Crash Course on Mathematica

Mathematica

Mathematica is a program that allows you to:
*) Make numeric and symbolic calculations.
*) Simplify complicated mathematical expressions.
*) Compute derivatives and integrals.
*) Solve equations.
*) Plot graphs in 2D and 3D.
*) Create animations.
*) Write programs and manipulate data.

Advantages:
*) It is easy to use.
*) It allows for symbolic calculations.
Disadvantages:
*) You need to pay to have it (students have good discounts).
*) It is not as efficient as other languages as Fortran or C when dealing with very involved computations. But it is a good way to learn the basic of programming and then move to other languages.

Format, Save and Reopen

You can use *Mathematica* simply as a word processor, as I have done so far.

1) Write:
Lecture 1 (date)

2) Notice that a bracket appeared at the far right. This is called a cell.

3) Click on the cell so that it is highlighted.

4) At the Tool Bar, go to Format -> Style -> Title
(play with the various other options for styles, fonts, text color, etc)

5) Follow items 1, 2, 3 above and now select at the keyboard "Alt+1". You get the same effect.
(in Format-> Style, you can find the equivalent "Alt+." for each style)

6) Delete one of the two cells by clicking on one of the right brackets and pressing "delete".

7) Save this file in the folder you created:
File -> Save As
Notice that the default name of the file was "Untitled-1" and you changed it to your desired name, which now appears at the title bar of the window.

8) Close *Mathematica.*

9) Go to the folder you created and click twice on the file you saved to open *Mathematica* and the file together with it.

The Basics

To execute an instruction, press SHIFT + ENTER

1) Write
The Basics
"Alt+4"

2) Write
To execute an instruction,
type the instruction and press SHIFT+ENTER
"Alt+5"

3) Type 2+3 and then press SHIFT+ENTER (i.e. hold down the Shift key and then the Enter key)

4) Click on the input cell,
Ctrl+C to save
Ctrl+V to paste it in a new cell
Add +8

Comments

A comment is a useful way to remind us of important details. It is written in between
(* ... *)

1) For example :
On the same cell, type
(* I can perform different calculations on the same cell*)
2 + 3
2 - 3
2^3
9^(1/2)
3!

2) Another example :
On the same cell, type
2*3
(* To multiply I can use the symbol * or just give a space between the numbers *)
2 3

Exact vs Approximate

On the same cell, type
2/3
2./3.
2./3
2/3.
What is the difference?
Mathematica sees 2 and 3 as exact numbers and for 2/3, it simply gives the exact fraction. To get an approximate answer to the division of 2 by 3, we need to use the decimal representation to at least one of the numbers.

Functions

Mathematica has thousands of built-in functions.
CAREFUL!
1) They always start with a *CAPITAL* letter. All *Mathematica*-defined symbols, commands and functions begin with a capital letter.
2) SQUARE BRACKETS need to be used for the function arguments.

Type

```
Sqrt[9]
Sqrt[10]
Sqrt[10.]

Sqrt[-16]
(* imaginary number I *)
Re[2+4 I ]
Im[2+4 I ]
Abs[2+4 I ]

(* Pi *)
Pi
1. Pi
N[Pi, 20]

(* natural logarithm *)
E
Exp[1]
Exp[1.]
N[E,20]
Exp[-3.]

6!
0!
```

Cos[Pi/3]
Cos[60.]
Cos[60. Degree]
ArcCos[0.5]
Pi/3.

Brackets

1) SQUARE BRACKETS are used for the function arguments: Sqrt[9]
2) ROUND BRACKETS are used for grouping: (2+3)*4 and NOT [2+3]*4
3) CURLY BRACKETS are used for lists: {1,2,3,4}

To open a new window

File -> New -> Notebook
or
CTRL+N

Variables

We can introduce variables and give them values.
-) Example:
a=2
b=3
a+b
From now on, whenever you type "a", *Mathematica* will equal it to 2. This will only disappear from its memory if you assign a new value to it, close *Mathematica,* or if you use the **Clear** function.
-) Type:
Clear[a]
a+b

*) You cannot start a variable name with a number.
2x is not accepted, but x2 is.
*) You cannot use words or letters that already have a meaning in *Mathematica*, such as E, N, Pi
-) Example:
Type:
E=4
Mathematica will complain.
Click on ">>" to find out why it did not like your choice.

Suppressing output

In the middle of long computations, you may want to hide some of the outputs.
-) Example: suppose you only want to see the final answer for a+b in
a=2
b=3

a+b^2
You can use semicolons for the first two lines, so their outputs will not appear.
Type:
a=2;
b=3;
a+b^2

Command "Print"

We may want to add text to the output, which we can do with "Print"
Type:
Clear[a,b,result];
a=Sqrt[2387];
b=Log[339.];
result=a*b;
Print["The result of my calculation was ", result];

Palettes

To compute the square root of 12, you can use any of the alternatives:
(i) Sqrt[12.]
(ii) 12.^(1/2)
(iii) From the menu bar: Palettes -> BasicMathAssistant -> select the square root symbol $\sqrt{\square}$ and you can now write the number 12. inside.

