Expt No.1 Date: 24th Jan 2021

# Introduction to MATLAB

**Objectives**

1. Familiarization with the MATLAB computing environment.
2. To be able to write scripts in MATLAB.

You are expected to do whatever is written in bold i*talics.* Note down all your observations in the space provided.

MATLAB stands for MATrix LABoratory. How to start MATLAB?

* Click on the MATLAB icon
* Go to start → Programs → MATLAB 6.1 → MATLAB 6.1

Either of these would open the MATLAB command window which is shown below:

Start MATLAB by doing either of the above.

Notice that there are three windows in the main window; we would focus our attention on the one titled Command Window. This window can be identified as the one with a double arrow >>. This double arrow is the command prompt of MATLAB, i.e. the commands are executed by typing them at the prompt and pressing the enter key. Note that it is an interpreter and you don’t need to compile a MATLAB program.

MATLAB essentially processes vectors and matrices. We will see how to define these.

Typing x =[1 2 3 4 5]

defines x as the row vector [1 2 3 4 5].

*Type x =[1 2 3 4 5] at the command prompt and observe what happens, write down your observation in the space below.*

*Repeat with x =[1 0 0 1]; (note the extra semicolon at the end)*

**

*Type x at the command prompt and note down the result.*

**

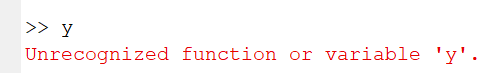
*Can you tell what the effect of the semicolon is?*

* *It is just executing but not printing it under the execution.*

Note: MATLAB is case sensitive; x and X is different, and all commands which are inbuilt should be typed in lowercase.

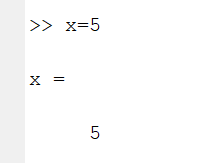
If you try typing a variable to which you haven’t assigned any value an error message will be produced. Try it.

*Type y at the command prompt and note down the error message that you get.*

**

If you want to assign a scalar value to a variable, the procedure is the same, just type x=5

*Type the command and observe the output.*

**

Observe that initially x was defined as vector, to redefine it as a scalar you just need to assign a scalar value to it as we did above.

# HELP

MATLAB help can be obtained in two ways; a very comprehensive graphical interface for help is provided in the menu bar of the MATLAB window. To obtain this, click on the menu item Help, and choose “MATLAB help”. A new window is opened, it is quite user friendly and you would be able to find your way through it.

If you want a particular command, type help command at the prompt, the help for that command will be printed on the command window.

*Use command line help to find out what the following commands do.*

1. help
2. who
3. whos
4. clear
5. clc
6. lookfor

Suppose you want to find the command for finding inverse sine (sin-1(x)). Use the lookfor command (type *lookfor sine* at the command prompt).

Working with vectors and matrices

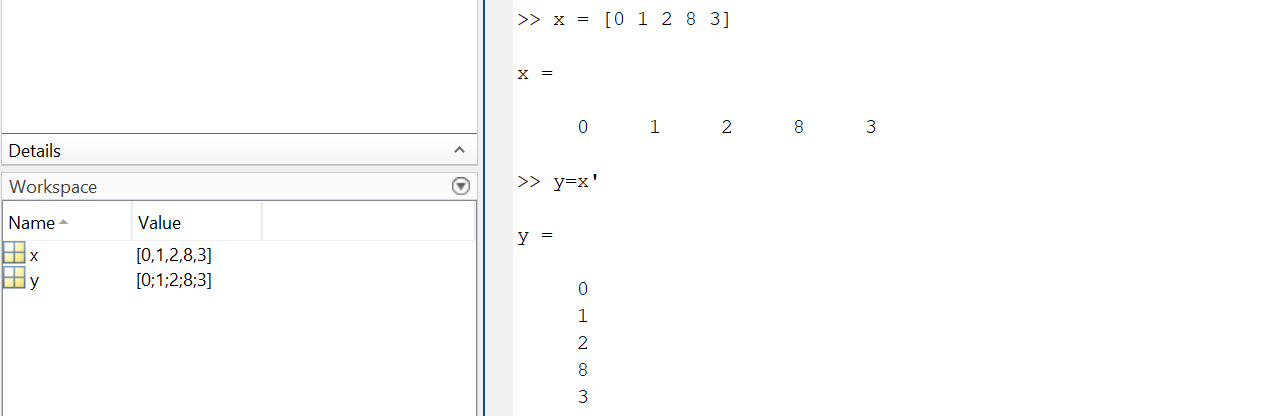
We saw how to define a vector in the previous section.

x = [0 1 2 8 3]

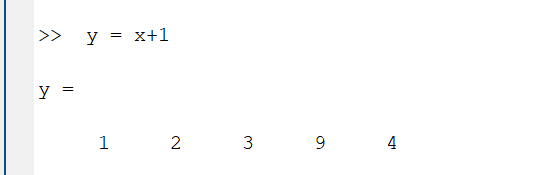
Value of the above vector can be displayed by typing x at the command prompt.

