

# **Computer Programming in C++ and MATLAB**

## **Contents**

### **Programming in C++**

<b>No.</b>	<b>Date</b>	<b>Name of the Problem</b>	<b>Page No.</b>
1	22.07.2016	Program to solve a quadratic equation	1--2
2	22.07.2016	Program to find the factorial of a non-negative integer	3--4
3	22.07.2016	Program to find GCD of two positive integers	5--6
4	29.07.2016	Program to arrange a list of numbers in ascending order	7--8
5	29.07.2016	Program for matrix addition	9--10
6	29.07.2016	Program for matrix multiplication	11--12
7	12.08.2016	Program for forward difference table	13--15
8	12.08.2016	Program for interpolation by Newton's forward formula	16--18
9	12.08.2016	Program for interpolation by Newton's backward formula	19--21
10	18.08.2016	Program for interpolation by Lagrange's formula	22--23
11	18.08.2016	Program for numerical integration by Trapezoidal rule	24--25
12	18.08.2016	Program for numerical integration by Simpson's one-third rule	26--27
13	09.09.2016	Program for solution of an equation by Newton Raphson method	28--29
14	09.09.2016	Program for Gauss elimination method	30-33
15	16.09.2016	Program for numerical solution of ODE by 4 <sup>th</sup> order Runge-Kutta method	34--35
16	16.09.2016	Statistical data analysis	36--40
17	16.09.2016	Program to find the correlation coefficient and regression lines of a bivariate data	41--43

# **Programming in MATLAB**

No.	Date	Name of the Problem	Page No.
1	23.09.2016	Matrices and matrix operations	44--46
2	23.09.2016	Eigenvalues, eigenvectors and decompositions of matrices	47--49
3	23.09.2016	Solution of a system of equations	50--51
4	21.10.2016	Conditional structure, User-defined functions, Loop structure	52--54
5	21.10.2016	Numerical integration	55
6	21.10.2016	Solution of polynomial and transcendental equations	56--58
7	28.10.2016	2D and 3D graphics	59---64
8	28.10.2016	Solutions of differential equations	65--70
9	28.10.2016	Handling complex numbers	71
10	04.11.2016	Statistical data analysis	72--76
11	04.11.2016	Finding correlation coefficient and curve fitting	77--79
12	04.11.2016	Solution of linear programming problems	80--82

# ***Programming in C++***

### ***Program to solve a quadratic equation***

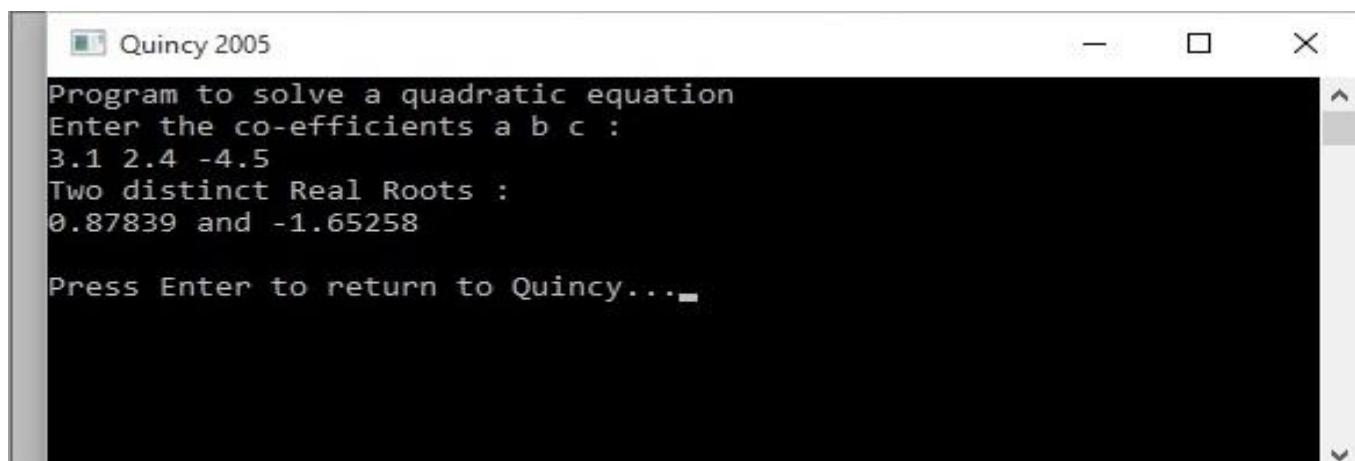
**Problem 1.** Write a program in C++ to solve the quadratic equation  $(2.1+k)x^2+2.4x-4.5=0$  correct to three decimal places, where  $k = 1$ . Run your program and write the output.

//Program to solve a quadratic equation

```
#include<iostream>
#include<cmath>
using namespace std;
int main()
float a,b,c,P,Q,D;
{
cout<<"Program to solve a quadratic equation :";
cout<<"Enter the co-efficients a b c :";
cin>>a>>b>>c;
D=b*b - 4*a*c;
P=- (b/ (2*a) );
Q=sqrt (fabs (D) ) / (2*a) ;
if (0<D)
{
cout<<"Two distinct Real Roots : "<<endl;
cout<<(P+Q)<<" and "<<(P-Q)<<endl; }

else if (D==0)
{
cout<<"Two Equal Real Roots : "<<endl;
cout<<P<<endl;
}
else
{
cout<<"Two distinct Imaginary Roots : "<<endl;
cout<<"("<<P<<" + i"<<Q<<")"<<" and "<<"("<<P<<" - i"<<Q<<")"<<endl;
}
return 0;
}
```