Save your work and how to print

Before we continue, it may be a good idea to save your works so far:
From the menu bar: File -> Save

If you ever decide to print it, use
File -> Print

Abort

Sometimes we might want to or need to interrupt a calculation, either because it will take days to give an answer or because we realized there was a mistake in the program we wrote.
From the menu bar: Evaluation -> Abort Evaluation
-) Example:
Abort the computation of 5^10000000

Getting help

Mathematica has various files and tutorials that explain how to use the program. A lot can also be found online.
From the menu bar: Help -> Wolfram Documentation
You can either select one of the topics from the list or use the "SEARCH" field, if you know which command you want to have more information about.

Sums and Products

Sum

-) Add the squares of the first 10 positive integers

```
Sum[ x^2, {x, 1, 10}]
```

-) Add the first 10 even numbers

```
Sum[ k, {k, 2, 20,2}]
```

Product

Multiply the first 10 primes

```
Product[ Prime[x], {x, 1, 10}]
```

Limits, Derivatives, Integrations

Limit

```
f[x_] := Sin[x]/x
f[0]
```

```
Plot[ f[ x ], {x, −10, 10} ]
```

```
Limit[ f[x], x −> 0 ]
```

Derivative

In[166]:=
```
poly[x_] := x^4 − x^3 + 2 x + 1
dpoly = D[poly[x], x]
```

Maxima and Minima

```
Clear[f]
f[r_] := r/2 - Pi r^3;
Plot[f[r], {r, -0.5, 0.5}]

g = D[f[r], r];
Solve[g == 0, r]
FindRoot[g == 0, {r, 0.2}]

FindMaximum[f[r], {r, 0.2}]
FindMaximum[f[r], {r, 0}]
FindMaximum[f[r], {r, -1}]

FindMinimum[f[r], {r, 0}]
```

Integration

```
(* Indefinite integral *)
Integrate[x, x]
Integrate[x^4 - 3 x^2, x]

(* Definite integral *)
Integrate[1/(1 + x + x^2), {x, 0, 1}]
Integrate[Sin[x]^p, {x, 0, Pi}]
Integrate[Integrate[x^2 + y^2 + 1, {y, 0, 2 - 2 x}], {x, 0, 1}]
Integrate[x^2 + y^2 + 1, {x, 0, 1}, {y, 0, 2 - 2 x}]

(* Numerical integration *)
Integrate[Exp[Sin[x]], {x, 0, 1}]
NIntegrate[Exp[Sin[x]], {x, 0, 1}]
```

Plots

```
f1=Plot[Cos[x], {x, 0, 8 Pi}, PlotStyle → {Red,Dashed}, AxesLabel → {"x","cos"}]
f2=Plot[Cos[2x], {x, 0, 8 Pi}, PlotStyle→ Black]
Show[{f1,f2}]

Plot3D[Cos[x] Sin[2 y], {x,0,24}, {y,2,7}]
```

Vectors and Matrices

```
vec= Table[k, {k,2, 10, 2}];
mat = Table[ Table[ i+j, {j,1,3} ], {i,1,3}  ];

MatrixForm[vec]
MatrixForm[mat]
```

To extract an element of the vector “vec”:

```
vec[[2]]
```

To extract an element of the matric “mat”:

```
mat[[2,3]]
```

CAREFUL! For Mathematica, the eigenvectors are the ROWS, while
for Python, Fortran and other, the eigenvectors are the COLUMNS!

```
ham = { {1., 2, 0}, {2, 1, 2}, {0, 2, 1} };
Eigenvalues[ham]
Eigenvectors[ham]
```

Random numbers

Table with 10 random numbers:

-) Integers 0 or 1

```
Table[ RandomInteger[ ] , { k, 1, 10 } ]
```

-) Integers between 2 and 5

```
Table[ RandomInteger[ { 2 , 5 } ] , { k, 1, 10 } ]
```

-) Reals between 0 and 1

```
Table[ RandomReal[  ] , { k, 1, 10 } ]
```

-) Reals between 1 and 4

```
Table[ RandomReal[ { 1 , 4 } ] , { k, 1, 10 } ]
```

-) Random numbers from a normal (Gaussian, Bell) distribution with mean 0 and standard deviation 1

```
Table[ RandomReal[ NormalDistribution[ 0, 1 ] ], { k, 1, 10 } ]
```

