

Title: Introduction to MATLAB

Abstract:

This experiment is designed to-

- 1.To understand the use of MATLAB for solving communication engineering problems.
- 2.To develop understanding of MATLAB environment, commands, and syntax.

Introduction:

Matlab is a high-performance language for technical computing. It integrates computation, programming and visualization in a user-friendly environment where problems and solutions are expressed in an easy-to-understand mathematical notation.

Matlab is an interactive system whose basic data element is an array that does not require dimensioning. This allows the user to solve many technical computing problems, especially those with matrix and vector operations, in less time than it would take to write a program in a scalar noninteractive language such as C or Fortran.

Matlab features a family of application-specific solutions which are called toolboxes. It is very important to most users of Matlab, that toolboxes allow to learn and apply specialized technology. These toolboxes are comprehensive collections of Matlab functions, so-called M files, that extend the Matlab environment to solve particular classes of problems.

Matlab is a matrix-based programming tool. Although matrices often need not to be dimensioned explicitly, the user has always to look carefully for matrix dimensions. If it is not defined otherwise, the standard matrix exhibits two dimensions' $n \times m$. Column vectors and row vectors are represented consistently by $n \times 1$ and $1 \times n$ matrices, respectively.

Matlab operations can be classified into the following types of operations:

- arithmetic and logical operations,
- mathematical functions,
- graphical functions, and
- input/output operations.

Performance Task:

1.

```
n = [1, 2, 3, 4, 5];
```

```
A = [42, 90, 18, 35, 75];
```

```
B = [67, 68, 45, 12, 18];
```

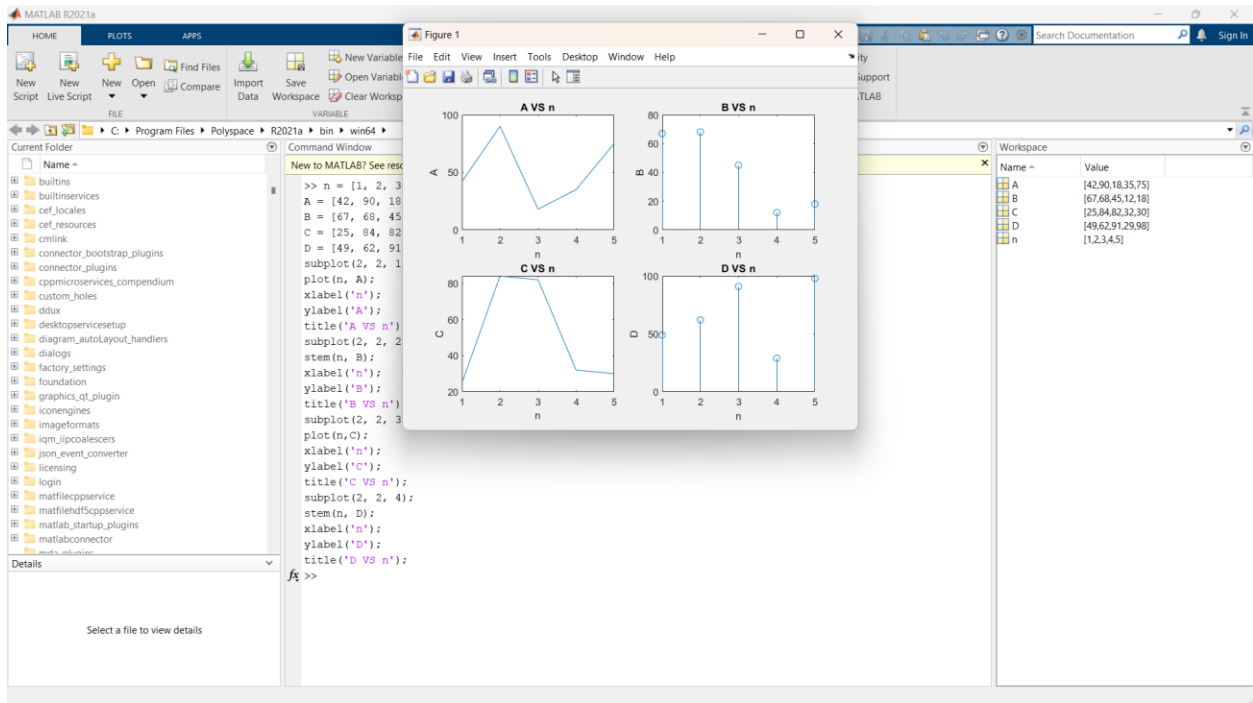
```
C = [25, 84, 82, 32, 30];
```

```
D = [49, 62, 91, 29, 98];
```

```
subplot(2, 2, 1);
```

```
plot(n, A);
```

```
xlabel('n');  
ylabel('A');  
title('A VS n');  
subplot(2, 2, 2);  
stem(n, B);  
xlabel('n');  
ylabel('B');  
title('B VS n');  
subplot(2, 2, 3);  
plot(n,C);  
xlabel('n');  
ylabel('C');  
title('C VS n');  
subplot(2, 2, 4);  
stem(n, D);  
xlabel('n');  
ylabel('D');  
title('D VS n');
```



2.