## Out Put:



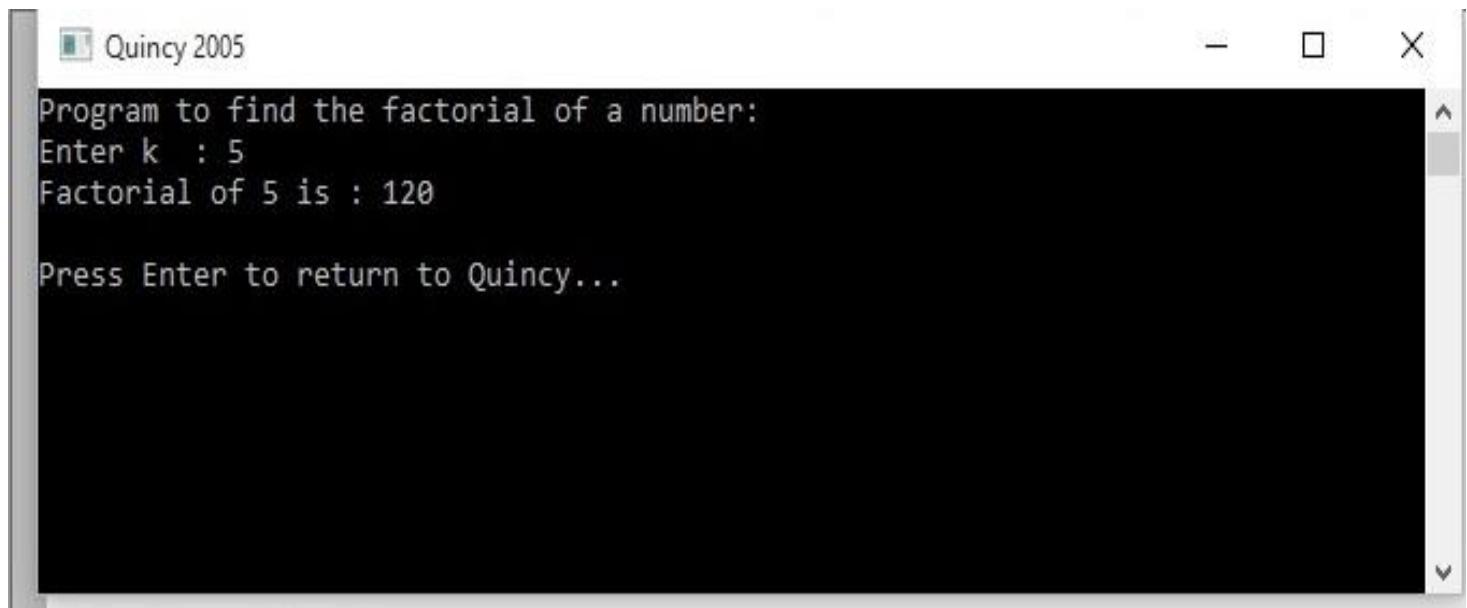
The screenshot shows a terminal window titled "Quincy 2005". The window contains the following text:  
Program to solve a quadratic equation  
Enter the co-efficients a b c :  
3.1 2.4 -4.5  
Two distinct Real Roots :  
0.87839 and -1.65258  
Press Enter to return to Quincy...  
The window has standard operating system window controls (minimize, maximize, close) at the top right.

## ***Program to find the factorial of a given non-negative integer***

**Problem 2.** Write a program in C++ to find the factorial of a given non-negative integer. Run your program and write the output.

```
// Program to find the factorial of a number:  
#include<iostream>  
using namespace std;  
int factorial(int n);  
int main()  
{  
    int k;  
    cout<<"Program to find the factorial of a number: "<<endl;  
    cout<<"Enter k : ";  
    cin>>k;  
  
    cout<<"Factorial of "<<k<<" is : "<<factorial(k)<<endl;  
  
    return 0;  
}  
  
int factorial(int n)  
{  
    int t=1;  
  
    if(0<n)  
    {  
        for(int i=1;i<n+1;i++)  
  
        {  
            t=i*t;  
        }  
    }  
  
    return t;  
}
```

## Out Put:



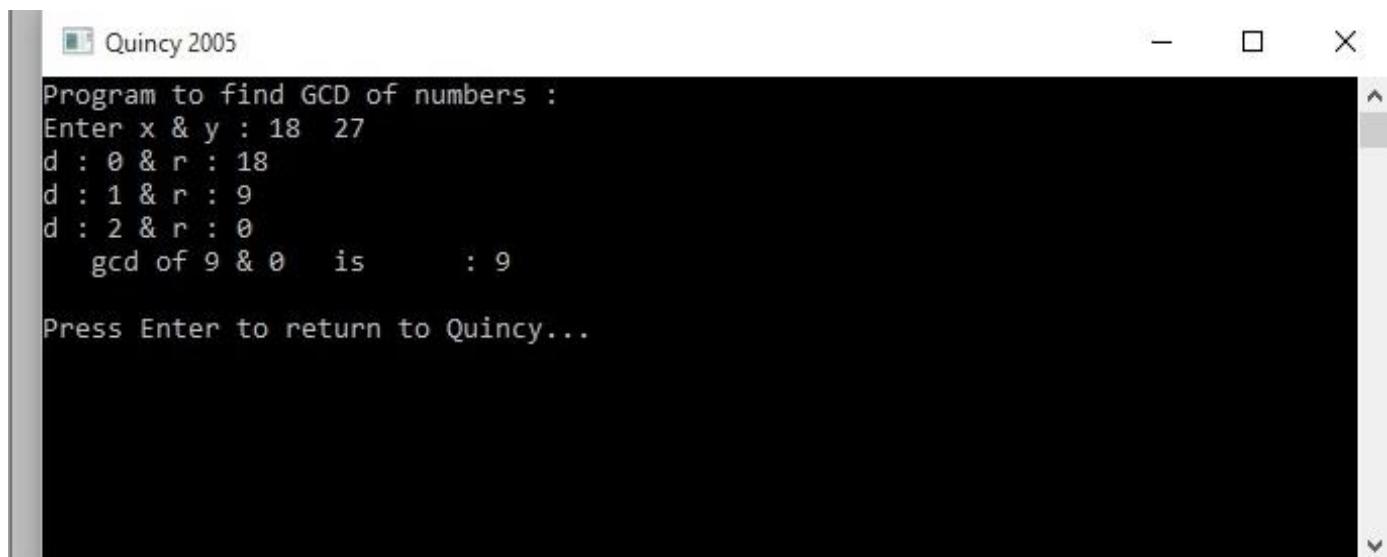
The screenshot shows a terminal window titled "Quincy 2005". The window contains the following text:  
Program to find the factorial of a number:  
Enter k : 5  
Factorial of 5 is : 120  
Press Enter to return to Quincy...

***Program to find the GCD of two given positive integers***

**Problem 3.** Write a program in C++ to find the GCD of two given positive integers. Run your program and write the output.

```
//Program to find GCD :  
#include<iostream>  
#include<cmath>  
using namespace std;  
int main()  
{  
    int x,y,d,r,temp;  
  
    cout<<"Enter x & y : ";  
    cin>>x>>y;  
    do  
    {  
        r=x%y;  
        d=(x-r)/y;  
        cout<<"d : "<<d<<" & r : "<<r<<endl;  
        temp=y;  
        y=r;  
        x=temp;  
    }  
    while(r!=0);  
    cout<<"gcd of "<<x<<" & "<<y<<" is : "<<temp<<endl;  
  
    return 0;  
}
```

## Out Put:



The screenshot shows a terminal window titled "Quincy 2005". The window contains the following text:

