Name: Andrew Brown	**Lab Time:** T 12:00
People Worked With: n/a	**Websites Used: Stack Overflow**
Time spent on zyBooks (hrs): 2	**Time spent on lab (hrs): 1.5**
Submission Instructions	
Turn all work in to Lab 1 on Gradescope (PDF) and Canvas (.zip file), even if it is not complete yet. If you are not finished, complete the assignment outside of lab and re-submit to Lab 1 on Gradescope and Canvas. All labs are typically due at the same time on Monday every week, but check Canvas if in doubt.	

BEFORE YOU BEGIN

1) **Please read "ENGR112 Assignment Instructions.pdf" in the Canvas module "Start Here."**
2) Set up a file directory according to the instructions.
3) ECampus
    a. Download (and read) the Lab1GettingStarted.m script
    b. Watch the Lab0 video

General note about labs

The labs are designed to introduce basic concepts to prepare you for the homework. *When you get stuck, ask for help either from the TAs or from your neighbors.*

Lab assignment instructions

Labs should be completed as follows:

- Read and complete zyBooks chapters and participation exercises **before lab**.
- Do the open-ended problems with step-by-step instructions listed here
    - Submit PDF to Gradescope, .zip to Canvas when completed. Labs will be graded separately from the homeworks.
- [Extra credit/optional] Do the zyBooks challenge listed at the end of the document
    - These are graded for completion only

Learning objectives:
You should be able to answer the following questions

- What is a script file?
- What is the difference between a script file and the command window?
- How do you re-write a word problem as a script?
- How do you create variables?
- How can you examine the values of variables?
- How do you use the debugger?

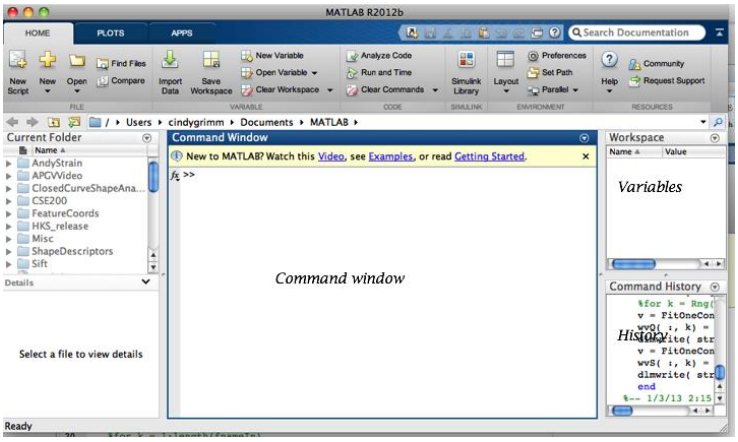
New MATLAB commands
These are highlighted in **bold** in the instructions below.

- `clear` – clear the window
- `clc` – clear the command window
- `clf` – clear the figure
- `fprintf()` – formatted print
- `ceil()` – round up to the nearest integer
- `disp( variable name )` – show the value(s) of the variable
- `plot(x,y)` – plot x versus y
- `linspace( firstValue, lastValue, numElements )` – make an array
- `hold on` – don't erase the current plot before drawing again
- `start:space:end or start:end` - colon operator

Lab Problems

Problem 0

First, familiarize yourself with the MATLAB environment. Open up the program. The main window will be some version of this (it might not automatically open all of the windows shown):



- Note: All of the window panes can be dragged out of the main window to be separate windows, then put back in again, rearranged and re-sized.

Problem 1

Create a variable and assign it values in a script. Print out values from the script. Equation to calculate: cosine and sine of a degree value input by the user.

Deliverables:

1) Turn in a script that does the following
    i) Uses input to get the degree from the user
    ii) Calculates cosine of that degree
    iii) Calculates sine of that degree
    iv) Prints out the answer
    v) Includes comments
2) Copy the command window output

General Guidelines

We're going to try this in the command window first, then copy it into the script

---

Step-by-Step Instructions:

1.a) Do the following in the main command window. This is just practice.

- Create a variable **t** by typing `t = 30`
    - Notice it shows up in the variable window
    - Notice how it printed out the value since you didn't include the semi-colon
    - Notice that the command shows up in the history window
- Now create a variable **x** which is cosine of **t** (assuming t is in degrees)
    - If x is not 0.8660, remember that cosd uses degrees and cos uses radians…
- Create another variable, **y**, which is sine of **t**. This time, suppress the output by adding a semi colon.
    - Check that you have **t**, **x**, and **y** in the variable window
    - Print out the value of **y** in the command window by typing **y** (no semi-colon)
- Change **t** to be 130
    - Did the value of **x** or **y** change? Can you explain why not?
- Use the up arrow key (or click in the command history window) to re-calculate **x** and **y**
    - Did the values of **x** and **y** change to -0.6428 and 0.7660?
- Now write a print statement (fprintf) that prints out the value of **t** (with units) to 2 decimal places

    - `fprintf('t is %0.2f\n', t);`
- Change the print statement in the following ways
    - Print out 6 decimal places
    - Print out **x** as well as **t** – you will need to add a %0.2f for the **x** and add **x** after **t**
- Type `clc` – what happens?
- Type `clear` – what happens? What happens if you type t? or x? or y?