*Check the result of the following:*

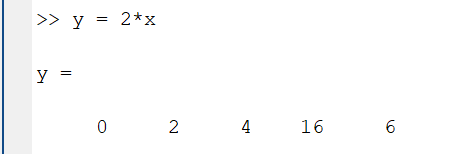
1. y = x’



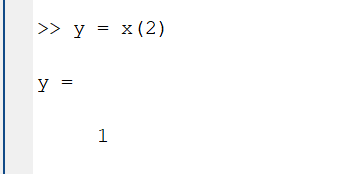
2. y = x+1



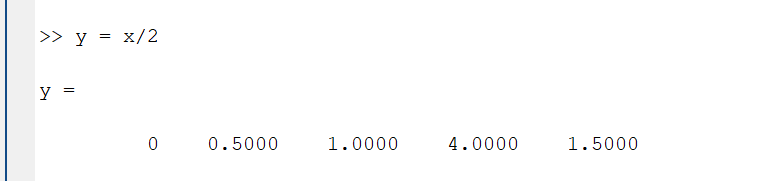
3. y = 2\*x



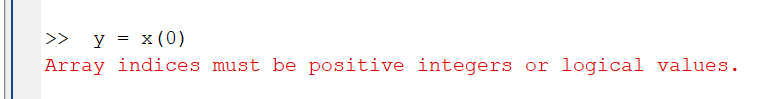
4. y = x(2)



5. y = x/2



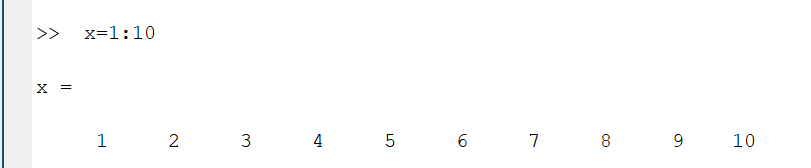
6. y = x(0)



The colon operator (:)

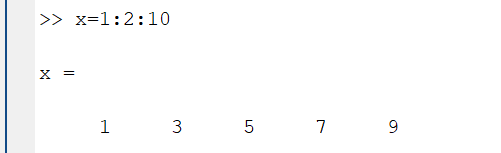
“:” (colon) is a very useful operator in MATLAB. Suppose you want to assign the numbers from 1 to 10 as elements of a vector, instead of typing all the values from 1 to 10, the colon operator can be used.

*Type x=1:10 and observe the result*

**

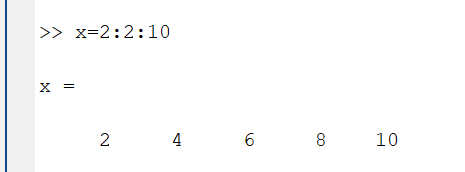
This assigns numbers 1 to 10 as elements of x. Instead of 1 to 10, if you need to assign only the odd numbers from 1 to 10 as elements of x, again colon operator is used.

*Type x=1:2:10 and observe the output*

**

*How would you assign the even numbers from 1 to 10 to a vector?*

*=>We need to choose the first variable 2 and keep the deference 2 upto 10.*

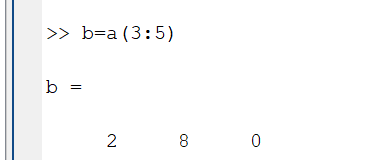
**

The colon operator can also be used for extracting elements from a vector or for modifying selected elements in a vector. The following demonstrates this. Start with a vector

*a=[6 3 2 8 0 2 1]*

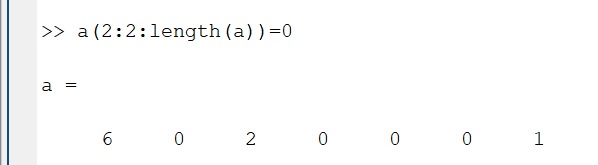
Suppose we want to extract three elements starting from the third element (which is 2) and assign it to another vector.