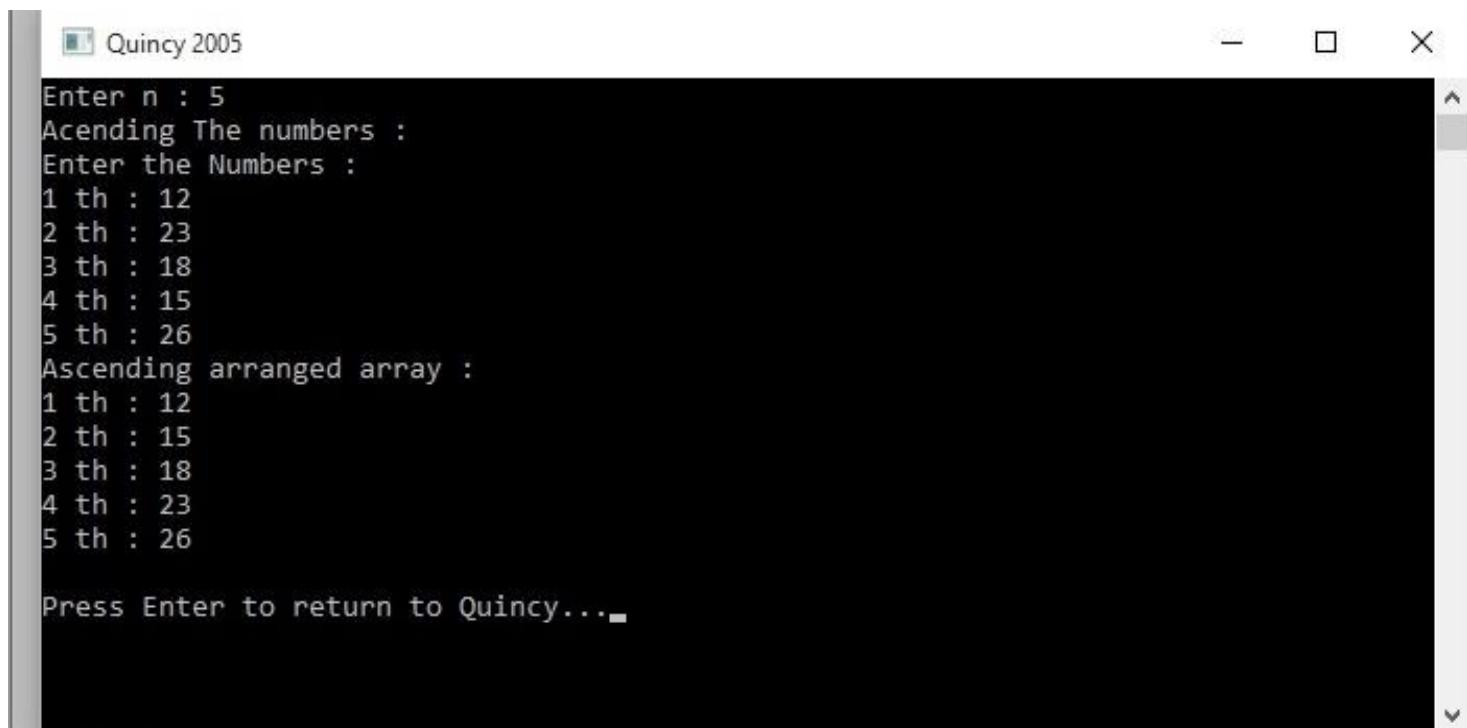
```
Program to find GCD of numbers :  
Enter x & y : 18 27  
d : 0 & r : 18  
d : 1 & r : 9  
d : 2 & r : 0  
gcd of 9 & 0 is : 9  
  
Press Enter to return to Quincy...
```

***Program to arrange a list of numbers in ascending order***

**Problem 4.** Write a program in C++ to arrange a list of numbers in ascending order. Run your program and write the output.

```
//Acending The numbers :
#include<iostream>
using namespace std;
int main()
{
    int n;
    float x[100],c;
    cout<<"Enter n : ";
    cin>>n;
    cout<<"Acending the Numbers :"<<endl;
    cout<<"Enter the Numbers : "<<endl;
    for(int i=0;i<n;i++)
    {
        cout<<(i+1)<<" th : ";
        cin>>x[i];
    }
    for(int i=0;i<n;i++)
    {
        for(int j=i+1;j<n;j++)
        {
            if(x[j]<x[i])
            {
                c=x[i];
                x[i]=x[j];
                x[j]=c;
            }
        }
    }
    cout<<"Ascending arranged array : "<<endl;
    for(int i=0;i<n;i++)
    {
        cout<<(i+1)<<" th : "<<x[i]<<endl;
    }
}
return 0;
```

## Out Put:



The screenshot shows a terminal window titled "Quincy 2005". The window contains the following text output:

```
Enter n : 5
Acending The numbers :
Enter the Numbers :
1 th : 12
2 th : 23
3 th : 18
4 th : 15
5 th : 26
Ascending arranged array :
1 th : 12
2 th : 15
3 th : 18
4 th : 23
5 th : 26

Press Enter to return to Quincy...■
```

