# EGR 111
Introduction to MATLAB

This lab introduces the MATLAB help facility, shows how MATLAB[TM] , which stands for MATrix LABoratory, can be used as an advanced calculator. This lab also introduces the assignment statement (=), parentheses () to change the order of operations, and the semi-colon (;) to suppress printing.

New MATLAB commands: help, lookfor, assignment operator =, semicolon, and parentheses ( )

1. Getting Started

MATLAB is available in the computer classrooms and open lab in Shiley Hall (or from anywhere using VDI). Start up MATLAB. When MATLAB is ready, the prompt ">>" will appear in the Command Window.
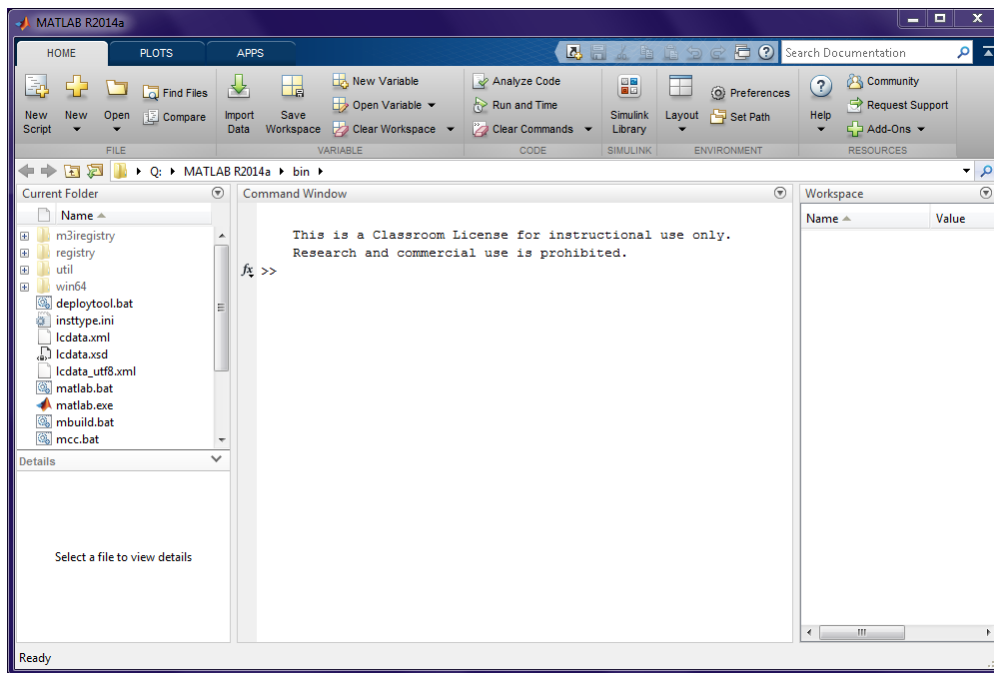
2. Getting Help

To launch the MATLAB help facility, click on the Help button (it looks like a question mark). A window should open that contains documentation for MATLAB. You can find information about a particular MATLAB command or topic by using the search box. There are also tutorials available if you click on the link "Getting Started with MATLAB".

If you know the name of the function or operator for which you want more information, you can go to MATLAB's command window and type the word `help` followed by the function name. For example, type the following command into MATLAB's command window and then press the Enter key: `help plot`

3. Calculations

MATLAB can do simple calculations, similar to your calculator. Type the following commands (each line followed by the Enter key):

```
4*5
3^2
```

In the commands above, the variable `ans` stores the result of the most recent calculation. This feature is similar to Texas Instruments calculators.

We can also store values into variables other than `ans`, so we can use the values in a later statement. Try typing the following:

```
x = 4*5
y = 3^2
```

The two commands above are examples of assignment commands. MATLAB evaluates the expression on the right hand side and stores the result in the variable name on the left hand side.

Often, we want MATLAB to compute a value, but we do not want the result to be printed to the screen. In order to suppress printing the result, we place a semicolon at the end of the line. Try typing the following:

```
z = 2*2;
```

Notice that nothing is printed when the line ends in a semicolon, but MATLAB does execute the command.

