# AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)

## DATA COMMUNICATION

**Spring 2022-2023**

**Section: D**

**LAB REPORT ON**

***Introduction to MATLAB***

**EXPERIMENT NO-01**

**Supervised By**

**DR.MD.HUMAYUN KABIR**

**Submitted By**

| Name | ID |
|---|---|
| S. M. Rasel | 22-48039-2 |

S.M. Rasel
22-48039-2

Title: Introduction to MATLAB

Objective: The objective of this experiment was to understand the use of MATLAB in various equations of data communication engineering problems. Using the obtained knowledge various commands, Syntax and tools had to be used and implemented to obtaing the necessary results.

Introduction: MATLAB is a high-performance language for techical compution. It integrates computation programming and visualization in a user-friendly environment where problem and solution are expressed in an easy-to-understand mathematical notation. MATLAB is an interactive system whose basic data element is an array that does not requrie diasensioning. This allows the user to solve many technical computing problems, especilly those with matrix and vector operation in less time than it would take to write a program in a scalar no interactive language such as C or FORTRAN. MATLAB features a family of applications-specific language solution which are called tool boxes. It is very important to most users of MATLAB that toolboxes allow to learn and apply specialized.

S.M. Rasel

S.M. Rasel
22-18039-2

features. These toolboxes are comprehensive collections of MATLAB fuction, so called M files extending the MATLAB environment to solve particular mathematical and engineering problems. MATLAB is a Matrix-based programming tool. Although matrics often need not be dimensioned explicitly, the user must always look carefully for matrix dimension. If it is not defined otherwise, the standard matrix exhibits two dimension $n \times m$. Coloumn vectors and row vector are represented consistently by $n \times 1$ and $1 \times n$ matrics, respectively.

Simulation Tools or Software: MatLaB 2016a.

Simulation:

Entering Matrices and Addressing the elements

Code:

```
>> A = [1 2 3; 8 6 4; 3 6 9]
```

S.M. Rasel
22-18039-2

The above code was used to create a matrix 'A' represe-ents the name of the Matrix and the value within the third brackets are elements of the Matrix. If there are multiple column in the matrix a space () was used to separate the elemints and to end a row a semicolon (;) was used after finishing entering the elements of a row.

Code:

```
>> A(1,3) + A(2,1) + A(3,2)
```

```
Command Window
>> A(1,3) + A(2,1) + A(3,2)

ans =

    17

fx >>
```

The above code sums up the seleted elements. To select the elements, the name of the matrix was use along with the position of the Matrix. If the matrix is A(m,n) 'm' represents the row and 'n' is Coloumn.

Code:

```
>> A(2:3,1,2)
```

```
Command Window
>> A(2:3, 1:2)

ans =

     8     6
     3     6

fx >>
```

S.M. Rasel
22-48039-2

The above code represents the code within a certain range. If the range of a Matrix is A(m:n, k:l), the range of the row 'm' to 'n' and the range of the column was 'k' to 'l'. All the elements within this range was display on the prompt.

Code:

```
>> A(1,1:2)
```

Command Window

```
>> A(1, 1:2)

ans =

     1     2

fx >>
```

The above code displayed the elements of the first row which are within '1' to '2' column. In this same way, the elements of a certain column can be represents as well.

Code:

```
>> A(:,2)
```

Command Window

```
>> A(:, 2)

ans =

     2
     6
     6

fx >>
```

In this code implementation, all the elements of a certain row or column were represented. In the all above example all, the elements of the 2nd column of 'A' matrix have been represend. To all rows ':' sign. was used.

S.M. Rasel (3)
22-48039-2

## Generating Matrices:

Code

```
>> v = (10:-2:0)
```

Command Window

```
>> v = (10:-2:0)

v =

    10     8     6     4     2     0

fx >>
```

A Matrix was generated. The start value of the Matrix was 10 and the end value was 0; The value decreased by -2. Here, a matrix was generated which had 1 row and 6 columns with 6 elements.

Code:

```
>> w = (5:10)
```

Command Window

```
>> w = (5:10)

w =

     5     6     7     8     9    10

fx >>
```

In the above implementation a matrix was generated which started with 5 and ended with 10. In this case, the difference between the starting and ending was not explicitly defined in the code. Hence, the value was increased by 1, which is the default incremental value


CS Scanned with CamScanner

S.M. Rasel
22-98039-2

Code

```
>>B =zeros(3,4)
```

```
Command Window
>> B = zeros(3, 4)

B =

     0     0     0     0
     0     0     0     0
     0     0     0     0

fx >>
```

Using the above command, a 3X4 matrix is generated which contained only zeros (0) as its elements.