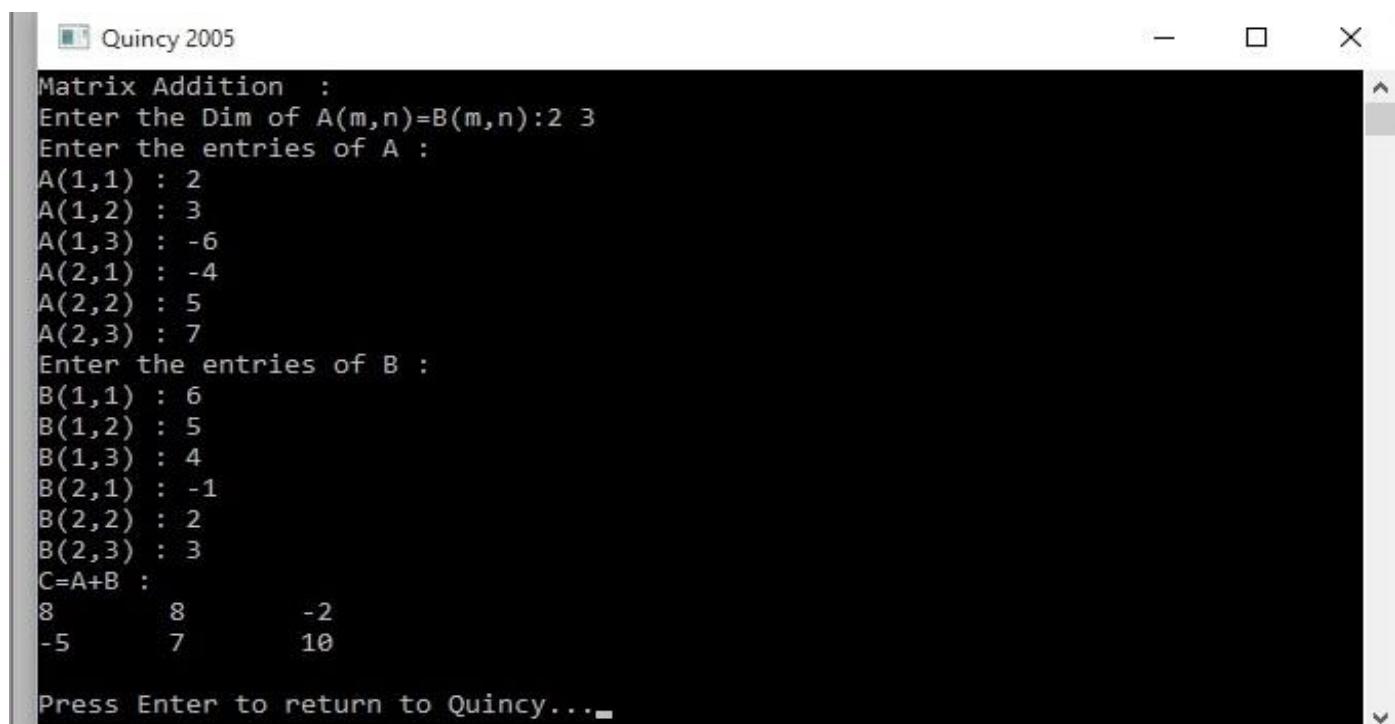
### ***Program for matrix addition***

**Problem 5.** Write a program in C++ to add two matrices. Run your program and write the output.

```
//Program for matrix addition
#include<iostream>
using namespace std;
int main()
{
int m,n;
float A[100][100],B[100][100],C[100][100];
cout<<"Matrix Addition : "<<endl;
cout<<"Enter the Dim of A(m,n)=B(m,n) : ";
cin>>m>>n;
cout<<"Enter the entries of A : "<<endl; for(int
i=1;i<m+1;i++)
{
for(int j=1;j<n+1;j++)
{
cout<<"A("<<i<<", "<<j<<") : ";
cin>>A[i][j];
}
}
cout<<"Enter the entries of B : "<<endl;
for(int i=1;i<m+1;i++)
{
for(int j=1;j<n+1;j++)
{
cout<<"B("<<i<<", "<<j<<") : ";
cin>>B[i][j];
}
}
cout<<"C=A+B : "<<endl;
for(int i=1;i<m+1;i++)
{
for(int j=1;j<n+1;j++)
{
C[i][j]=A[i][j]+B[i][j];
cout<<C[i][j]<<"      ";
}
}
cout<<endl;
}
```

```
return 0;  
}
```

## Out Put:



Quincy 2005

```
Matrix Addition :  
Enter the Dim of A(m,n)=B(m,n):2 3  
Enter the entries of A :  
A(1,1) : 2  
A(1,2) : 3  
A(1,3) : -6  
A(2,1) : -4  
A(2,2) : 5  
A(2,3) : 7  
Enter the entries of B :  
B(1,1) : 6  
B(1,2) : 5  
B(1,3) : 4  
B(2,1) : -1  
B(2,2) : 2  
B(2,3) : 3  
C=A+B :  
8      8      -2  
-5      7      10  
Press Enter to return to Quincy...■
```

## ***Program for matrix multiplication***

**Problem 6.** Write a program in C++ to find the product of two matrices. Run your program and write the output.

```
//Program For Matrix Multiplication:
#include<iostream>
using namespace std;
int main()
{
int m,n,p;
float A[100][100],B[100][100],C[100][100];
cout<<"Matrix Multiplication:"<<endl;
cout<<"Enter the Dim of A(m, n) : ";
cin>>m>>n;

cout<<"Enter the entries of A :
"<<endl; for(int i=1;i<m+1;i++) {

    for(int j=1;j<n+1;j++)
    {
        cout<<"A("<<i<<","<<j<<") : ";
        cin>>A[i][j];
    }
}
cout<<"Enter the dim of B("<<n<<",p)
:"; cin>>p;
for(int i=1;i<n+1;i++)
{
    for(int j=1;j<p+1;j++)
    {
        cout<<"B("<<i<<","<<j<<") : ";
        cin>>B[i][j];
    }
}
cout<<"C=AB : "<<endl;
for(int i=1;i<m+1;i++)
{
    for(int j=1;j<p+1;j++)
    {
```

```
C[i][j]=0;
for(int k=1;k<n+1;k++)
{
    C[i][j]=C[i][j] + A[i][k]*B[k][j];
}
cout<<C[i][j]<<"";
cout<<endl;
}

return 0;
}
```

## Out Put:



Quincy 2005

```
Matrix Multiplication:
Enter the Dim of A(m,n) : 2 3
Enter the entries of A :
A(1,1) : 2
A(1,2) : 3
A(1,3) : -2
A(2,1) : 5
A(2,2) : 6
A(2,3) : -7
Enter the dim of B(3,p) :3
B(1,1) : 6
B(1,2) : 4
B(1,3) : -5
B(2,1) : 3
B(2,2) : 8
B(2,3) : 9
B(3,1) : 1
B(3,2) : 4
B(3,3) : 3
C=AB :
19      24      11
41      40       8

Press Enter to return to Quincy....
```

***Interpolation***  
***Computing a difference table***

**Problem 7.** Write a program in C++ to construct the forward difference table for the following data

$x$	$f(x)$
7.05	1.73168
7.10	1.72827
7.15	1.72484
7.20	1.72138
7.25	1.71791

where  $k=1$  . Run your program and write the output.so,

$x$	$f(x)$
7.05	1.73168
7.10	1.72827
7.15	1.72484
7.20	1.72138
7.25	1.71791

```
//Computing a difference table
#include<iostream>
using namespace std;
int main()
{
int n;
float d[100][100],h,a,b;
//d[i][j]= Ai[yj],c[i] is coefficient array//
cout<<"Enter n : ";
cin>>n;
cout<<"Compute Difference table"<<endl;
cout<<"Enter a : ";
cin>>a;

cout<<"Enter b : ";
cin>>b;
```

```

h=fabs (b-a) /n;

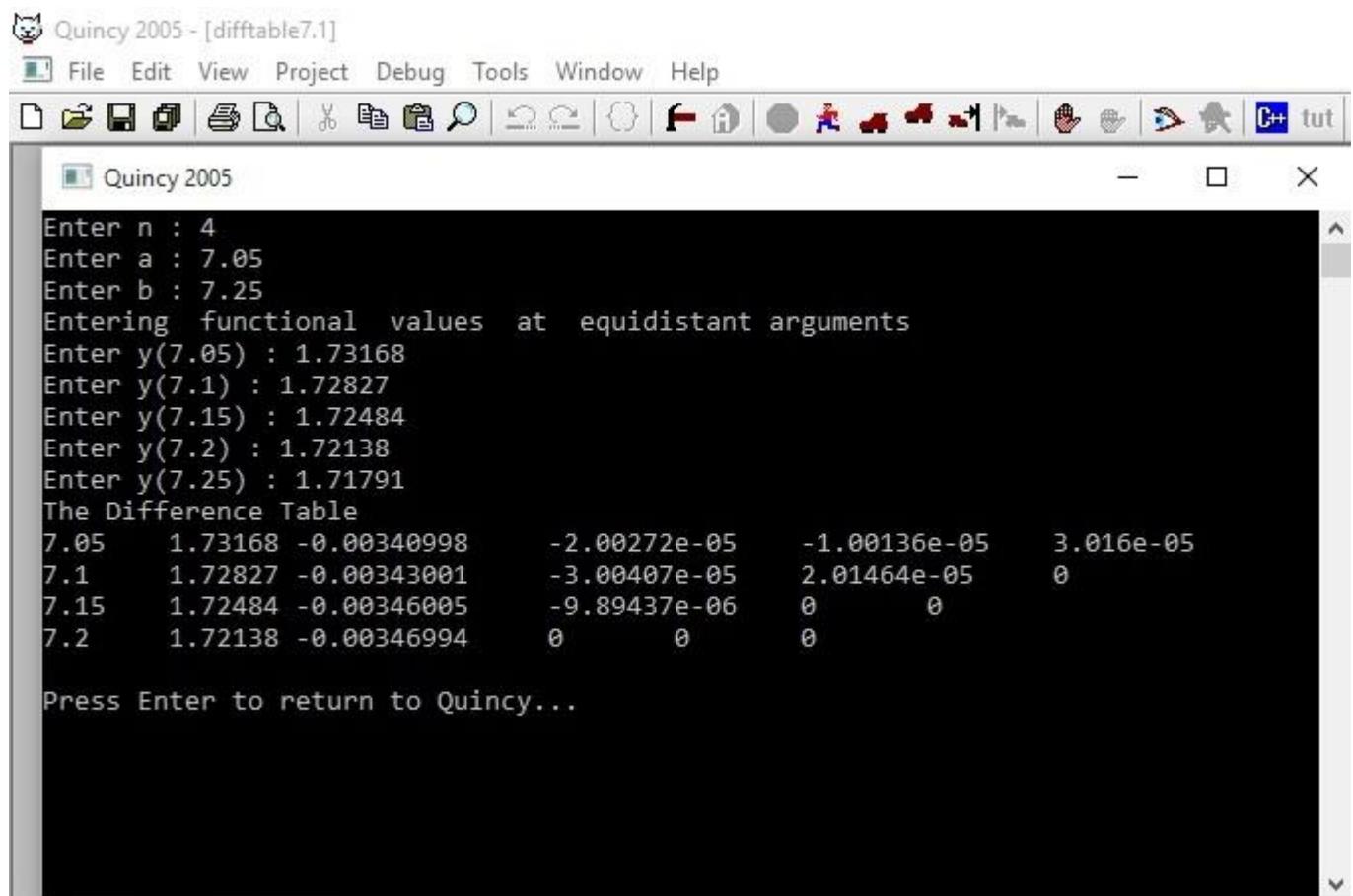
cout<<"Entering functional values at equidistant
arguments"<<endl;
for(int i=0;i<n+1;i++)
{
    cout<<"Enter y("<<(a+i*h)<<") : ";
    cin>>d[0][i];
}

//Difference Table//
for(int i=1;i<n+1;i++)
{
    for(int j=0;j<n;j++)
    {
        if(n+1==i+j)
            d[i][j]=0;
        else
            d[i][j]=d[i-1][j+1]-d[i-1][j];
    }
}
cout<<"The Difference Table"<<endl;

                for(int j=0;j<n;j++)
{
    cout<<(a+j*h)<<"      ";
    for(int i=0;i<n+1;i++)
    {
        cout<<d[i][j]<<"      ";
    }
    cout<<endl;
}
return 0;
}