*b=a(3:5)*

**

If we want to make the numbers at even locations of *a* zero, the following command can be used.

*a(2:2:length(a))=0*

**

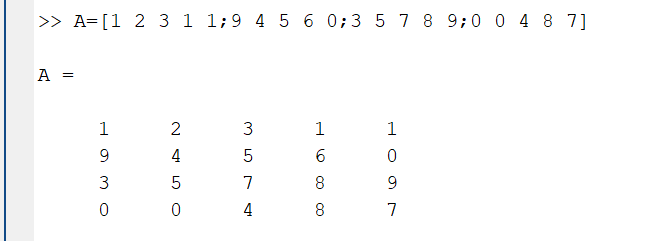
Notice that we have used a new function in the above command - length(a). The length function returns the length (number of elements) of a vector or a string. [Strings are written using single quotes in MATLAB. Eg. s1=’this is a string’ defines s1 as a string with the value given in quotes.]

Matrix

We can define a matrix in MATLAB in the following way.

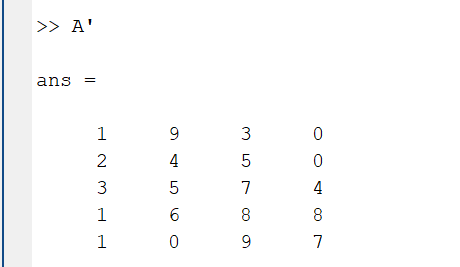
*A=[1 2 3 1 1;9 4 5 6 0;3 5 7 8 9;0 0 4 8 7]*

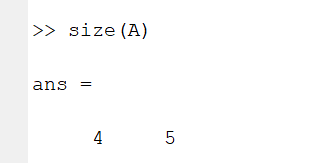
Rows of the matrix are separated by semicolon. *Type the above in the command prompt and see what you get.*

**

*A’* gives transpose of the matrix, and the command *size* gives the number of rows and columns of a matrix.

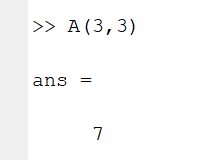
*Find the transpose and size of the above matrix.*

**

**

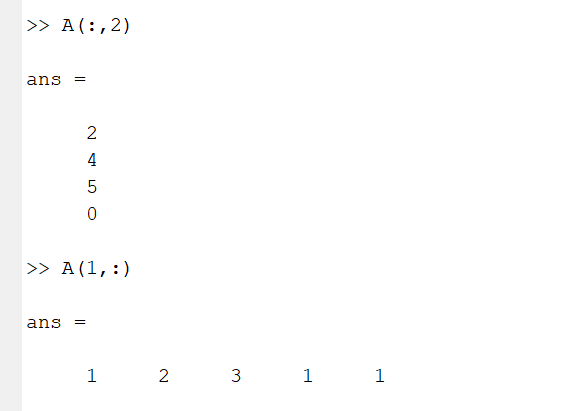
A(m,n) gives the mth row, nth column element of matrix A.

*Find A(3,3)*

**

A(:,n) extracts the nth column and A(m,:) extracts the mth row of the matrix.

*Extract the second column and first row of matrix A.*

**

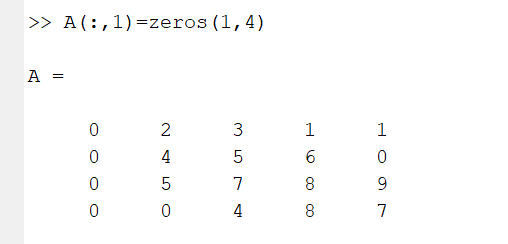
A(i:j,m:n) extracts a smaller matrix from A by extracting the rows from i to j and columns from m to n of A.

*How do you extract the matrix from A?*

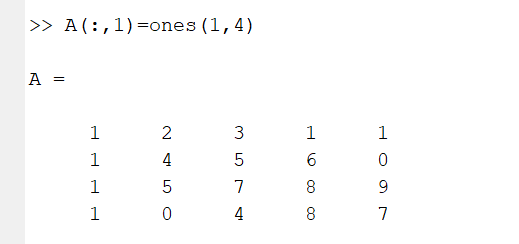
*What does A(i:j,m:n)=0 do?*

*Try out the following two commands, what does each do?*

*1.* A(:,1)=zeros(1,4)



1. A(:,1)=ones(1,4)



Notice the use of two new commands *zeros* and *ones*, use help to find out what they do.