To see the value stored in a variable, just type the variable name as follows:

```
z
```

We can also use the variables in expressions. Try typing the following command:

```
a = x + y
```

In MATLAB, anywhere you can put a constant number (like 4), you can also put a variable (like x) or even an expression (like 3*x+4).

In MATLAB, several commands can be placed on a single line. If you separate the commands by a comma, the results of each command are printed to the Command Window. If you separate the commands by a semicolon, the commands are executed but nothing is printed to the Command Window. For example, in the command below, the values of the variables `a` and `c` are computed and stored, but they not printed because they are followed by a semicolon.

```
>> a = 1; b = 2*4, c = 5+8;
b =
     8
```

In MATLAB, you can recall previous commands by pressing the up arrow on the keyboard. So if you wanted to run the above command again, you could just hit the up arrow, edit the command if needed, and then Enter. If you type the first few characters of a command and then hit the up arrow, MATLAB recalls the most recent command that starts with those characters. You can also click on a previous command in the command history window and drag it to the Command Window. Try these features out to see how they can save a lot of typing.

4. Rules for variable names: In MATLAB, variable names must begin with a letter and can contain letters, digits, and underscores (but not spaces or operators like -, +, *, /). Variable names are case sensitive, so x is <u>not</u> the same variable as X. Examples of valid variable names are x, x2, and number_of_wheels. Examples of variable names that are NOT valid are 2x, yes!, and x*2.

Which of the following are valid variable names?

      size
      numberOfPeople
      size_of_table
      2b_or_not_2b
      rows&columns

Check your answers by attempting to assign a value to each of the above names. For example, type "size = 4" into MATLAB's command window and see if it generates an error.

5. Order of Operations

The order in which operations are computed are determined by the following table.

Order	Operator	Symbol
1	Parentheses	( )
2	Power	^
3	Unary Plus and Minus	+ and -
4	Multiplication and Division	* and /
5	Addition and Subtraction	+ and -

Parentheses are computed first, followed by the other operators in the order shown in the table. So in the expression 4*(3+2), the value of (3+2) is computed first, then the result is multiplied by 4. In the expression 3*2^2, the power operator (^) has higher precedence than multiply (*), so 2^2 is computed first, and the result is then multiplied by 3. So 3*2^2 = 3*(2^2) = 3*(4) = 12.

Operators on the same level of precedence are computed left to right. So 6/3*2 = (6/3)*2 = (2)*2 = 4 because division and multiplication are on the same level.

Unary minus refers to the case when a minus sign operates on a single value (as opposed to subtracting two values). For example, in the expression -2^2, the power has a higher precedence than the unary minus operator, so -2^2 = -(2^2) = -4.

Determine the value of the following expressions and check your answers with MATLAB.

2*1+3 = _____

10/5*2 = _____

4/2+3 = _____

-2^2+1 = _____

4+2/2+3 = _____

2^2*3 = _____

-2*3^2+1 = _____

6. The MATLAB Workspace

Notice the "Workspace" in the upper right corner of the MATLAB window. Are the names inside of the Workspace familiar? The Workspace lists all variables that you have created and which are currently being stored in the computer's memory. Within MATLAB, you retrieve these stored value by using their variable names inside of the Command Window.

7. Built-in Functions

MATLAB has a huge number of built-in functions that can compute almost any common mathematic function. For example, type the following to use the square root and cosine functions:

```
d = sqrt(9)
e = cos(0)
```

Exercise 1: Let two points p1 and p2 be defined by the x and y coordinates as p1 = (x1, y1) and p2 = (x2, y2). The distance between two points p1 and p2 is given by $d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$. Write the MATLAB commands to define the values of variables x1, y1, x2, y2 for the points p1 = (-1,3) and p2 = (2,-1), and compute the distance between p1 and p2 (see figure below).
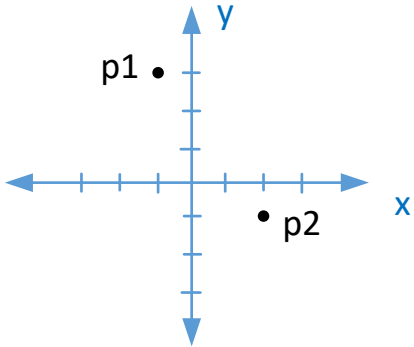
__Checkpoint 1:__ Show the instructor your commands and the results for Exercise 1.