Code:

```
>>C = ones(2,5)*6
```

```
Command Window
>> C = ones(2, 5) * 6

C =

     6     6     6     6     6
     6     6     6     6     6

fx >>
```

Using the 'ones' function, a (m,n) matrix is generated which contains only '1' as elements of its matrix After generating the ones matrix, 6 it's multiplied

Code:

```
>> D= rand (1,5)
```

```
Command Window
>> D = rand(1, 5)

D =

    0.8147    0.9058    0.1270    0.9134    0.6324

fx >>
```

To Create a matrix with random values, 'rand' function is used. In the above code, a (1X5) matrix is created with random values. the 'rand' function use's positive values only to fill the elements of matrix.

22-48039-2

Code:

```
>> E = randn(3,3)
```

```
Command Window
>> E = randn(3, 3)

E =

   -1.3077    3.5784    3.0349
   -0.4336    2.7694    0.7254
    0.3426   -1.3499   -0.0631

fx >>
```

The above function works just like 'rand', The difference between the two is 'randn uses any real number as elements of the matrix that it is implemented on.

Deleting rows and columns:

Code:

```
>> A(2,:)=[]
```

```
Command Window
>> A(2, :) = [ ]

A =

     1     2     3
     3     6     9

fx >>
```

To delete a certain element, the above code was used. In the above implementation, the whole 2nd row of Matrix 'A' was deleted.

Array Orientation:

Code:

```
>> a = 0:3;
>> b = a'
```

```
Command Window
>> a = 0:3

a =

     0     1     2     3

fx >>
```

22-18039-2

## Scalar – Array Mathematics:

Code:

```
>> c = [1 2 3 4; 5 6 7 8;
        9 10 11 12];
>> 2*c-1
```

```
Command Window
>> b = a'

b =

     0
     1
     2
     3

fx >>
```

```
Command Window
>> c = [1 2 3 4;5 6 7 8;9 10 11 12];
>> 2*c-1

ans =

     1     3     5     7
     9    11    13    15
    17    19    21    23

fx >>
```

The above implementation creates a matrix. called 'c', All the elements of Matrix 'c' have been multiplied by 2 and sub by 1 from each element

## Array – Array Mathematics

Code:

```
>> d = [1 2 3; 4 5 6]
>> e = [2 2 2; 3 3 3]
>> f = d + e
>> g = 2*d - e
>> h = d.*e
>> d./e
>> e.\d
```

```
Command Window
>> d = [1 2 3; 4 5 6];
>> e = [2 2 2; 3 3 3];
>> f = d + e

f =

     3     4     5
     7     8     9

>> g = 2*d - e

g =

     0     2     4
     5     7     9

>> h = d.*e

h =

     2     4     6
    12    15    18

>> d./e

ans =

    0.5000    1.0000    1.5000
    1.3333    1.6667    2.0000

>> e.\d

ans =

    0.5000    1.0000    1.5000
    1.3333    1.6667    2.0000

fx >>
```

At first two matrics were entered called 'd' and 'e'. 'f' holds the value of the sum of 'd' and 'e'. Then 'd' is was multiplied by 2 and then subtracted by 'e'. Element-by-element mult.

S.M. Kasel
22-48039-2

Code:

```
>> A = [1 2 3; 4 5 6]
>> B = [1 2; 3 4; 5 6]
>> C = A * B
```

```
Command Window
>> A = [1 2 3; 4 5 6]

A =

     1     2     3
     4     5     6

>> B = [1 2; 3 4; 5 6]

B =

     1     2
     3     4
     5     6

>> C = A * B

C =

    22    28
    49    64

>>
```

