A Very Short Introduction to Mathematica

J. T. Shen

September 24, 2020

0.1 What Is Mathematica?

From the website of Wolfram Research, the company which sells this product:

> From simple calculator operations to large-scale programming and interactive document preparation, Mathematica is the tool of choice at the frontiers of scientific research, in engineering analysis and modeling, in technical education from high school to graduate school, and wherever quantitative methods are used.

In short, it is your 18.01, 18.03, 18.06 instructors at your service, but you have to learn the syntax and how to give (correct) commands so they can serve you better.

0.2 Where Can I Get Mathematica?

[update by WT 2020: Mathematica is licensed for use by MIT students, faculty and staff on MIT-owned and personal machines, you can download a copy from IST] or, you can use Mathematica on Athena.

0.2.1 How to Use Mathematica?

[update by WT 2020: this was originally written primarily for use on Athena. If you have a copy on your own machine you can either run in the notebook mode (mathematica.exe, interpreter-based graphical notebook) or the math shell (math.exe, traditional command line interface to the interpreter). In either case you type in commands and it responds with the answer.]

- Log onto any Athena machine. Type

  ```
  add math
  ```

  ```
  mathematica
  ```

  in the shell window.

  Or, alternatively,

  - The following assumes you didn't change the default configuration of the window manager.

- Click on the footprint ("Menu") at lower left part of your screen.
- Go to "math/plotting" → "Mathematica", and select it.
- A window with white background should show up on your screen. That window is called "notebook", where you give your commands to Mathematica and Mathematica gives you back the output.
  - Optional. Go to "File" → "Palettes", where you can invoke several palettes that enable you to input by just clicking.
- Test. Type

```
2+2
```

  and press SHIFT key and ENTER key at the same time. If you see the correct answer shows up, Congratulations!!

  - The `In[1]`, and `Out[1]` means your input and Mathematica's output.
  - In Mathematica, always press SHIFT key and ENTER key at the same time to evaluate your input.

0.2.2 Give your input to Mathematica

There are generally two methods to input your expressions to Mathematica: one is in text form and another one is in traditional form. The text form input usually is faster, you type every mathematical expressions in plain text directly from your keyboard, while in the traditional form you use "Palettes" (explained below) and you only have to fill the blanks. This form is more readable. For example, say, we want to evaluate

$$\int_0^\infty e^{-x^2} dx$$

In the text form, we type from the keyboard

```
Integrate[Exp[-x^2], {x,0,Infinity}]
```

while in the traditional form, you click on an icon from the palette, and get

$$\int_{\square}^{\square} \blacksquare d\square$$

you then use `Tab` key to navigate to fill the blanks.

0.3 Elementary Mathematica Syntax

0.3.1 Define a function

Start by an example: How to define $f(x) = x^2$:

- In text form, type

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

  Or, using the palettes:

- Go to "File" → "Palettes" → "3. BasicInput". The long stripe window should show up. It will be referred to as "small window" hereafter.

- Type

  ```
  f[x_]:=
  ```

  in your notebook.

- Click the $\blacksquare^{\square}$ icon in the small window Then the same pattern shows up in your notebook, right after your input `f[x_]:=`$\blacksquare^{\square}$.

  - Type "x".
  - Press "TAB" key.
  - Type "2".

- Press SHIFT ENTER to evaluate the definition.

- Type

```
f[2]
```

and press SHIFT and ENTER keys. You should see the correct answer, 4, shows up immediately.

In Mathematica, an ordinary function $f(x) =$ is always written as `f[x_]:=` when you define it at the first time. Here `f` is called "header" and can be any string of word, like `myStep`, which easily reminds you what it is. [Note: Mathematica's built-in functions all begin with capital letter, so it is a good practice to name your functions to begin with lower case to avoid accidentally shadow the built-in functions.] ( ) is typed as `[ ]`. The important point is we type `x_` rather than simply `x`. The underscore line `_` right after `x` tells Mathematica to square *any* given input. If you omit this, Mathematica would not square your input unless the input is explicitly an `x`. You can think `:=` to be "defined as" at this moment. The `Tab` key brings you from the base (x) to the position of the exponent 2. (Alternatively, you can also directly click on the position where you want to input.)

To define $f(x, y) = x^2 + y^2$, the first part of the input would look like `f[x_,y_]:=` and similar procedures for the remaining. Also, in Mathematica, $xy = x * y$ is `x y`, with a blank in between; $x \div y$ is `x/y`.