```

## Out Put:



Quincy 2005 - [difftable7.1]

File Edit View Project Debug Tools Window Help

D E F G H I J K L M N O P Q R S T C+ tut

Quincy 2005

```
Enter n : 4
Enter a : 7.05
Enter b : 7.25
Entering functional values at equidistant arguments
Enter y(7.05) : 1.73168
Enter y(7.1) : 1.72827
Enter y(7.15) : 1.72484
Enter y(7.2) : 1.72138
Enter y(7.25) : 1.71791
The Difference Table
7.05    1.73168 -0.00340998    -2.00272e-05    -1.00136e-05    3.016e-05
7.1     1.72827 -0.00343001    -3.00407e-05    2.01464e-05    0
7.15    1.72484 -0.00346005    -9.89437e-06    0          0
7.2     1.72138 -0.00346994    0          0          0

Press Enter to return to Quincy...
```

***Interpolation***  
***Newton's forward interpolation formula***

**Problem 8.** Write a program in C++ to compute the value of  $f(2.16)$  by Newton's forward interpolation formula:

$x$	$f(x)$
2.15	1.9288
2.20	1.9275
2.25	1.9261
2.30	1.9244
2.35	1.9224
2.40	1.9202

where  $k=1$  . Run your program and write the output. so,

$x$	$f(x)$
2.15	1.9288
2.20	1.9275
2.25	1.9261
2.30	1.9244
2.35	1.9224
2.40	1.9202

```
//Newton's forward interpolation formula
#include<iostream>
using namespace std;
int main()
{
int n;
float d[100][100],h,a,b,x,u,sum,term;
//d[i][j]= Ai[yj],c[i] is coefficient array//
cout<<"Enter n : ";
cin>>n;
cout<<"Newton's forward interpolation :"<<endl;
```

```

cout<<"Enter a : ";
cin>>a;
cout<<"Enter b: ";
cin>>b;

h=fabs(b-a)/n;

cout<<"Entering functional values at equidistant
arguments"<<endl;
for(int i=0;i<n+1;i++)
{
    cout<<"Enter y("<<(a+i*h)<<") : ";
    cin>>d[0][i];
}

//Difference Table//
for(int i=1;i<n+1;i++)
{
    for(int j=0;j<n;j++)
    {
        if(n+1==i+j)
            d[i][j]=0;
        else
            d[i][j]=d[i-1][j+1]-d[i-1][j];
    }
}
cout<<"The Difference Table"<<endl;
for(int j=0;j<n;j++) {
    for(int i=1;i<n+1;i++)
    {
        cout<<d[i][j]<<"      ";
    }
    cout<<endl;
}
cout<<"Enter x : ";
cin>>x;
u=(x-a)/h;
sum=d[0][0];
for(int i=1;i<n+1;i++)
{
    term=d[i][0];
    for(int j=0;j<i;j++)
    {
        term=term*(u-j)/(j+1);
    }
    cout<<"term("<<i<<") : "<<term<<endl;
}

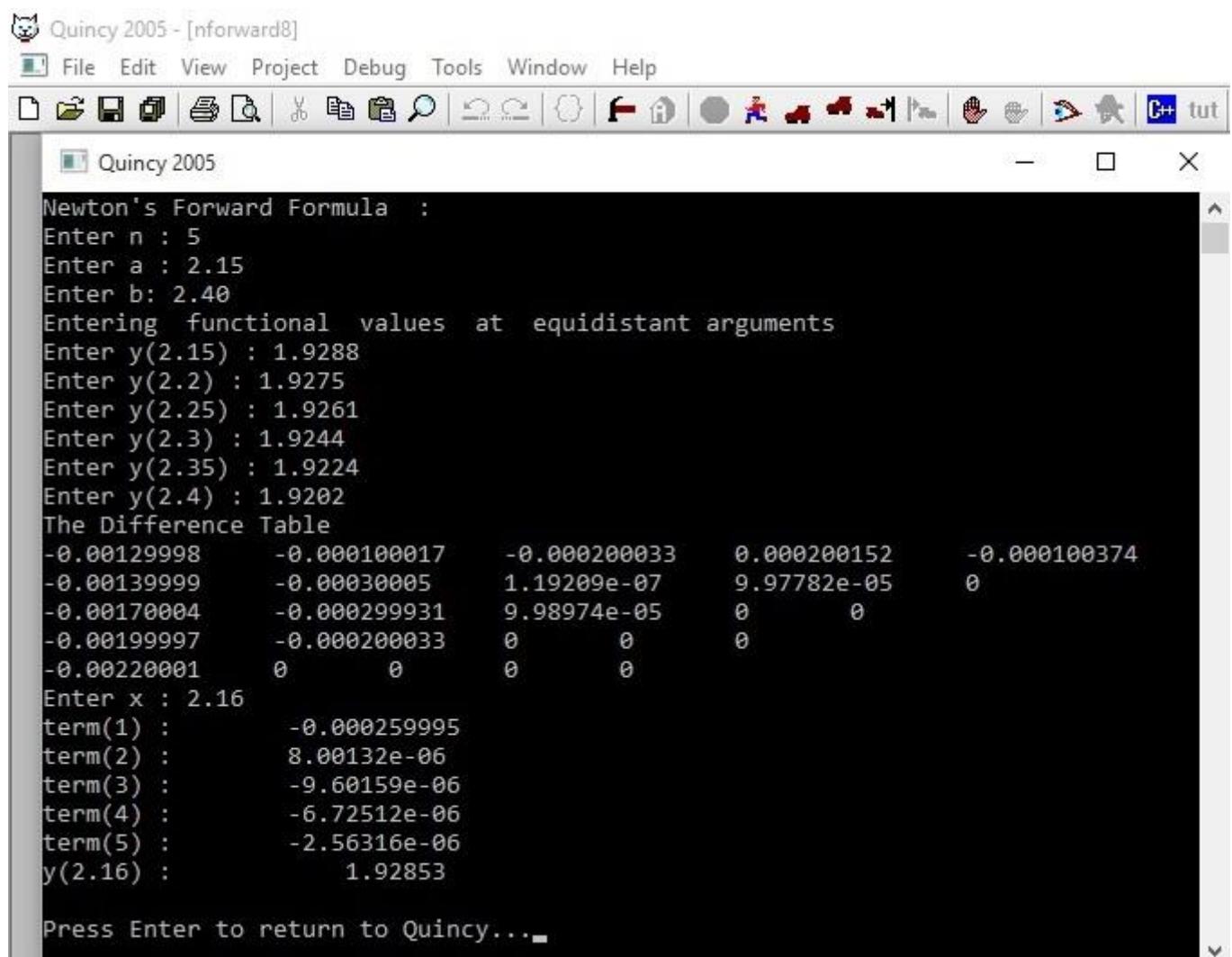
```

```

sum=sum + term;
}
cout<<"y ("<<x<<") : "<<sum<<endl;
return 0;
}

