****

**Republic of Iraq**

**Ministry of Higher Education**

**and Scientific Research**

**Al-Mustaqbal University College**

**Computer Engineering Techniques Department**

**(عملي)**

**Subject: Digital Signal Processing**

**Third stage**

**Experiment No. 1**

**By**

**Sarah Abbas & Raya Abdulwahab**

**Experiment Number 1**

**Name of Experiment:** Familiarization with MATLAB software and general functions.

**Aim:** To familiarize with MATLAB software and general functions.

**Theory:-**

MATLAB is a software package for high performance numerical computation and visualization provides an interactive environment with hundreds of built in functions for technical computation, graphics and animation. The MATLAB name stands for MATrix Laboratory produced by Mathworks Inc., USA.

At its core ,MATLAB is essentially a set (a “toolbox”) of routines (called “m files” or “mex files”) that sit on your computer and a window that allows you to create new variables with names (e.g. voltage and time) and process those variables with any of those routines (e.g. plot voltage against time, find the largest voltage, etc).

It also allows you to put a list of your processing requests together in a file and save that combined list with a name so that you can run all of those commands in the same order at some later time. Furthermore, it allows you to run such lists of commands such that you pass in data and/or get data back out (i.e. the list of commands is like a function in most programming languages). Once you save a function, it becomes part of your toolbox (i.e. it now looks to you as if it were part of the basic toolbox that you started with). For those with computer programming backgrounds: Note that MATLAB runs as an interpretive language (like the old BASIC). That is, it does not need to be compiled. It simply reads through each line of the function, executes it, and then goes on to the next line. (In practice, a form of compilation occurs when you first run a function, so that it can run faster the next time you run it.)

**MATLAB Windows:**

MATLAB works with through three basic windows.

**Command Window:** This is the main window. It is characterized by MATLAB command prompt >> when you launch the application program MATLAB puts you in this window all commands including those for user-written programs ,are typed in this window at the MATLAB prompt.

**Graphics window:** the output of all graphics commands typed in the command window are flushed to the graphics or figure window, a separate gray window with white background colour the user can create as many windows as the system memory will allow.

**Edit window:** This is where you write edit, create and save your own programs in files called M files.

**Input-output:** MATLAB supports interactive computation taking the input from the screen and flushing, the output to the screen. In addition it can read input files and write output files.

**Data Type:** The fundamental data –type in MATLAB is the array. It encompasses several distinct data objects- integers, real numbers, matrices, character strings, structures and cells. There is no need to declare variables as real or complex, MATLAB automatically sets the variable to be real.

**Dimensioning:** Dimensioning is automatic in MATLAB. No dimension statements are required for vectors or arrays. We can find the dimensions of an existing matrix or a vector with the size and length commands.

**Where to work in MATLAB?**

All programs and commands can be entered either in the

a) Command window. b) As an M file using Matlab editor.

Note: Save all M files in the folder 'work' in the current directory. Otherwise you have to locate the file during compiling. Typing quit in the command prompt >> quit, will close MATLAB Development Environment.

Basic commands in Matlab

1. T = 0:1:10

This instruction indicates a vector T which as initial value 0 and final value 10 with an increment of 1

Therefore T = [0 1 2 3 4 5 6 7 8 9 10]

1. F= 20:1:100

Therefore F = [20 21 22 23 24 ……… 100]

1. T= 0: 1/pi: 1

Therefore T= [0, 0.3183, 0.6366, 0.9549]

1. zeros (1, 3)

The above instruction creates a vector of one row and three columns whose values are zero

Output= [0 0 0]

1. zeros (2,4)

Output = 0 0 0 0

0 0 0 0

1. ones (5,2)

The above instruction creates a vector of five rows and two columns Output = 1 1

1 1

1 1

1 1

1 1

7. a = [ 1 2 3] b = [4 5 6]

a.\*b = [4 10 18]

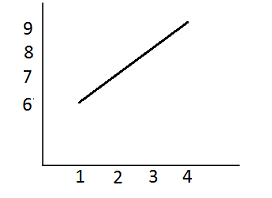
8. If C= [2 2 2]

b.\*C results in [8 10 12]

9. plot (t, x)

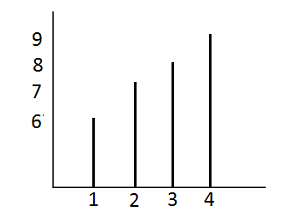
If x = [6 7 8 9] t = [1 2 3 4]

This instruction will display a figure window which indicates the plot of x versus t.

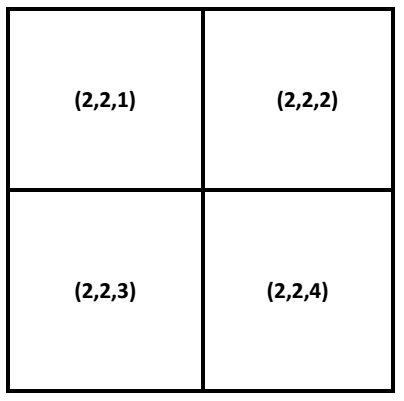


10. stem(t,x) :-

This instruction will display a figure window as shown



11. Subplot: This function divides the figure window into rows and columns. Subplot (2 2 1) divides the figure window into 2 rows and 2 columns 1 represent number of the figure



12. Conv

Syntax: w = conv(u,v)

Description: w = conv(u,v) convolves vectors u and v. Algebraically, convolution is the sameoperation as multiplying the polynomials whose coefficients are the elements of u and v.

13. Disp

Syntax: disp(X)

Description: disp(X) displays an array, without printing the array name. If X contains a textstring, the string is displayed. Another way to display an array on the screen is to type its name,but this prints a leading "X=," which is not always desirable. Note that disp does not displayempty arrays.

14. xlabel

Syntax: xlabel('string')

Description: xlabel('string') labels the x-axis of the current axes.

15. ylabel

Syntax :ylabel('string')

Description: ylabel('string') labels the y-axis of the current axes.

16. Title

Syntax : title('string')

Description: title('string') outputs the string at the top and in the center of the current axes.

17. grid on

Syntax : grid on

Description: grid on adds major grid lines to the current axes.

18. FFT

FFT(X) is the discrete Fast Fourier transform (FFT) of vector X. For matrices, the FFToperation is applied to each column. For N-D arrays, the FFT operation operates on the firstnon-singleton dimension. FFT(X,N) is the N-point FFT, padded with zeros if X has less than Npoints and truncated if it has more.

19. ABS

Absolute value.

ABS(X) is the absolute value of the elements of X. When X is complex, ABS(X) is the complexmodulus (magnitude) of the elements of X.

20. ANGLE Phase angle.

ANGLE(H) returns the phase angles, in radians, of a matrix with complex elements.