

MatLab — Getting Started

GETTING STARTED:

- MatLab is loaded on several computers around campus, including the Physical Sciences Computer Lab, the Mathematics Computer Lab, and the Computer Science Lab.
- Use a thumb drive, or other memory device, or email the files to yourself. Click on the “MatLab 6.5” shortcut on the computer "Desktop" – if the shortcut is not there, then click on “start” (lower left corner) → “All Programs” → “MatLab 6.5” → “MatLab 6.5”.

MATLAB OVERVIEW:

- MatLab has 3 basic components:
 - 1) Command Window
 - 2) Plot Window
 - 3) m-files (for saving your commands as files so that you can use them again – the m-file actually becomes a command recognized by MatLab, in fact, all MatLab commands are written as m-files)

MATLAB COMMANDS:

- The MatLab “command” window should appear showing the MatLab command prompt `>>`
- Try entering the following commands into the command window (see why it is called the command window):


```
>> x = [0 1 2 3 4]
>> y = [0 1 4 9 16]
>> plot(x,y)
```
- Close the figure window (the plot window) when you are done.

Note: Instead of writing the x-values as "x = [0, 1, 2, 3, 4]", you could write the x-values as "x = [0:4]".

CREATING M-FILES:

- The area showing the “Current Directory” should start at “C:\MATLAB6p5\work”. If you are using a thumb drive (or other portable memory device), click on the “...” button to browse to change directories to your thumb drive (or other portable memory device).
- `>> pwd` (to see the working directory – should be the your thumb drive).
- `>> dir` (to see the contents of the current directory).
- Use the MatLab file editor (the MatLab file editor can be utilized by clicking on the "File" pull down menu, then click on "New", then move the pointer to the right and click on "M-file", to create a new file.
- Type the following in your new file (the first line is a comment line, not a command line – comment lines start with the % symbol – note that there is no `>>` symbol before the command lines in the file):


```
%test1.m
clear
x = [0, 1, 2, 3, 4]
y = [0, 1, 4, 9, 16]
plot(x,y)
```

- Click on the “File” button, then click on the “Save As” button. Choose your thumb drive within the "Save In:" area. Then type the name of the program in the "File name:" area – the MatLab editor will add the ".m" extension, so just name the file "test1".
- **IMPORTANT NOTE:** Filenames must not have any blank spaces, they must begin with a letter, and contain only lowercase letters or numbers. Do not use special characters such as "?", "<", etc. The file must end with the ".m" extension to be recognized by the MatLab command window (note that the .m extension is added automatically by the MatLab editor).
- You may also use a simple editor such as "notepad" to create and modify m-files (NOTE that the name of the file must include the ".m" extension must be saved in double quotes, e.g., "test1.m", to avoid having an additional extension of .txt).

RUNNING AN M-FILE FROM THE COMMAND PROMPT (>>):

- >> dir (to see the contents of the current directory; the file you created, test1.m, should be shown; there should not be a .txt extension after the .m extension).
- >> test1 (type test1, without the .m extension, to run the m-file program; note that the m-file executes the commands, with the advantage that you can save and modify the file; once the m-file is created, you can modify it with other editors, if you like).
- Close the figure window (the plot window) when you are done.

OTHER M-FILES TO TRY (use the “Save As” command to re-save your file with a new filename):

- Smoother parabola:


```
%test3.m
clear
x = [0 : 0.2 : 4]
y = x.*x
plot(x,y)
```
- To avoid printing all of the x values and y values in the command window, the ; used at the end of a matrix suppresses echoing (printing in the command window):


```
%test4.m
clear
x = [0 : 0.1 : 4];
y = x.^2;
plot(x,y)
```
- A very handy command is figure(gcf). This command brings the plot window to the front each time:

```
%test5.m
clear
x = [0 : 0.1 : 4];
y = x.^2;
plot(x,y); figure(gcf);
grid
title('Plot of Parabola Section')
xlabel('x values')
ylabel('y values')
```