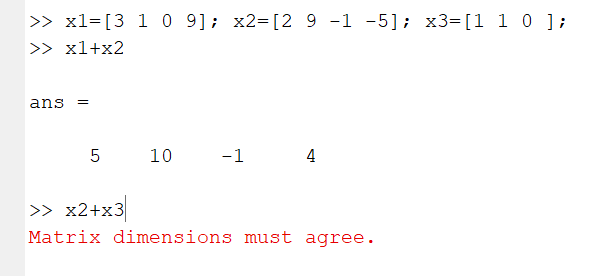
Find out what the following commands does.

1. reshape
2. fliplr
3. flipud
4. inv(A)
5. eye(n)

*Arithmetic operations with vectors*

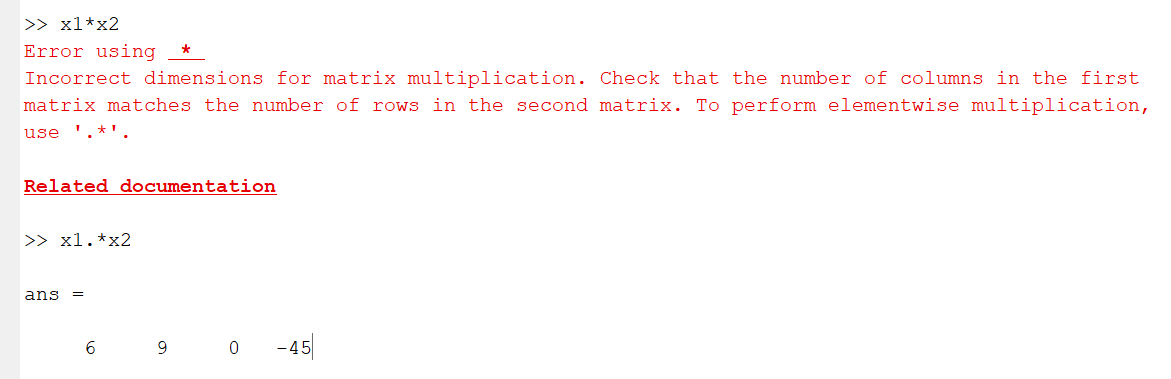
Let us see how to do addition, multiplication etc. with vectors. x1=[3 1 0 9]; x2=[2 9 -1 -5]; x3=[1 1 0 ];

*Find x1+x2 and x2+x3.*

**

From the result, observe that two vectors should be of same length so that you can add them.

*Try doing x1\*x2.Can you explain the result that you get?*

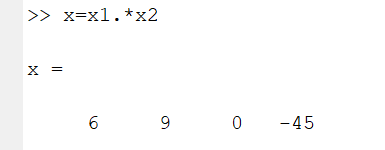
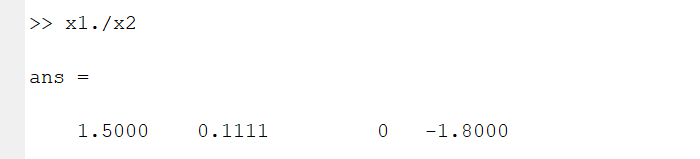
**

Suppose we want to do point wise multiplication of x1 and x2,i.e. first element of x1 should be multiplied with first element of x2 and so on. For this, we can use the following command.

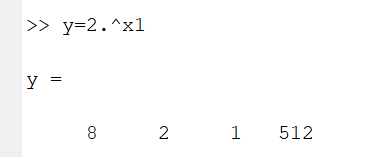
*x=x1.\*x2*

Similarly point wise division can be done by x1./x2.