0.3.2 Lists in Mathematica

The numbers of structure type in Mathematica is far fewer than that in most other programming languages. One of the most important types is `List`. A list is simply a collection of objects. The objects can be anything: number, string, function, ... etc, and embraced by a pair of `{` and `}`. A list can be 1-dimensional, 2-dimensional ... . You can also think a list as a vector.

If there is a general pattern of the elements in the list, we use the command `Table` to generate the list. Example: we want to generate the list `{0,1,4,9,16,25}`.

- Type

```
data=Table[i^2 , {i,0,5}]
```

and press SHIFT, ENTER keys.

- We give the list a name `data` so we can reuse it later if needed.
- Here we use `=` rather than `:=` to assign value to `data`.
- `Table` defines a list. The syntax looks like

  `Table[`*general pattern of each element*`,` *iterator, initial value, final value, increment*`]`

  In our case
  - general pattern of each element: `i^2`.
  - iterator: `i`.
  - initial value: `0`.
  - final value: `5`.
  - If the increment is not specified, as in our case, it takes default value, 1.

- Type `data[[4]]` and SHIFT RETURN. You should get 9. Here `[[4]]` means to take the 4th element of `data`.

Define a matrix

Again, start by example. Assume we want to define a 2×2 matrix

$$\text{myMatrix}=\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

In the text form, we type

```
myMatrix={{a,b},{c,d}}
```

If there is general pattern between elements, you still can use the `Table` command. Now you can go to the small window, and find an icon in the middle look like a traditional matrix with 4 blank squares inside: $\begin{pmatrix} \square & \square \\ \square & \square \end{pmatrix}$ Again, use `Tab` key to switch positions to next input.

- The command

  ```
  Eigenvalues[myMatrix]
  ```

  gives you the eigenvalues of your matrix: (λ_1, λ_2).

- The command

  `Eigenvectors[myMatrix]`

  gives you the eigenvectors of your matrix: $(\mathbf{v}_1, \mathbf{v}_2)$.

- The command

  `Eigensystem[myMatrix]`

  gives you the eigenvalues and eigenvectors of your matrix: $((\lambda_1, \lambda_2), (\mathbf{v}_1, \mathbf{v}_2))$.

0.3.3 Integration

$$\int_0^{\pi/2} x^2\, dx$$

is input, in the text form, as

```
Integrate[x^2, {x,0, Pi/2}]
```

But now you can find an icon from the small window enable you to input an integral exactly looks like traditional form.

When you use `Integrate`, then Mathematica will try to do it analytically. If you don't want analytic result, or you know the analytic result doesn't exist, then you should use the command `NIntegrate`. Mathematica then will do the integral numerically.

The general syntax looks like
`Integrate[`*integrand, variable, lower limit,upper limit*`]`

0.3.4 Plot a function

To plot x^2 from $(-4, 4)$, type

```
Plot[x^2, {x,-4,4}]]
```

To plot $x^2 + y^2$ on $x : (-2, 2)$ and $y : (-3, 3)$, type

```
Plot[x^2+y^2,{x,-2,2},{y,-3,3}]
```

again, you can use small window to get traditional input. There are quite a few options of command you can give to the function `Plot` to make your graphics look more beautiful (grids, ticks, axes labels, color, legends . . . etc.). You can find them from "Help" as described at the end of this tutorial.

0.3.5 Tensor

A tensor of rank k is essentially a k-dimensional table of values. To be a true rank k tensor, it must be possible to arrange the elements in the table in a k-dimensional cuboidal array. There can be no holes or protrusions in the cuboid.

Here is a $2 \times 3 \times 2$ tensor.

```
t = Table[i1+i2 i3, {i1, 2}, {i2, 3}, {i3, 2}]
```

when evaluated, you get

```
{{{2,3},{3,5},{4,7}},{{3,4},{4,6},{5,8}}}
```

0.3.6 Recursion

Another nice feature of Mathematica, which is very close to our mathematical thinking, is the recursion.

Mathematica allows a function to call itself in the definition of the function!

Here's an example which defines a factorial function:

```
f[1]=1

f[n_]:=n f[n-1]
```

If you type `f[10]` and evaluate, you would get 3628800, which is 10!. This feature makes many difficult-at-first-look problems become one-liner – we can code it with only one line.

0.4 Questions?

If you have any questions, a very good starting point is, go to "Help" in the menu bar of the notebook window, click on "Help ...", choose "Master Index", and then type a keyword, say, `list`, or `plot` in the input line. Mathematica has a very good documentation for each function.