or simply,

```
Table[ RandomReal[ NormalDistribution[   ] ], { k, 1, 10 } ]
```

Do-loop and If-statement

```
Do[
a = k^2;
Print[ a ];
,{k, 1, 4}];
```

```
Do[
a = k  + 5;
If[ Mod[a, 2] == 0, Print[ a , " is multiple of 2"], Print[ a , " is not multiple of 2"]  ];
,{k, 1, 4}];

Do[
Do[
a = k + 2 j;
Print["k=", k," j=",j," a=",a];
,{k,1,3}];
,{j,1,2}];

tot=50;
Do[
num[k] = Sqrt[3. k];
,{k,1,tot}]
lis = Table[{k,num[k]}, {k,1,tot}];
ListPlot[lis]

ListPlot[lis, Joined -> True, PlotStyle -> Black]

ListPlot[lis, Joined -> True, PlotStyle -> Black,LabelStyle→Directive[Black,Bold,Medium], AxesLabel->
{"x","y"}]

ListPlot[lis, Joined -> True, PlotStyle -> Black,LabelStyle→Directive[Black,Bold,Medium], AxesLabel->
{"x","y"}, PlotRange → {0, 30}]

ListPlot[lis, Joined -> True, PlotStyle -> Black,LabelStyle→Directive[Black,Bold,Medium], AxesLabel->
{"x","y"}, PlotRange →{{0,100}, {0, 30}}]
```

Solve: polynomial equations

***) *Mathematica* solves exactly any polynomial equation of degree less than 5**

```
Solve[x^2 + 3 x - 5 == 0, x]

Clear[roots];
roots = x /. Solve[x^4 - 2 x^3 + x + 5 == 0, x]
roots[[ 1 ]]
2. roots[[ 3 ]]

Solve[x^5 -2 x^3 + x + 5 == 0, x]
```

***) For polynomial equations of degree greater than 4, *Mathematica* can find approximate solutions**

```
Solve[x^5 -2 x^3 + x + 5 == 0, x]

NSolve[x^5 -2 x^3 + x + 5 == 0, x]
Solve[x^5 -2 x^3 + x + 5. == 0, x]
```

***) System of polynomial equations**

```
Where does the line y = x + 2 intersects the parabola y = 16 - x^2?
Plot[ {  x +2, - x^2 + 16}, {x, - 5, 4}]
Dynamic[MousePosition[“Graphics”] ]
NSolve[ { y == x +2, y == - x^2 + 16}, {x, y} ]
```

***) System of linear equations**
(linear equations are polynomial equations of degree one)

```
(i) The system has a unique solution
2x+y+z=7
x-4y+3z=2
3x+2y+2z=13

Solve[{2 x + y + z ==7, x - 4 y + 3 z ==2, 3 x + 2 y + 2 z ==13}, {x, y, z}]

Clear[A,b]
A = { { 2, 1, 1 }, {1, -4, 3}, {3, 2, 2}};
b = {7, 2, 13};
LinearSolve[A,b]



(ii) The system has infinite solutions
3x+2y-z+w=0
x-3z=-1
-y+w=2

Solve[{3 x + 2 y - z + w ==0, x - 3 z == -1, -y + w ==2}, {x,y,z}]

Clear[A,b]
A = { { 3, 2, -1, 1 }, {1, 0, -3, 0}, {0, -1, 0, 1}};
b = {0,-1,2};
LinearSolve[A,b]

(iii) The system has no solution
2x+y+z=7
x-4y+3z=2
3x-3y+4z=13

Solve[{2 x + y + z ==7, x - 4 y + 3 z ==2, 3 x - 3 y +4 z ==13}, {x, y, z}]
```

```
Clear[A,b]
A = { { 2, 1, 1 }, {1, -4, 3}, {3, -3, 4}};
b = {7, 2, 13};
LinearSolve[A,b]

Det[A]
```

Transcendental equations

(Equations involving exponential, logarithmic, and trigonometric functions can only occasionally be solved with Solve or NSolve, most commonly we need FindRoot)

```
Clear[x];
Solve[ x^2 == Exp[x],  x ]
NSolve[ x^2 == Exp[x],  x]
FindRoot[ x^2 == Exp[x],  {x, 1.} ]

Where do the functions Sin[x] and x^2-1 cross?
A graph of the function helps selecting an initial guess
To find where they meet:
Plot[{Sin[x], x^2 -1}, {x, -Pi, Pi}]       ( Or the root of the function:  Plot[ Sin[x] -  x^2  + 1,  {x, -Pi, Pi}] )
The two functions intersect near x= - 1 and x=1
FindRoot[Sin[x] == x^2 - 1, {x, -1}]
FindRoot[Sin[x] == x^2 - 1, {x, 1}]

FindRoot[Exp[2 x] -2 Exp[ x] + 1 ==0, {x,90}]
FindRoot[Exp[2 x] -2 Exp[ x] + 1 ==0, {x,90}, MaxIterations -> 300]

*)
```

(i) FindRoot[lhs == rhs, {x, xo}] solves the equation lhs=rhs using Newton's method with starting value xo.

Newton's method fails if the derivative of the function cannot be computed.

(ii) FindRoot[lhs == rhs, {x, xo,x1}] solves the equation lhs=rhs using (a variation of) the secant method with starting values xo and x1.

The secant method is a bit slower.

```
FindRoot[ Exp[-x] == x, {x, 1} ]
FindRoot[ Exp[-x] == x, {x, 1, 2} ]

*) System of equations
FindRoot[ { Exp[x]  + Log[y] == 2, Sin[x] + Cos[y] ==1 }, {x, 1}, {y,1} ]
```