1.b) Do the following in the editor window - we're going to save what you just did in a script

- Create a new script (button in the upper left)
- Add a comment at the top about what the script does
    - `% This is who I am`
    - `% This is my comment`
- Add clear and clc at the top, with a comment about what they do
- Now write three lines of code – set **t** to be 30, then calculate **x** and **y** from **t** as before
    - Use semi-colon (;)to suppress the output
- Run the script (big green arrow at the top)
    - You'll be asked to save the script – save it with a **useful** name like Lab1_Problem1.m, using the directory structure given in the assignment instructions
    - You may be prompted to either change the folder or add path -> always do change folder
- Go to the command window and type **x**. Did it work?
- Now add an `fprintf` statement to the script to print out t, x, and y.
- Run the script again and check that your print statement printed the correct thing to the command window.
- One last thing – instead of setting t, use input
    - `t = input('Value for t in degrees:');`
- Run the script again – this time you'll be taken to the command window and prompted for a value for **t**. Enter 130.
    - Did it print out the correct values? If so, you're done – copy the script and the command window output below
- If it didn't work, ask your neighbor for help…

Self-check:
For t = 130
`x is -0.6XXXX8, y is 0.7XXXX4`

Relevant zyBooks sections

- MATLAB/Interpreter, Comments, Scripts
- Variables and assignments
- Numeric expressions

- Internal mathematical functionsBasic input: The input() function
- Floating-point formatting in fprintf
- Repeated commands and the up-arrow key

Answer script here:

[20 pts] Appropriately commented, including units
[20 pts] Used input
[20 pts] Correct equations
[30 pts] Correct fprintf

```
%Lab 1 Andrew Brown

clc %This command clears the command window
clear %This command clears all variables and functions from the
workspace.

%I am practicing defining variables and using the "input" and "fprintf"
%functions.

% Practice defining variables.
t=input('Value for t in degrees:'); %Practice using the "input"
function.
x=cosd(t);
y=sind(t);

%Practice using the print function.
fprintf('t is %0.0f deg.\nx is %0.6f. \ny is %0.6f. \n', t, x, y)
```

Command Window Output

[10 pts] for nice formatting of fprintf, using a sentence for the output

```
Value for t in degrees:130
t is 130 deg.
x is -0.642788.
y is 0.766044.
```

Problem 2

Write a script to calculate the following equations. Demonstrate that it works for three different values of x by changing the value of x in the script then running it. Assume x is in radians (Set x to: 10.2, 12.8, and 0.1). Hint: the Matlab command for e^x is `exp(x)`.

$$y = \frac{\sqrt{2x}(4x^3)}{4x + 7^{x/10}}$$

$$z = \log_{10}(2x + 5) + \frac{4x + e^x}{2/3 + 4x^2}$$

Common gotchas

1) Forgetting the * between 4 and x
2) Forgetting to put the denominator in parenthesis ()
3) Forgetting to put the numerator in parenthesis ()
4) Forgetting to put the exponent (x/10) in parentheses
5) Using the correct log

Deliverables:

1) Turn in a script that does the following
   i) Calculates y and z from x
   ii) Prints out the answer (include value of all three variables in your fprintf statement)
   iii) Includes comments
2) Copy the command window output three times, once for each value of x given above

General Guidelines

Much like the previous problem, only the equations are a bit more complicated

---

Step-by-Step Instructions:

2.a) Create a new script

- Put your comments at the top, along with clear and clc
- Set the value of x
- Write the equation for y and the one for z
- Write the fprintf statement (print y and z to 6 decimal places)

2.b) Run and debug the script

- Run the script (big green arrow)
  - Again, save it with a useful name
- If you got it right, it will just print out the answer
- You may get red text if the MATLAB syntax is incorrect
  - Read the text – if you click on the line number it will take you to that error in the script
  - If you don't understand the error, try cutting and pasting into Google
  - If you still can't figure it out, grab a TA or a neighbor
- If it ran but it printed out the wrong number… (see self-check below)
  - Look for the 2 gotchas listed above
  - Double check the equation – did you copy it correctly?
  - Check that you're using the radians version of sin, not sind
  - Grab a TA or a neighbor

2.c) Run your script 3 times with the 3 values of x, copying and pasting the answer in below

Relevant zyBooks sections – same as above

- Variables and assignments
- Numeric expressions
- Math functions
- Basic output I

Self-check:

```
For x= 10.200000, y is XX8.XXX8XX, z is XX.045578
For x= 12.800000, y is X7X.XXX970, z is XXX.702069
For x= 0.100000, y is X.XXX260, z is X.845962
```

Note: This self-check shows answers printed to 6 decimal places. You should use %0.06f in an fprintf statement to print to 6 decimal places.