$F = 336;$

$f = 0.05;$

$A1 = 2;$

$A2 = 8;$

$j1 = \text{deg2rad}(12);$

$j2 = \text{deg2rad}(30);$

$t = -30:0.1:30;$

$x1 = A1 * \cos(2 * \pi * f * t) + j1;$

$x2 = A2 * \cos(2 * \pi * f * t) + j2;$

`subplot(3, 1, 1);`

`plot(x1);`

`xlabel('Time(t)');`

`ylabel('Amplitude(x1)');`

`title('t VS x1');`

`subplot(3, 1, 2);`

`plot(x2);`

```

xlabel('Time(t)');

ylabel('Amplitude(x2)');

title('t VS x2');

x3 = x1 + x2;

subplot(3, 1, 3);

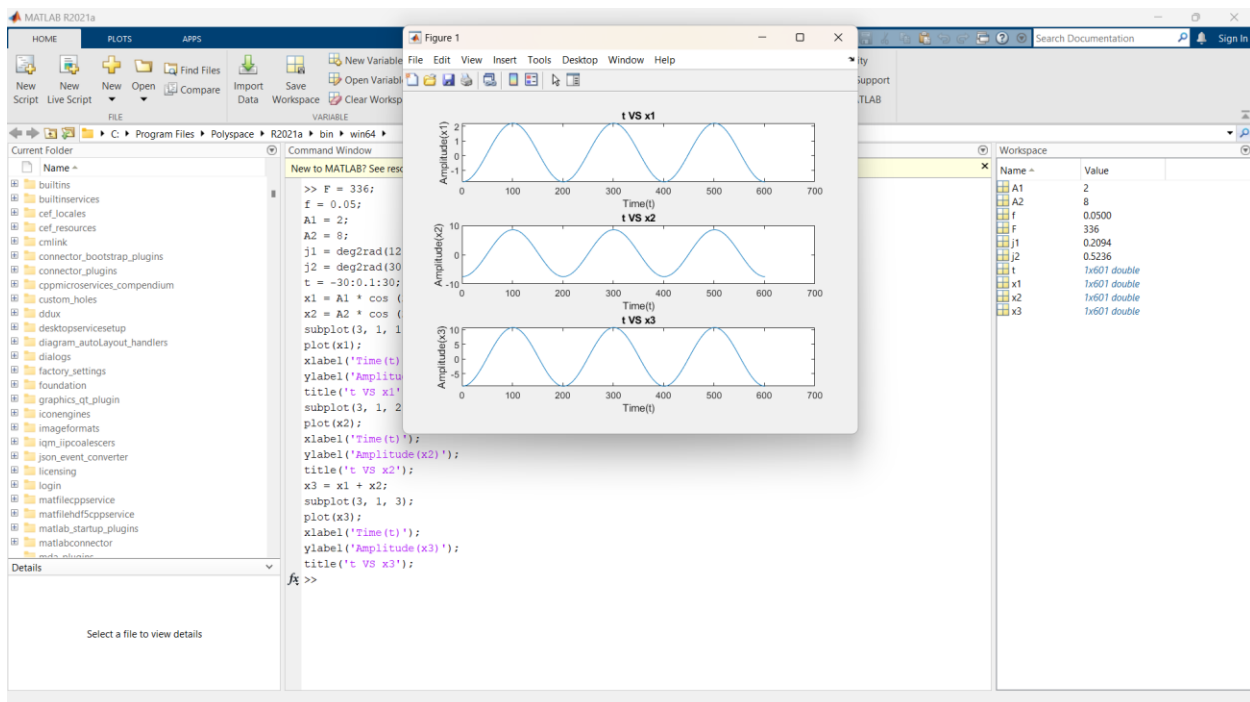
plot(x3);

xlabel('Time(t)');

ylabel('Amplitude(x3)');

title('t VS x3');

```



Discussion:

Matlab is a matrix-based programming tool. Although matrices often need not to be dimensioned explicitly, the user has always to look carefully for matrix dimensions. If it is not defined otherwise, the standard matrix exhibits two dimensions' $n \times m$. Column vectors and row vectors are represented consistently by $n \times 1$ and $1 \times n$ matrices, respectively.

Conclusion:

Matlab is a high-performance language for technical computing. It integrates computation, programming and visualization in a user-friendly environment where problems and solutions are expressed in an easy-to-understand mathematical notation.