

## Introduction to MATLAB

This note will introduce you to MATLAB for the purposes of this course. Most of the emphasis is on how to set up MATLAB on your computer. The purposes of this supplement are two. One purpose is to thoroughly familiarize you with the MATLAB development cycle. The other purpose is to teach you how to prepare written output for submission that completely shows your work.

**NOTE.** Members of the class may have quite different experience levels with MATLAB. These notes are written as though you are new to it. This note describes exactly how you are to proceed. Be sure that you read everything very carefully. If you are experienced at MATLAB, then you probably have preferences for your own configurations. You may configure MATLAB as you wish. However, note well that we are going to set some MATLAB defaults in order to standardize some working parts of the course. **You must prepare and submit your work exactly as I describe in this document.**

### Accessing MATLAB

You should be able to access MATLAB on TAMU machines. If you have purchased MATLAB, then install MATLAB according to the instructions on your copy. (Note: You are not required to purchase MATLAB. The Student Edition of MATLAB lacks some features that we may use in this course. MATLAB on TAMU machines should have all important features.)

### Customizing MATLAB for Math 470

We are going to customize your MATLAB workspace in order to optimize your ability to work interactively while preserving your work.

1. You will need to define a directory in your workspace to hold your MATLAB work, preferably a new, clean one. Please name the directory MY470DIR. MY470DIR is an excellent place to keep copies of the various MATLAB homework assignments in the course as well as other materials that are Math 470 related. For example, the file that you are preparing to contain this assignment (**tw0101**) should be stored there, as well as all other assignments. Now create MY470DIR.
2. The authors have prepared a large number of MATLAB routines customized for their textbook. They have compiled these on a separate web site (cf. p. 527 of the text):

`http://www.prenhall.com/washington`.

However that site requires that you download the files one at a time. For your convenience, I have collected all of these routines into a single ZIP file called MATH470.zip. You should download MATH470.zip into MY470DIR. Here is the link

<http://www.math.tamu.edu/~jon.pitts/470/2005c/Supplements/MATH470.zip>.

Click on the link and download MATH470.zip to MY470DIR.

3. Unzip the MATH470.zip file into MY470DIR. This will create two directories in MY470DIR. These two directories are MYMATLAB and TWMATLAB. TWMATLAB contains all the MATLAB routines prepared by Trappe and Washington. MYMATLAB is reserved for your custom MATLAB scripts and text files as described below.
4. Boot MATLAB.
5. We standardize the MATLAB desktop. Click on Desktop-->Desktop Layout-->Default.
6. We are going to permanently set some paths in MATLAB. Click on File-->Set Path. A "Set Path" window opens. Note that there is a long list of directories (folders) in the "MATLAB Search Path" box that MATLAB examines while you work. Click on the "Add with Subfolders" button. The "Browse for Folder" window opens. Browse until you find your MY470DIR directory (folder). Click on MY470DIR and hit "OK". The "Browse for Folder" window closes and you are back at the "Set Path" window. Re-examine the "MATLAB Search Path" box. Three new paths should have been added to MY470DIR, MY470DIR/MYMATLAB, and MY470DIR/TWMATLAB. Click "Save". (You may be informed that the system cannot save the file pathdef.m, and requesting if you want to save it in another location. If this happens, click "Yes" and choose a location. On Calclab, I saved this file in my home directory. Save and Close when done.) Click "Close". This concludes your path setting.
7. Now we'll discuss the MATLAB startup directory. You will need to go to the MY470DIR whenever you work on MATLAB work for this course. The default MATLAB desktop has a "Current Directory" window that shows you the directory that you are currently in. If that directory is not MY470DIR, then you will need to mouse to it from the "Current Directory" window.

Note that one can configure MATLAB so that you always start up in MY470DIR. The configuration procedure depends on your MATLAB installation. Check the MATLAB Help facility (see next section) and follow instructions if you wish to exercise this option.

## **MATLAB Help**

From the MATLAB desktop, click on Workspace and click on Workspace-->Help if "Help" is not already checked. Alternatively click on Help-->MATLAB Help. On a PC you may simply

press function key “F1”. This brings up the “Help Window” and access to the extensive MATLAB Help system. You can reference the Help system from the “Contents”, “Index”, or “Search” tabs in the “Help Navigator” window. In this course, we will try to use standard MATLAB features as often as possible, but you will likely find yourself using the Help system from time to time. The Help system is your friend and you should use it freely.

If you are new to MATLAB, then I strongly recommend that you take a partial tour of the MATLAB system. From the “Contents” tab in the “Help Navigator” window, you may click at the very top on “Begin Here” or slightly further down click on “MATLAB”. Either way will show you a link to “Getting Started” which you should click on. Both the Help Navigator-->Contents window and the Help window itself should now be showing you sequence of topics: “Introduction”, “Matrices and Arrays”, etc. I strongly recommend that you read all the topics and work through all the examples under “Introduction” and “Matrices and Arrays”. This is not to turn in. But it will go a very long way in familiarizing you with the MATLAB system of primary interest to us.