Grading Criteria:
[30 pts] Script has comments
[20 pts] Correct equation
[20 pts] fprintf correctly formatted, outputs values for all 3 variables
[30 pts] 10 pts each for each x value shown in command window output

Answer script here:

```
%Lab 1 Andrew Brown

clc
clear

%I am practicing simple mathmatical calculations and functions.
```

```
%Practice defining variables using mathmatical functions and syntax.
x=10.2; %Units are radians
y=((sqrt(2*x)*(4*x^3)))/((4*x)+(7^(x/10)));
z=log10((2*x)+5)+(((4*x)+exp(x))/((2/3)+(4*x^2)));

%Practice using the "fprintf" function.
fprintf('For x = %0.6f radians, y is %0.6f, and z is %0.6f.\n', x, y, z)
```

Command window output

```
For x = 10.200000 radians, y is 398.777844, and z is 66.045578.
For x = 12.800000 radians, y is 670.823970, and z is 553.702069.
For x = 0.100000 radians, y is 0.001260, and z is 2.845962.
```

Problem 3

Create a script that plots a circle.

Note: This is a bit of a look ahead to next week…

Deliverables:

1) Turn in a script that does the following
   i) Calculates an entire array of t values
   ii) Calculates the cosine and sine of those t values
   iii) Plots the results
2) Copy the resulting figure

General Guidelines

Much like the first problem, only for arrays instead of

Step by Step Instructions:

3.a) Practicing making arrays in the command window

- In the command window, type `ts = linspace(0, 2 * pi, 30);`
  - I use **ts** because this makes multiple "t's"! 30, to be precise, from 0 to 2 pi
  - Look in the variable window – what do you see? You can click on the variable and see all of the values in a spread sheet.
  - Type `disp(ts)` in the command window – this command shows all the values without the "ts =" at the top
- Still in the command window, make the cosine and sine values:
  - xs = cos(ts);
  - ys = …
- Notice that the "right" thing happened – for each t value, you get one x value out that is the cosine of t.
- Pretty picture:
  - Type `plot(ts,xs);`
  - Now type `plot(ts,ys)`
    - What happened?
  - Now type
    - ```
      plot( ts, xs, '-r')
      hold on
      plot(ts, ys, '-g')
      ```
  - Explain to yourself what hold on and the '-r' did.
    - Note that those are single ticks (') are the character under the " key. Microsoft word converts them to some weird character, so cutting and pasting doesn't work
    - If you got the correct ' mark, the text inside will turn purple
      ```

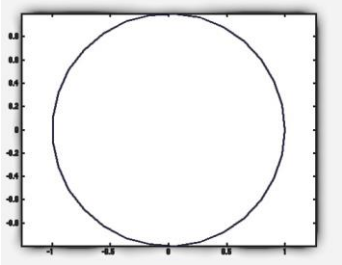
```
>> plot(xs,ys, '-r')
```

- One last thing
  - `clf` – what happens?
  - `plot(xs,ys)` – what do you get?
    - Note, if it looks squished, type `axis equal`

3.b) Move it to a script

- Two ways to do this; start from scratch or copy your script from problem 1 and just change t, add the plot, and take out the fprintf
- See the homework instructions on how to either save the figure as an image or do a screen grab and paste it in.

Self-check:



Grading Criteria:
[20 pts] Commenting
[20 pts] Creating t values
[20 pts] Calculating x and y
[20 pts] Plotting
[20 pts] Copying plot into the box below

Answer script here:

```
%Lab 1 Andrew Brown

clc
clear

%I am practicing using arrays, calculating sines and cosines, and ploting.

%Practice making an array calculation and establishing variables.
ts=linspace(0, 2*pi, 30);
xs=cos(ts);
ys=sin(ts);

%Practice plotting on one graph and changing plot colors.
plot(ts, xs, '-r')
hold on %adds previous plot to next plot
plot(ts, ys, '-g')

clf %Clear current figure
plot(xs, ys)
axis equal %makes the x and y axis equal to e/o.
```

Plot here: