Mathematica* has a lot of pre-programmed functions in it's memory, all of which have their first letters capitalized. Have you ever noticed that? Look at Plot, or Sin, or Clear, or Expand, they all have capital letters starting them off. So if you try to use a capital letter to define a function, *Mathematica* might be fussy. Consider yourselves forewarned....

Now let's see what happens when we try to use *Mathematica* to perform a function, but we forget to capitalize the function name.

```
In[91]:= Clear[f, x]
         f[x_] = x^2 E^(-0.1 x)
Out[92]= e^(-0.1 x) x^2
In[93]:= plot[f[x], {x, 0, 10}]
Out[93]= plot[e^(-0.1 x) x^2, {x, 0, 10}]
```

Notice the lower case 'p' in plot. So all *Mathematica* is just telling you that yeah, you typed in 'plot[f[x],{x,0,10}]' and nothing more, since it didn't recognize the plot command. Let's see what happens when we capitalize it:

```
In[94]:= Plot[f[x], {x, 0, 10}]
```




```
Out[94]= - Graphics -
```

Beautiful.

Let's look at some of the other functions that are in *Mathematica's* memory, without capital letters:

What happens when we try to take the derivative of sin[7x].

```
In[95]:= Clear[x]
         D[sin[7 x], x]
Out[96]= 7 sin'[7 x]
```

See what happened? *Mathematica* thought sin was an unknown function name, like the f in f[x]. That's why there's a sin' in front of the [7x], that's *Mathematica*'s way of

accounting for taking the derivative of that function. This is what the answer should look like:

```
In[97]:= D[Sin[7 x], x]
Out[97]= 7 Cos[7 x]
```

We all knew that, right?

This is my favorite. You know our good friend the natural base, or e? Well, e is nothing but a number right? Well, that number is also another pre-programmed *Mathematica* function. And I'd have to say, this is the most common mistake made in problems in these courses. See if this looks familiar to you:

```
In[98]:= Clear[f, x]
         f[x_] = 47 e^(10 x)
Out[99]= 47 e^(10 x)
In[100]:= D[f[x], x]
Out[100]= 470 e^(10 x) Log[e]
```

Uh, huh. How about another:

```
In[101]:= Clear[f, x]
          f[x_] = Sin[x^2 e^x]
Out[102]= Sin[e^x x^2]
In[103]:= D[f[x], x]
Out[103]= Cos[e^x x^2] (2 e^x x + e^x x^2 Log[e])
```

Something's kind of fishy here. Let's check out one more:

```
In[104]:= Clear[f, x]
          f[x_] = e^x
Out[105]= e^x
In[106]:= D[f[x], x]
Out[106]= e^x Log[e]
```

What the heck is that Log[e] doing in all of those?? I'll tell ya. It's there, because *Mathematica* doesn't know that you want it to be THE E, it just assumes it's some variable, like a or b.

```
In[107]:= Clear[f, x]
          f[x_] = a^x
Out[108]= a^x
In[109]:= D[f[x], x]
Out[109]= a^x Log[a]
```

See? So how do we fix it? You got it, capitalize your E. Or there is one you can use on your palettes that you find under File->Palettes->Basic Input. Check it out:

```
In[110]:= Clear[f, x]
          f[x_] = 47 E^(10 x)
Out[111]= 47 ℯ^(10 x)
In[112]:= D[f[x], x]
Out[112]= 470 ℯ^(10 x)
In[113]:= Clear[f, x]
          f[x_] = Sin[x^2 E^x]
Out[114]= Sin[ℯ^x x^2]
In[115]:= D[f[x], x]
```

```
Out[115]= (2 e^x x + e^x x^2) Cos[e^x x^2]
In[116]:= Clear[f, x]
          f[x_] = E^x
Out[117]= e^x
In[118]:= D[f[x], x]
Out[118]= e^x
```

Much better.