It is especially useful for you to see and appreciate how easily MATLAB may be used interactively. Just by working at the “>>” prompt, you can use do almost everything in MATLAB interactively. If you want to know  $5+2$ , you just type  $5+2$  at the interactive prompt `>> 5+2`, hit enter, and MATLAB will return `ans=7`. If you meant to type  $5+3$ , then you need merely type once more to enter `>> 5+3` and MATLAB will rework the problem for you with answer `ans=8`. If you take the tour as recommended above, then you will soon learn how MATLAB can save intermediate results for use in further calculations. This makes MATLAB extremely good for working on problems **incrementally**. You do a little bit, debug and polish it, and then do a little bit more until you are done.

### Recording your MATLAB computations

As powerful as it is, there is a difficulty with the interactive MATLAB approach. The difficulty is that it is not so obvious how to save both your commands and your results so that they can be printed and examined by someone else, such as a grader or your instructor. We will now examine a clever method for preserving the powerful interactive, incremental development process, AND also for getting both your instructions and your results down cleanly into a text file from which it can be copied into a Word or RTF document and reviewed.

We are going to go through the process step-by-step of how you can do your MATLAB homework interactively and iteratively and still produce a text file of your work. We are going to have MATLAB create a text file of your work on this assignment. Please work through this step by step.

1. Bring up MATLAB. First we’ll unclutter the screen a little. Click on “Desktop”. Ensure that “Command Window” and “Current Directory” are checked. Now uncheck “Command

History” and “Workspace”. As a matter of taste, you may wish to drag the center bar on the screen to the left, reducing the size of the “Current Directory” window and enlarging the “Command Window”.

2. Change to your MYMATLAB directory. It contains a file called `TEMPLATE.m`. `TEMPLATE.m` is a **script** or “dot m” file. All script files are text files. They contain sequences of MATLAB commands. Script files may be edited or executed.

3. Click on `TEMPLATE.m`. Clicking on a script file brings up the contents of the file in a separate MATLAB “Editor” window. You may edit these script files In the “Editor” window. Move down to the line in the file that says

```
delete TEMPLATE.txt; diary TEMPLATE.txt.
```

Edit that line to

```
delete tw0101.txt; diary tw0101.txt.
```

Notice that “tw0101” is just the assignment handle of this assignment.

4. Next move down to the line that says

```
Assignment handle here
```

and edit that line to say

```
Assignment handle tw0101.
```

5. Next move down to the line that says

```
Student name or Study Group name
```

and edit that line to say

```
Student name: <your name>
```

where <your name> is your actual name.

6. Click on File-->Save As, type in the file name as `tw0101`. Click on “Save”.

A check of the “Current Directory” window shows that you have a new script file `tw0101.m` for this assignment.

7. Now we will **execute** the script file `tw0101.m` and see what happens. In order to execute `tw0101.m`, you may do one of two things. You may click on the button in the toolbar that says either “Run” or “Save and run”, or on a PC, you can press the “F5” key which does the same thing. Execute the `tw0101.m` script.

A check of the “Current Directory” window shows that you have two new files `tw0101.m`, `tw0101.txt` for this assignment.

8. In summary, always save the `TEMPLATE.m` file unedited. For MATLAB problems, use it to create new script files for specific assignments just as we have done here. If you follow the pattern here, then for each MATLAB assignment, you will produce two files. One is a script file and the other is a text file. Both files have the name of the assignment.

### The MATLAB development cycle

This section will prepare you to be able to complete the MATLAB portion of this assignment **tw0101** as detailed in Exercise #1 below. The *purpose* of walking through this exercise is to teach you the MATLAB development cycle that you will use for each computational assignment in the course. We’ll be summarizing how this works for this assignment **tw0101**. This builds on the preparation work that you have done just above, and you should refer there if necessary. After you have completed this assignment, you should be ready to do any MATLAB assignment in the course.

1. Generally you will need to have three windows open on your MATLAB desktop. These are the “Current Directory” window and the “Command Window” (usually together), and the “Editor” window (usually floating separately from the other two). If you have completed the previous steps correctly, you should be in your `MYMATLAB` directory, which should contain a script file `tw0101.m` and a text file `tw0101.txt`. The script file `tw0101.m` should still be loaded into your “Editor” window.

**Remark.** It is worth noting one nice tip here. Assuming that the script `tw0101.m` is already in the “Editor” window, if you click on the file `tw0101.txt` in the “Current Directory” window, then *both* files will be loaded into the “Editor” window. You can switch to view one or the other by clicking on one of the two tiny tabs that are the bottom left of the “Editor” window. You do not want to edit the file `tw0101.txt` because it gets deleted and remade every time you save and run the script file `tw0101.m`. However, the software is smart and it updates the file `tw0101.txt` in the “Editor” window whenever you recompile the script `tw0101.m`. If you want to see how the file `tw0101.txt` looks after when you recompile the script `tw0101.m`, then all you have to do is to view it in the “Editor” window. This is useful for example if you are tweaking the look and feel of your file `tw0101.txt` before turning it in.