*Find x1.\*x2 and x1./x2 for the vectors given above.*

** 

*Find the result for y=2.^x1, explain what this does.*

**

Commands for creating plots

You can generate 2-D and 3-D plots using MATLAB. Suppose you want to plot the equation

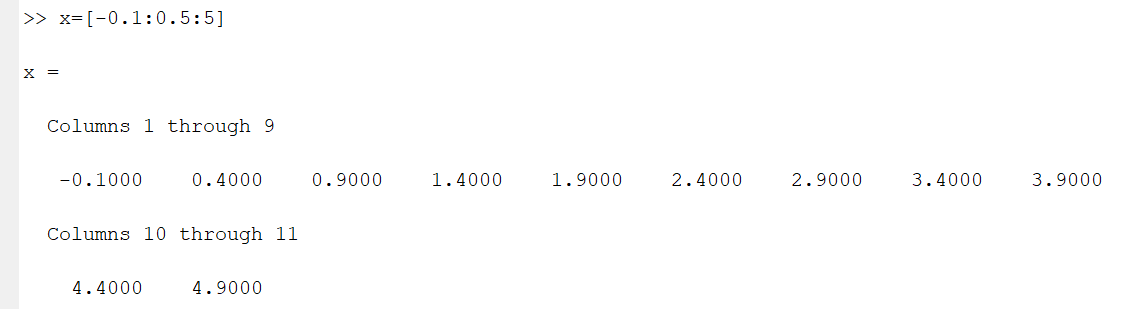
y=2\*x+5

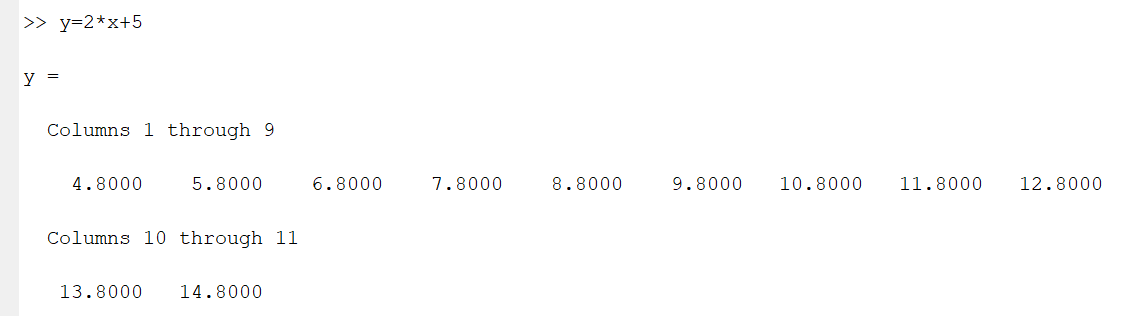
One way to do is create a set of (x,y) values and use the MATLAB function *plot* for plotting these points. This is done as follows.

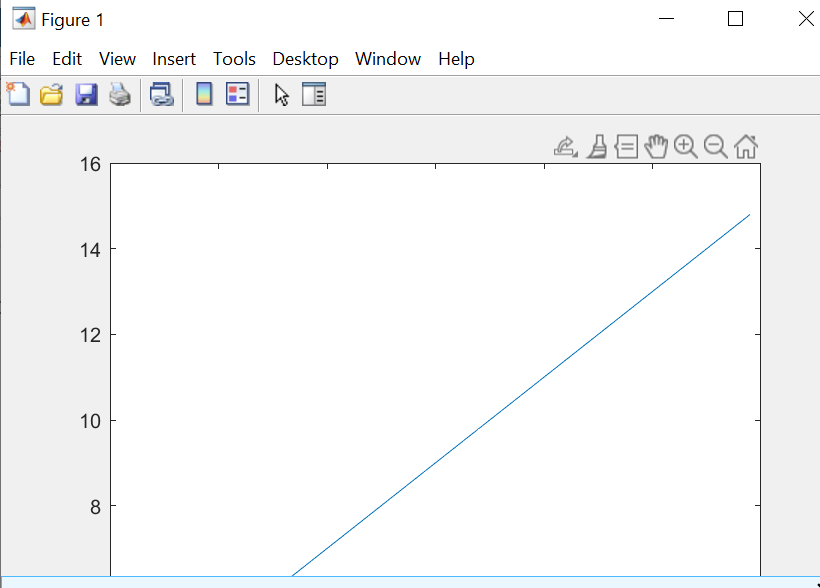
x=[-0.1:0.5:5]

y=2\*x+5 plot(x,y)