```

## Out Put:



The screenshot shows the Quincy 2005 IDE interface with a window titled "Quincy 2005" displaying the execution of a C++ program. The program calculates the value of a function y at x=2.16 using Newton's Forward Difference formula. The output includes the input values (n=5, a=2.15, b=2.40), the functional values y(2.15) to y(2.4), the difference table, and the calculated terms.

```

Newton's Forward Formula :
Enter n : 5
Enter a : 2.15
Enter b: 2.40
Entering functional values at equidistant arguments
Enter y(2.15) : 1.9288
Enter y(2.2) : 1.9275
Enter y(2.25) : 1.9261
Enter y(2.3) : 1.9244
Enter y(2.35) : 1.9224
Enter y(2.4) : 1.9202
The Difference Table
-0.00129998 -0.000100017 -0.000200033 0.000200152 -0.000100374
-0.00139999 -0.00030005 1.19209e-07 9.97782e-05 0
-0.00170004 -0.000299931 9.98974e-05 0 0
-0.00199997 -0.000200033 0 0 0
-0.00220001 0 0 0 0
Enter x : 2.16
term(1) : -0.000259995
term(2) : 8.00132e-06
term(3) : -9.60159e-06
term(4) : -6.72512e-06
term(5) : -2.56316e-06
y(2.16) : 1.92853

Press Enter to return to Quincy...

```

***Interpolation***  
***Newton's backward interpolation formula***

**Problem 9.** Write a program in C++ to compute the value of  $f(7.68)$  by Newton's backward interpolation formula:

$x$	$f(x)$
7.45	1.53566
7.50	1.52909
7.55	1.52258
7.60	1.51613
7.65	1.50973
7.70	1.50338

where  $k=1$  . Run your program and write the output. So,

$x$	$f(x)$
7.45	1.53566
7.50	1.52909
7.55	1.52258
7.60	1.51613
7.65	1.50973
7.70	1.50338

```
//Newton's Backward Interpolation formula
#include<iostream>
using namespace std;
int main()
{
int n;
float d[100][100],h,a,b,x,u,sum,term;
//d[i][j]= Ai[yj],c[i] is coefficient array//
Cout<<"Newton's Backward interpolation :"<<endl;
cout<<"Enter n : ";
cin>>n;

cout<<"Enter a : ";
cin>>a;
cout<<"Enter b: ";
```

```

cin>>b;

h=fabs (b-a) /n;

cout<<"Entering functional values at equidistant
arguments"<<endl;
for(int i=0;i<n+1;i++)
{
    cout<<"Enter y(" <<(a+i*h) <<") : ";
    cin>>d[0][i];
}

//Difference Table//
for(int i=1;i<n+1;i++)
{
for(int j=0;j<n;j++)
{
    if(n+1==i+j)
        d[i][j]=0;
    else
        d[i][j]=d[i-1][j+1]-d[i-1][j];
}
}

cout<<"The Difference Table"<<endl;
for(int j=0;j<n;j++) {
for(int i=1;i<n+1;i++)
{
    cout<<d[i][j]<< " ";
}
cout<<endl;
}
cout<<"Enter x : ";
cin>>x;

u=(x-b)/h;

sum=d[0][n];
for(int i=1;i<n+1;i++)
{
    term=d[i][n-i];
    for(int j=0;j<i;j++)
    {
        term=term*(u+j)/(j+1);
    }
    cout<<"term("<<i<<") : "<<term<<endl;
    sum=sum + term;
}

```

```
cout<<"y ("<<x<<") : "<<sum<<endl;
return 0;
}
```

## Out Put:

The screenshot shows the Quincy 2005 IDE interface with a menu bar and toolbar. The main window displays the following code output:

```
Newton's Backward interpolation :
Enter n : 5
Enter a : 7.45
Enter b: 7.70
Entering functional values at equidistant arguments
Enter y(7.45) : 1.53566
Enter y(7.5) : 1.52909
Enter y(7.55) : 1.52258
Enter y(7.6) : 1.51613
Enter y(7.65) : 1.50973
Enter y(7.7) : 1.50338
The Difference Table
-0.00656998 5.99623e-05 0 -9.89437e-06 1.96695e-05
-0.00651002 5.99623e-05 -9.89437e-06 9.77516e-06 0
-0.00645006 5.00679e-05 -1.19209e-07 0 0
-0.00639999 4.99487e-05 0 0 0
-0.00635004 0 0 0 0
Enter x : 7.68
term(1) : 0.00254001
term(2) : -5.99384e-06
term(3) : 7.62939e-09
term(4) : -4.06647e-07
term(5) : -5.89142e-07
y(7.68) : 1.50591

Press Enter to return to Quincy...
```