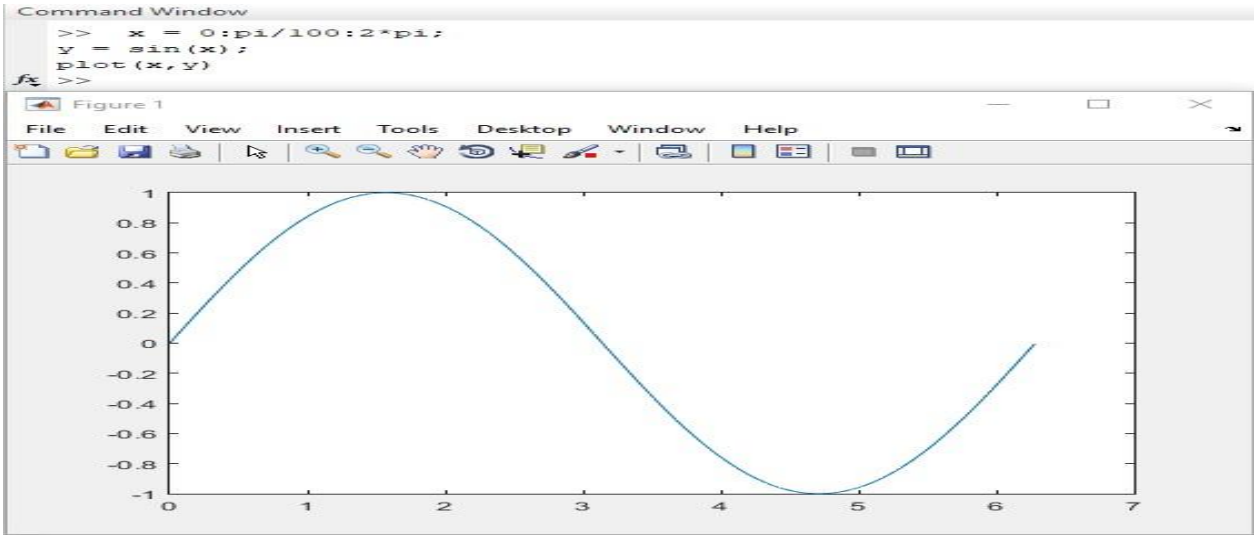
The above implemention peform'd To peform this '*' operator

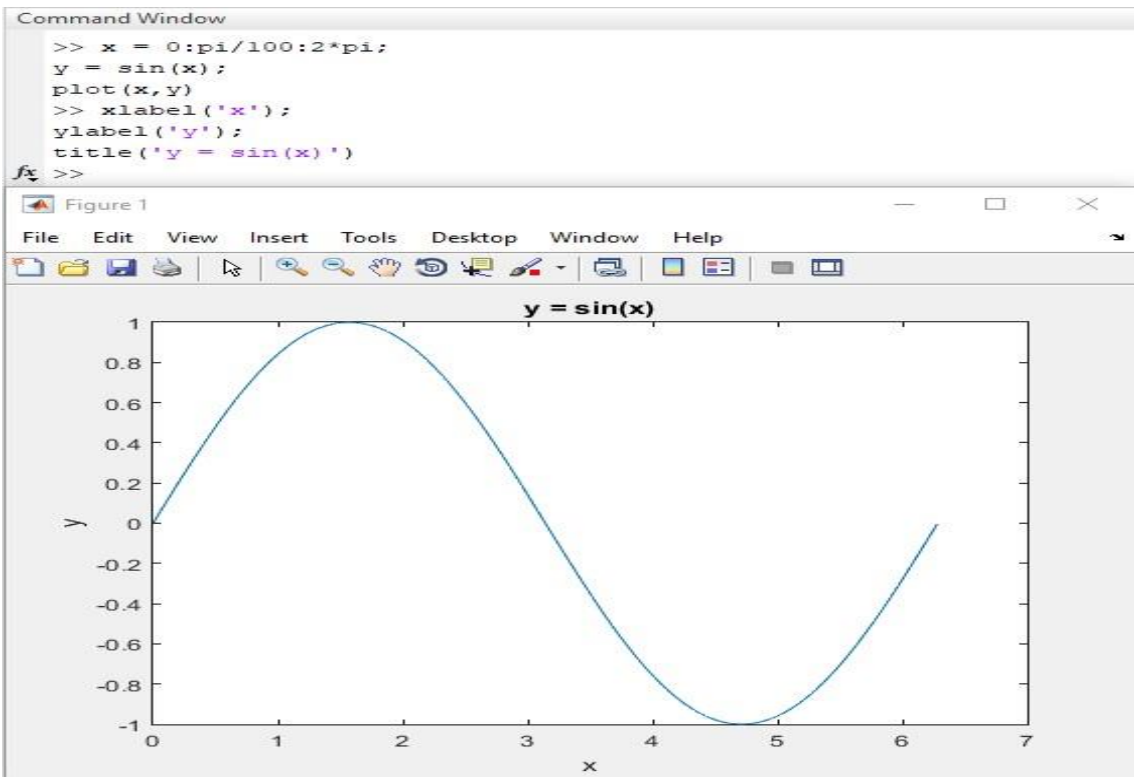
## Creating a plot:

Code:

```
>> x = 0: pi/100: 2*pi;
>> y = sin(x);
>> plot (x,y)
```

The above code implementation plots a curve of a certain equation. ('x') represents the range of the values and the difference between each point of the range. 'y' also used to plotted on the graph. The function of 'plot', the ~~#~~ plotted graph emerged.