*Do the above and observe the plot. Plot the graph in the space below.*

**

**

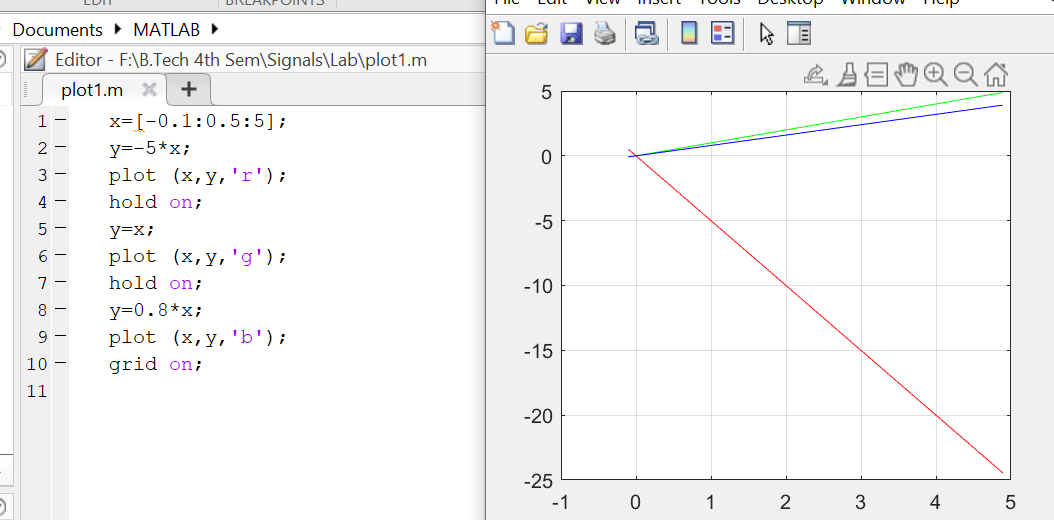
**

*Use help to find out what the following commands do.*

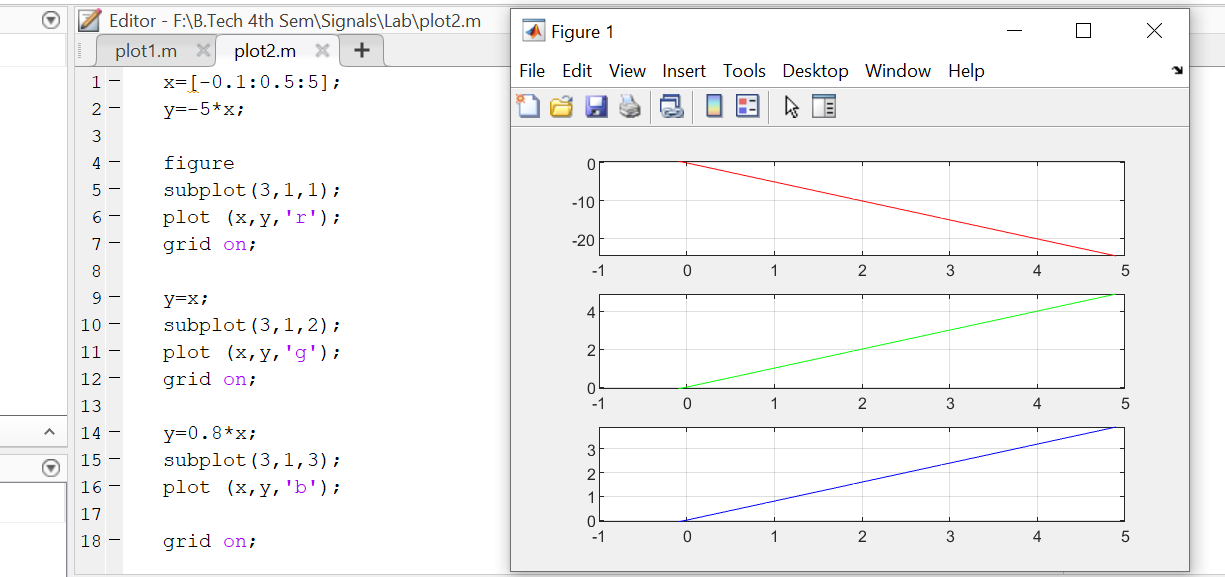
1. Figure
2. Grid
3. Title

1. xlabel
2. ylabel
3. clf
4. hold
5. subplot

*Use the* hold *command to plot y=-5\*x, y=x and y=0.8\*x in the same graph, use different colors for the three lines.*