***Interpolation***  
***Lagrange's interpolation formula***

**Problem 10.** Write a program in C++ to compute the value of  $y$  for  $x=8.25$  by Lagrange's interpolation formula:

$x$	8.0	8.1	8.3	8.5	8.6
$y$	1.2364	1.2326	1.2261	1.2210	1.2188

where  $k = 1$ . Run your program and write the output. so,

$x$	8.0	8.1	8.3	8.5	8.6
$y$	1.2364	1.2326	1.2261	1.2210	1.2188

```
//Lagrange's Interpolation formula
#include<iostream>
using namespace std;
int main()
{
    int n;
    float x[100],y[100],sum,term,p;
    cout<<"Lagrange's interpolation :"<<endl;
    cout<<"Enter n : ";
    cin>>n;

    for(int i=0;i<n+1;i++)
    {
        cout<<"Enter x("<<i<<") : ";
        cin>>x[i];

        cout<<"Enter y("<<i<<") : ";
        cin>>y[i];

        cout<<endl;
    }

    cout<<"Enter x : ";
}
```

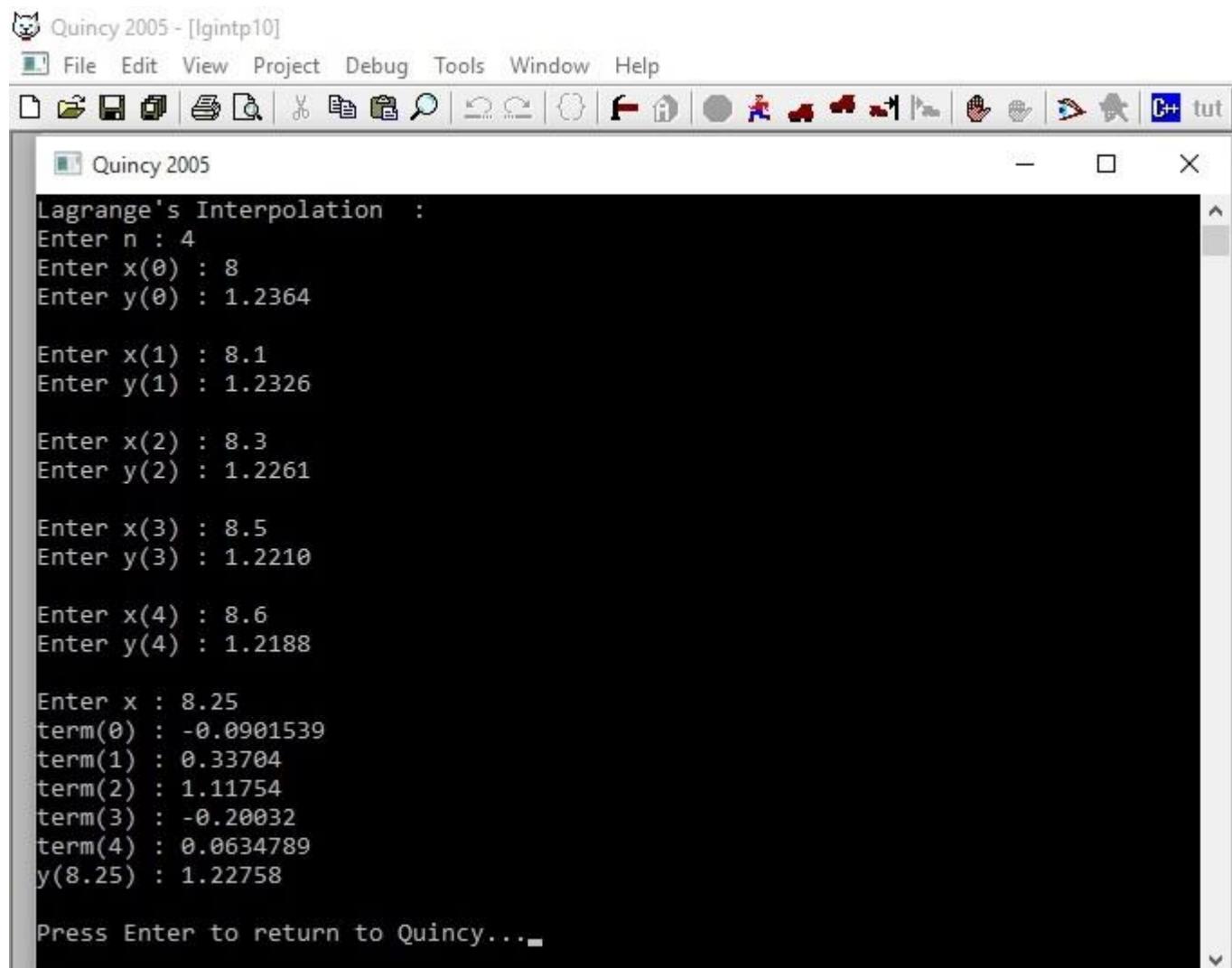
```

    cin>>p;

sum=0;
for(int i=0;i<n+1;i++)
{
    term=y[i];
    for(int j=0;j<n+1;j++)
    {
        if(j!=i)
            term=term*(p-x[j])/(x[i]-x[j]);
    }
    cout<<"term("<<i<<") : "<<term<<endl;
    sum=sum+term;
}
cout<<"y("<<p<<") : "<<sum<<endl;
return 0;
}

```

## Out Put:



The screenshot shows the Quincy 2005 IDE interface. The title bar reads "Quincy 2005 - [lgintp10]". The menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Print, and search. The main window is titled "Quincy 2005" and displays the following text:

```

Lagrange's Interpolation :
Enter n : 4
Enter x(0) : 8
Enter y(0) : 1.2364

Enter x(1) : 8.1
Enter y(1) : 1.2326

Enter x(2) : 8.3
Enter y(2) : 1.2261

Enter x(3) : 8.5
Enter y(3) : 1.2210

Enter x(4) : 8.6
Enter y(4) : 1.2188

Enter x : 8.25
term(0) : -0.0901539
term(1) : 0.33704
term(2) : 1.11754
term(3) : -0.20032
term(4) : 0.0634789
y(8.25) : 1.22758

Press Enter to return to Quincy...

```

## ***Numerical integration by trapezoidal rule***

**Problem 16.** Write a program in C++ to compute the value of the following integral correct to four significant figures by Trapezoidal rule taking 13 ordinates:

$$\int_{20^\circ}^{40^\circ} \frac{\cos 2x}{1 + (k+1)\sin x} dx$$

where  $k = 1$ . Run your program and write the output.

```
//Integration by Trapezoidal-rule
#include<iostream>
#include<cmath>
using namespace std;
float f(float x);
int main()
{
    int n,i=1;
    float a,b,x,y,h,s=0,s3,sum;
    cout<<"Trapezoidal Rule : "<<endl;
    cout<<"give n & a & b : "<<endl;
    cin>>n>>a>>b;
    h=fabs(b-a)/n;
    do
    {
        x=a+i*h;
        s=s+f(x);
        i++;
    }
    while(i<n);
    s3= f(a)+f(b);
    sum=(s3 + 2*s)*(h/3);
    cout<<"THE VALUE OF INTEGRATION IS : "<<fabs(sum)<<endl;
    return 0;
}
float f(float x)
```

```
{  
float pi=2*acos(0.0);  
x=x*pi/180;  
return cos(2*x)/(1+2*sin(x));  
}
```

## Out Put:

Quincy 2005 - [trap11]

File Edit View Project Debug Tools Window Help

Quincy 2005

```
Trapezoidal Rule :  
give n & a & b :  
12  
20  
40  
THE VALUE OF INTEGRATION IS : 3.40252  
Press Enter to return to Quincy...
```

## ***Numerical integration by Simpson's one-third rule***

**Problem 12.** Write a program in C++ to compute the value of the following integral correct to four significant figures by Simpson's one-third rule taking 13 ordinates:

$$\int_{1.1}^{4.4} \frac{\ln(2.2+k+3.3x)}{3.3+2.2x^2} dx$$

where  $k = 1$ . Run your program and write the output.

