

NAME	MUHAMMAD ROMAN KHAN
ID	16242
SUBJECT	COMPUTER PROGRAMMING
SEMESTER	02
EXAME	FINAL TERM
DATE	29-06-2020
SUBIMMITTED	ENGR ASHRIF ALI

Q1 (A) what are the Basic Plots and Graphs of MATLAB?

(ANS)

The Basic Plots and Graphs of MATLAB

Following table describes basic plots and graphs.

box - Axis border

errorbar - Plots error bars along curve

hold - Retains current graph while adding new graphs

line - Creates line object

LineStyle (Line Specification) - Syntax of Line Specification String

loglog - Log to log scale plot

plot - 2-D line plot

plot3 - 3-D line plot

plotyy - 2-D line plots with y-axis on both left and right side

polar - Polar coordinate plot

semilogx - Semilogarithmic plot

semilogy - Semilogarithmic plot

subplot - Creates axis in tiled positions

xlim - Sets or queries x-axis limits

ylim - Sets or queries y-axis limits

zlim - Sets or queries z-axis limits

Q1 (b) Plot the function;

$$y(x)=\sin(x)$$

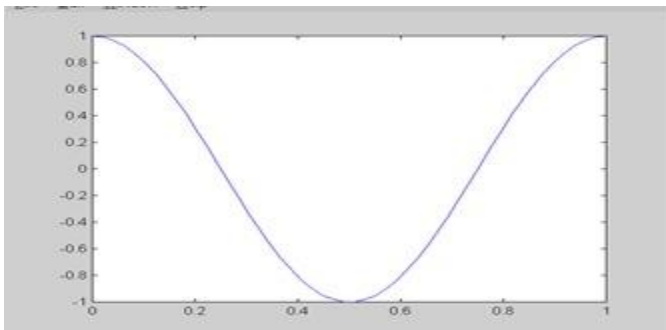
For $0 \leq x \leq 6$

$$y(x)=2\sin(x)e^{-0.2x}$$

For $0 \leq x \leq 15$

(Ans)

```
X = linspace(0 : 6 : 15);
% Create y
Y=x .* cos(x) .* sin(x);
% Plot the function
Plot(x, y, 'b-', Linewidth, 2);
Grid on;
Xlabel (X , fontsize, 15);
Ylabel (Y, Fontsize, 15);
Title (Y = X .* cos(x) .* sin(x), Fontsize, 15);
Axis equal;
% Make a black line at the x and y axis
Line (x lim, (0 , 0), Linewidth, 2);
Line (0 , 0), y lim, linewidth, 2);
```



Q.2 Plot the function;

f(x)=sin2x and its derivative d/dx sin2x on same graph

Ans;

Find the f(x)=sin2x and the derivative of sin2x =?

$$d/dx \sin(2x) = \cos(2x) = 2 \cos 2x$$

$$\sin(2x) = 2 \sin(x) * \cos(x)$$

$$(2 \cos x) \cos x + 2 \sin x (-\sin^2x) = 2 \cos^2x - 2 \sin^2x$$

Now plotting the value;

$$X = 0 : \pi / 100 : 2\pi ;$$

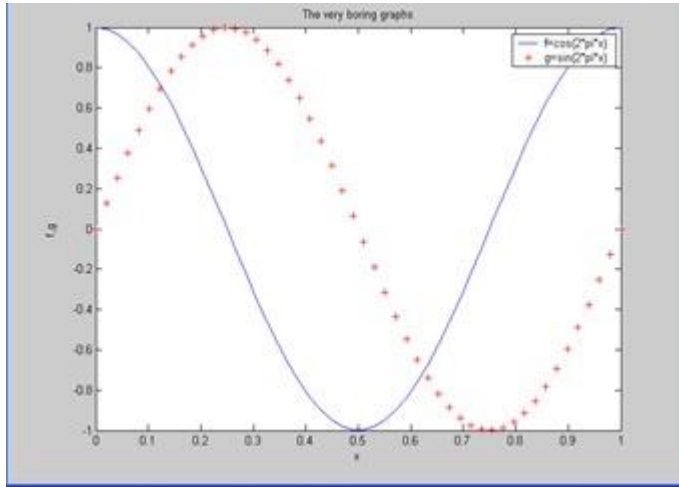
$$Y1 = \sin(2 * x)$$

$$Y2 = 2x \cos(2 * x)$$

Plot (x , y1 , 'K' , x , y2 , 'b' ..');

Title ('plot of f(x) = sin(2x) and its derivative');

Xlabel ('x');
Ylabel ('y');
Legend ('f(x)' , 'd/dx f/x' , 't1');
Grid on;



**Q.3 (a) what are the memory Management Functions in MATLAB?
(Ans).**

The memory management functions.