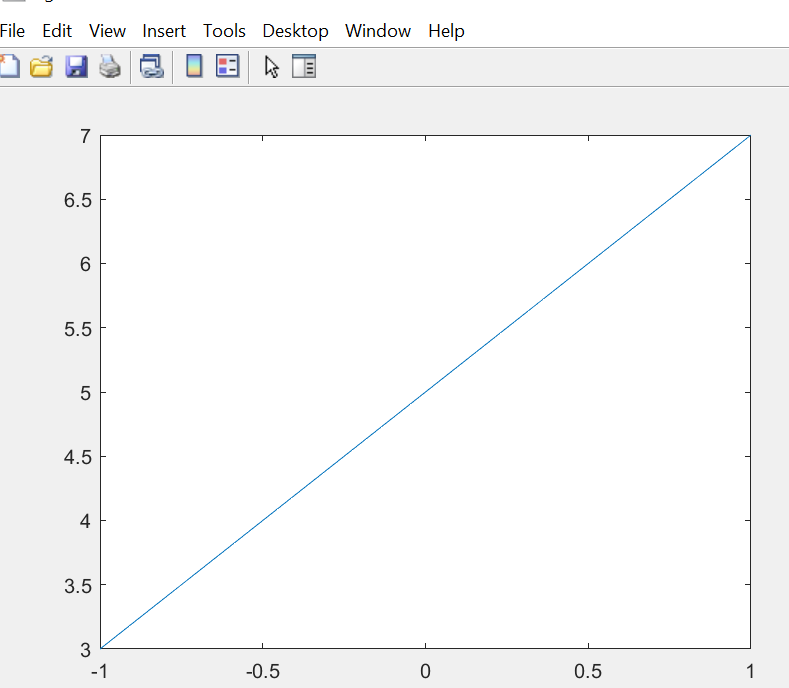
**

*Use the* subplot *command to plot the above three equations on three smaller plots in a single figure window.*

**

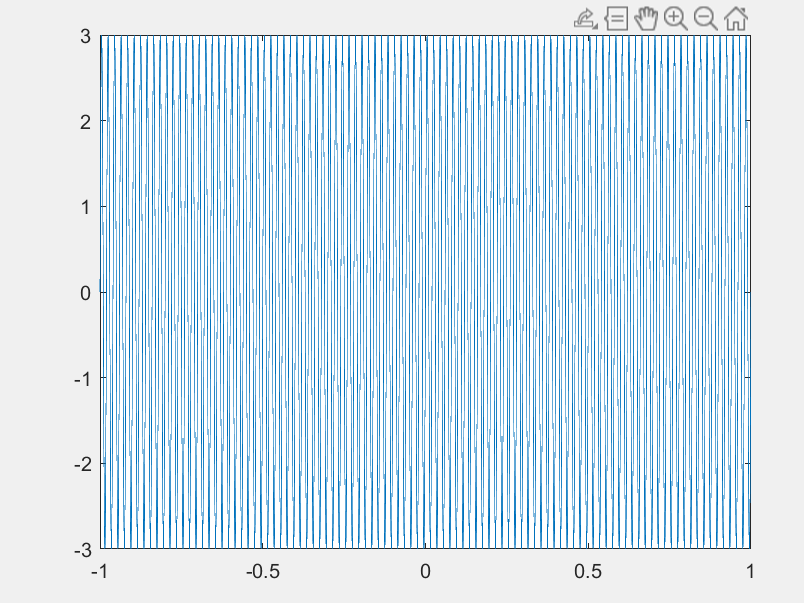
A second way o f plotting y=2\*x+5 is by using the *fplot* command. Use help to find how fplot works. Using fplot, this line is drawn as

fplot(‘2\*x+5’,[-1,1])



Use plot and fplot to plot the function y=3 sin(100\*pi\*t). Note that *pi* is a constant in MATLAB, by typing *pi* you get the value of π.





Programming

Similar to other programming languages, MATLAB has also programming constructs like if, while, for etc. Use MATLAB help to find how these are used.

Input can be taken from keyboard, by using the function input. Use help to find out how input works.

Script

Till now you have been typing the commands at the prompt. Instead, all the commands can be written in a file and saved under a file name with extension .m; if the file name is xy.m, typing xy at the command prompt will execute the commands written in xy.m. For example

*x=input(‘Enter an integer: ‘) y=factorial(x)*

save these two commands in a file named myfactorial.m and type myfactorial at command prompt. Observe the result

**Result**