2. The optimum MATLAB development cycle takes advantage of the interactive and iterative features of the language. The general idea is that you enter a little bit, run it, edit it if you

need to correct something, run it again, rinse and repeat, until you have that little bit perfect. Then you leave that alone and move on to the next little bit, until step by step you have written, edited, debugged, and published your entire assignment.

Here is precisely how you will do that. You will **enter** and **execute** all your MATLAB commands inside the “Editor” window within the script file `tw0101.m` as described above. After you have entered some MATLAB commands and executed the script, the **results** of your of your work so far will appear in the “Command Window” for you to check. At the same time, and this is the key fact, all of your results to that point will be **recorded** in the script file `tw0101.txt`. Each time that you add or change something in the script file `tw0101.m` and then execute the script, the “Command Window” will be completely cleared and then redone with your new work. Likewise the results file `tw0101.txt` will be invisibly deleted and immediately completely rebuilt to reflect your current progress. When you are all done, you will turn in for grading only the final version of the text file `tw0101.txt`, because that file contains the complete record of your work.

3. Exercise #1 asks that you work through and turn in every example in §C.1 (Getting Started with MATLAB) in the text, pp. 528-533. Let’s see how this will work. At this minute, your `tw0101.m` file looks something like this:

```
%  
% Standard template for Math 470 MATLAB Homework  
%  
echo on  
delete tw0101.txt; diary tw0101.txt  
clear; clc; close all;  
format rat  
%  
% Assignment handle: tw0101  
% Student name: Jon Pitts  
%  
%  
% Computations go here.  
%  
%  
echo off; diary off
```

4. Turn to p. 528 in the text. We are going to go through the first example which is

```
>> 2^7 + 125/5
```

First, in order to let the reader keep track, we are now working on p. 528 and we need to record that into our script. After we record the page number and the instruction, our script file `tw0101.m` should look like this:

```
%  
% Standard template for Math 470 MATLAB Homework  
%  
echo on  
delete tw0101.txt; diary tw0101.txt  
clear; clc; close all;  
format rat  
%  
% Assignment handle: tw0101  
% Student name: Jon Pitts  
%  
%  
% Computations go here.  
%  
% p. 528  
%  
2^7+125/5  
%  
%  
echo off; diary off
```

5. When you run this script, your “Command Window” will look something like this.

```
format rat  
%  
% Assignment handle: tw0101  
% Student name: Jon Pitts  
%  
%  
% Computations go here.  
%  
% p. 528
```

```
%  
2^7+125/5
```

```
ans =
```

```
153
```

```
%  
%  
%  
echo off; diary off
```

If you had made a mistake, then you could correct it now in the “Editor” window and rerun the script. You can tell by inspection that you’ve done this example correctly

6. The “Command Window” is not a permanent record of your work. The permanent record has been saved in the text file `tw0101.txt`. Click on `tw0101.txt` in the “Editor” window and what you see should look something like this:

```
clear; clc; close all;  
format rat  
%  
% Assignment handle: tw0101  
% Student name: Jon Pitts  
%  
%  
% Computations go here.  
%  
% p. 528  
%  
2^7+125/5  
  
ans =  
  
153  
  
%  
%  
%
```

```
echo off; diary off
```

7. You are ready to move on to the next example which is at the top of p. 528. The example is
- ```
>> 11*( (128/(9+7) - 2^(72/12)))
```

You enter the new page number and this command into your script file `tw0101.m` inside the “Editor” window. The result should look something like this:

```
%  
% Standard template for Math 470 MATLAB Homework  
%  
echo on  
delete tw0101.txt; diary tw0101.txt  
clear; clc; close all;  
format rat  
%  
% Assignment handle: tw0101  
% Student name: Jon Pitts  
%  
%  
% Computations go here.  
%  
% p. 528  
%  
2^7+125/5  
%  
%  
% p. 528  
%  
11*( (128/(9+7)-2^(72/12)) )  
%  
%  
echo off; diary off
```

8. You can edit and run this example until you have done it correctly. Every time you edit and rerun it, you recreate your permanent record in text file `tw0101.txt`.

9. Now you will continue in this manner through all the examples up through §C.1, p. 533.

## Exercises

1. (20 points) Read Appendix C in the text through §C.1 (Getting Started with MATLAB), pp. 527-533. You are to prepare a script file **tw0101.m** and a text file **tw0101.txt** in which you work every example in §C.1, pp. 528-533. You can tell the examples in the text by watching for the `>>` symbol which instructs you to enter MATLAB commands. Print out and turn in your text file **tw0101.txt**.