Command Window
>> x = 0:pi/100:2*pi;
y = sin(x);
plot(x,y)
>>
Figure 1
File Edit View Insert Tools Desktop Window Help

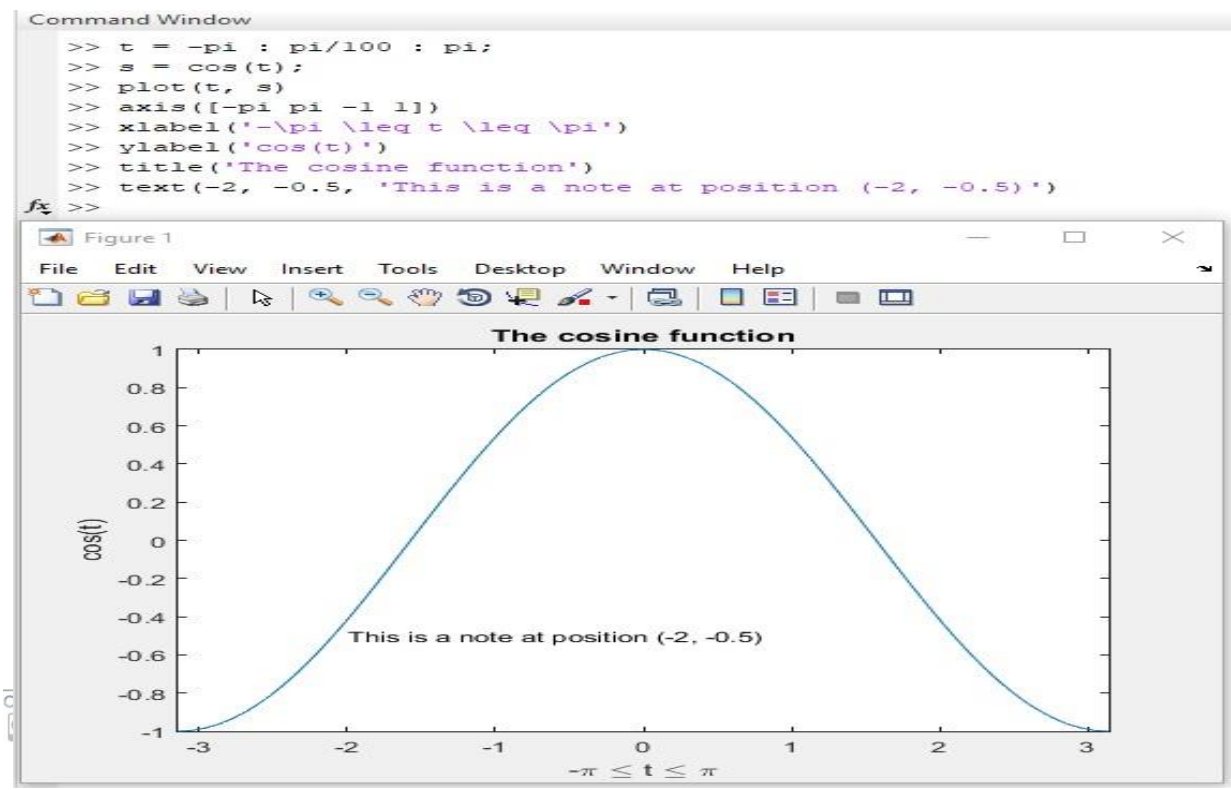

Command Window
>> x = 0:pi/100:2*pi;
y = sin(x);
plot(x,y)
>> xlabel('x');
ylabel('y');
title('y = sin(x)')
>>
Figure 1
File Edit View Insert Tools Desktop Window Help
y = sin(x)
y
x

Expt No: 01 Date: 03 March 2025 Data Sheet

ST ID: 22-48039-2 Name: S.M. Rasel

03/03

```
>> t = -pi : pi/100 : pi;
   s = cos(t);
   plot (t,s)
   axis ([-pi pi -1 1])
   xlabel('-\pi \leq t \leq \pi')
   ylabel('cos(t)')
   title('The cosine function')
   text(-2,-0.5,'This is a note at position(-2,-0.5)')
```



S.M. Rasel
22-18039-2

Code:

```
>>x = 0: pi/100: 2*pi;
>>y = sin(x);
>>plot (x,y)
>>xlabel ('X');
>>ylabel ('y');
>>title('y = sin(x)')
```

formatting a graph is another important part of graph plotting. To table the x-axis and y-axis, the function 'xlabel' and 'ylabel'. 'title' was used to give the title of the graph.

Discussion and Conclusion: From the above simulation, various functions of MATLAB. were observed in hand. Various functions that were available on MATLAB were learned and observed. Using this knowledge, graphs were plotted in MATLAB Software.

References:

1. Prakash C. Gupta, "Data Com.", prentice Hall India
2. William Stallings, "Data Com." Person
3. AIUB Data Com. Engineering Lab Mannal.