Below is the list of memory management functions.

1. clear - Removes variables from memory.
2. pack - Saves the existing variables to disk, and then reloads them contiguously.
3. save - Selectively persists variables to disk.
4. load - Reloads a data file saved with the save function.
5. quit - Exits MATLAB and returns all allocated memory to the system.

Memory Management

When a MEX function returns control to MATLAB®, it returns the results of its computations in the output arguments—the mxArray contained in the left-side arguments plhs[]. These arrays must have a temporary scope, so do not pass arrays created with the mexMakeArrayPersistent function in plhs. MATLAB destroys any mxArray created by the MEX function that is not in plhs. MATLAB also frees any memory that was allocated in the MEX function using the mxMalloc, mxRealloc, or mxFree functions.

Any misconstructured arrays left over at the end of a binary MEX file execution have the potential to cause memory errors.

MathWorks® recommends that MEX functions destroy their own temporary arrays and free their own dynamically allocated memory. It is more efficient to perform this cleanup in the source MEX file than to rely on the automatic mechanism. For more information on memory management techniques.

Q3) (B). Write a MATLAB function to calculate the distance between two points (x_1, y_1) and (x_2, y_2) in Cartesian coordinate system.

(Ans).

```
D = sqrt (x(p1)-x(p2))^2+(y(p1)-y(p2))^2);
>> plot (x1,y1,'sk',x2,y2,'r--')
>> legend ('7 Data Points','629 Data Points', 'Location', 'NorthEast')
>> title ('Some Sine Curves!')
>> xlabel ('x')
>> ylabel ('sin(x)')
>> grid on
>> axis tight
```

Q.4 (a) what do you mean by M-file in MATLAB?

(Ans). m-file

An m-file, or script file, is a simple text file where you can place MATLAB commands. When the file is run, MATLAB reads the commands and executes them exactly as it would if you had typed each command sequentially at the MATLAB prompt. All m-file names must end with the extension '.m' (e.g. test.m). If you create a new m-file with the same name as an existing m-file, MATLAB will choose the one which appears first in the path order (type help path in the command window for more information). To make life easier, choose a name for your m-file which doesn't already exist. To see if a filename.m already exists, type help filename at the MATLAB prompt.

USE M-FILE

For simple problems, entering your requests at the MATLAB prompt is fast and efficient. However, as the number of commands increases or trial and error is done by changing

certain variables or values, typing the commands over and over at the MATLAB prompt becomes tedious. M-files will be helpful and almost necessary in these cases.

M-File Functions

Functions are program routines, usually implemented in M-files, that accept input arguments and return output arguments. They operate on variables within their own workspace. This workspace is separate from the workspace you access at the MATLAB command prompt.

The Function Workspace

Each M-file function has an area of memory, separate from the MATLAB base workspace, in which it operates. This area, called the function workspace, gives each function its own workspace context.

While using MATLAB, the only variables you can access are those in the calling context, be it the base workspace or that of another function. The variables that you pass to a function must be in the calling context, and the function returns its output arguments to the calling workspace context. You can, however, define variables as global variables explicitly, allowing more than one workspace context to access them.

Simple Function Example

The average function is a simple M-file that calculates the average of the elements in a vector:

- function y = average(x)
- % AVERAGE Mean of vector elements.
- % AVERAGE(X), where X is a vector, is the mean of vector elements.
- % Nonvector input results in an error.
- [m,n] = size(x);
- if (~((m == 1) | (n == 1)) | (m == 1 & n == 1))
- error('Input must be a vector')
- end
- y = sum(x)/length(x); % Actual computation

Try entering these commands in an M-file called average.m. The average function accepts a single input argument and returns a single output argument. To call the average function, enter

- z = 1:99;
- average(z)
- ans =
- 50

Plot the resulting function using a 3-point thick dashed red line.

(Ans). Generate a plot for the function of $y(x) = x^2 - 3x + 2$

Sol;

$$Y(x) = x^2 - 3x + 2$$

$$x = 0 : 1 : 3$$

$$y = x^2 - 3x + 2;$$

plot (x,y)

title (plot of $x^2 - 3x + 2$);

xlabel ('x');

ylabel ('y');

grid on;