There are other things that *Mathematica* is very picky about that aren't so obvious. Take multiplication for one. If you want to multiply two things together, whether it is a number and a number, a number and a variable, or a variable and a variable, all you need to do is put a space in between the two. Or the three, or the four, you get the picture. See how it works:

```
In[119]:= Clear[x]
In[120]:= 4 4
Out[120]= 16
In[121]:= 4 x
Out[121]= 4 x
In[122]:= x x
Out[122]= x^2
```

Here's without the space:

```
In[123]:= 44
Out[123]= 44
In[124]:= 4 x
Out[124]= 4 x
In[125]:= xx
Out[125]= xx
```

So the 4x worked all right, but the other two obviously didn't. So it's safe to say that we should just always go with the space. And of course, you can use the asterisk also:

```
In[126]:= 4 * 4
Out[126]= 16
In[127]:= 4 * x
Out[127]= 4 x
In[128]:= x * x
Out[128]= x^2
```

Let's look at a case where it may not be so obvious. Let's say we have a function of x with a couple constants thrown in that aren't actually numbers:

```
In[129]:= Clear[f, x, a, b]
          f[x_] = abx^2
Out[130]= abx^2
```

Looks okay at this point, right? Let's see what happens when we take the derivative:

```
In[131]:= f'[x]
Out[131]= 0
```

Whoa, what happened there? Well, there were no spaces between any of the letters, so *Mathematica* couldn't distinguish the constants from the variables. Now look what happens when we add the spaces:

```
In[132]:= Clear[f, x, a, b]
          f[x_] = a b x^2
Out[133]= a b x^2
In[134]:= f'[x]
Out[134]= 2 a b x
```

Just like we thought it should be.

Another problem that is very hard to find and catch is when we forget to use the Clear function on the variables and functions we wish to use in a problem. Let's see an example where this problem occurred.

Say you are given the function f[x] = E^x + 3, and you are asked to find the value of f[0]. So you do this:

```
In[135]:= Clear[x];
          x = 0;
          f[x_] = E^x + 3
Out[137]= 4
```

Just what we thought it should be.
Now you are asked to do something with the function Cos[x]. So you define:

```
In[138]:= f[x_] = Cos[x]
Out[138]= 1
```

...but that is not the output you expected. What happened here. *Mathematica* still thinks that the value of x should be 0. As a result Mathematica evaluates f[0] and returns the answer. You should have done this:

```
In[139]:= Clear[f, x]
          f[x_] = Cos[x]
Out[140]= Cos[x]
```

Perfect, just what you wanted.

Similar to forgetting to clear variables and functions is forgetting to re-run cells when you start in the middle of a lesson. Let's say yesterday you quit working on G.7.b.ii. and saved your work. Then today you come in and start back at G.7.b.ii, but the results you get are very strange. The problem is that you did not run all the previous cells in earlier parts of the problem. Go back to the beginning of the problem, in this case G.7 and run all the cells for that problem. *Mathematica* does not remember all the stuff you did in previous times using it.

Another problem that is easy to overlook is when you want *Mathematica* to display the results of what it is doing, but it will not. Most likely you put a ";" at the end of the statement. For instance you want to know the value of Sin[.3]. So you type:

```
In[141]:= Sin[.3];
```

...but nothing comes out. This is because there is a semicolon at the end of the statement which tells *Mathematica* not to display the results of the calculation. So get rid of the ";" and see what happens.

```
In[142]:= Sin[.3]
```

```
Out[142]= 0.29552
```

Exactly what we wanted.

Something to remember: *Mathematica* does not distinguish between what you normally think of as Log and Ln. To *Mathematica* Log is actually Ln. So when you see Log, it is actually what your normally call Ln.

A funny quirk of *Mathematica* is the fact that when you wish to define a function, say f[x], you have to put a "_" after the variable name. Let's see what I mean.

```
In[143]:= Clear[f, x];
          f[x] = x^2 + 2
Out[144]= 2 + x^2
```

So then you try to evaluate f[x] for x=10.

```
In[145]:= f[10]
Out[145]= f[10]
```

What happened here? *Mathematica* should have returned 102, not f[10]. We forgot to put the "_" after the x. Take a look here.

```
In[146]:= Clear[f, x];
          f[x_] = x^2 + 2;
          f[10]
Out[148]= 102
```

That's what we wanted.

Are there any mistakes that you commonly make that we did not list here? Please email them to zvikas@post.tau.ac.il