//Integration by Simpos's Method

```
#include<iostream>
#include<cmath>
using namespace std;
float f(float x);
int main()
{
    int n,i=1,j=2;
    float a,b,x,y,h,s1=0,s2=0,s3,sum;
    cout<<"give n & a & b : "<<endl;
    cin>>n>>a>>b;
    h=fabs(b-a)/(2*n);
    do
    {
        x=a+i*h;
        y=a+j*h;
        s1=s1+f(x);
        s2=s2+f(y);
        i=i+2;
        j=j+2;
    }
    while(i<=2*n-1 && j<=2*n-2);
    s3= f(a)+f(b);
    sum=(s3 + 4*s1 + 2*s2)*(h/3);
    cout<<"THE VALUE OF INTEGRATION IS : "<<fabs(sum)<<endl;
    return 0;
}
```

```
float f(float x)
{
    return (log(3.2 + 3.3*x)) / (3.3 + 2.2*x*x);
}
```

## Out Put:

```
Quincy 2005 - [smson12]
File Edit View Project Debug Tools Window Help
Quincy 2005
Simpson one-third rule :
give n & a & b :
6
1.1
4.4
THE VALUE OF INTEGRATION IS : 0.461601
Press Enter to return to Quincy...
C++ tut
```

***Numerical solution of an equation by  
Newton-Raphson method***

**Problem 13.** Write a program in C++ for finding the smallest positive root of the equation  $e^{a \sin x} + \log_a(1+x) - 2.2 = 0$  correct to five decimal places by Newton-Raphson method, where  $a = \frac{20.1+k}{10}$  and  $k=1$ . Run your program and write the output. So,  $a=2.11$ ;

```
//Newton-Raphson method
#include<iostream>
#include<cmath>
using namespace std;
float f(float x);
float df(float x);
int main()
{
    int n=1;
    float h,x,c;
    cout<<"Newton-Raphson method :"<<endl;
    cout<<"Enter x : ";
    cin>>x;

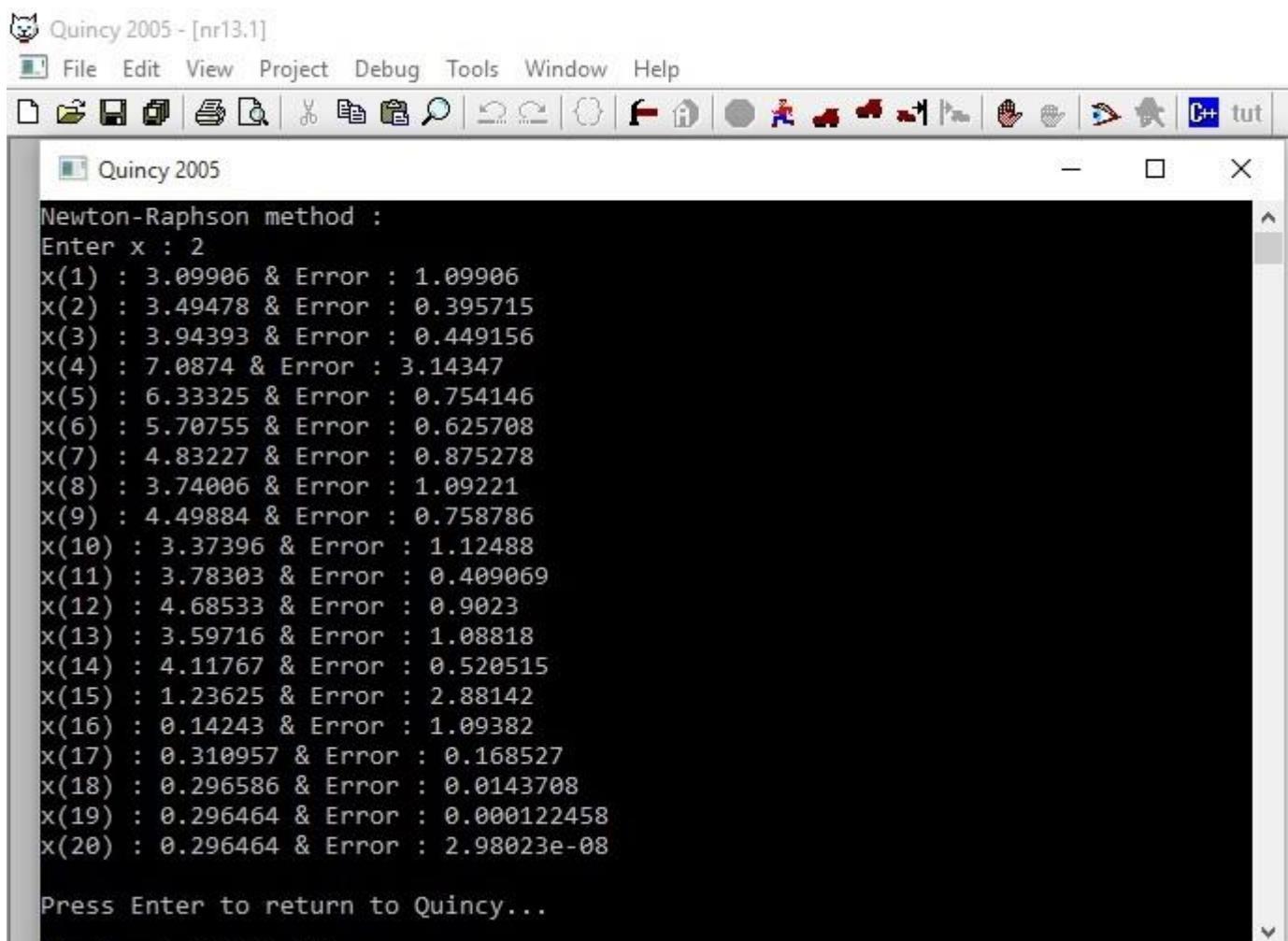
    do
    {
        c=x;
        h=-f(x)/df(x);
        x=x+h;
        cout<<"x ("<<n<<") : "<<x<<" & Error : "<<fabs(x-
c)<<endl;
        n++;
    }
    while(0.00001<fabs(x-c));

    return 0;
}

float f(float x)
```

```
{  
  
float a=(20.1+1)/10;  
    return (exp(a*sin(x)) + (log(1+x)/log(a))-2.2);  
}
```

## Out Put:



The screenshot shows the Quincy 2005 IDE interface. The menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Print, and search. A status bar at the bottom right shows "C++ tut". The main window displays the following text:

```
Quincy 2005 - [nr13.1]  
File Edit View Project Debug Tools Window Help  
Quincy 2005  
Newton-Raphson method :  
Enter x : 2  
x(1) : 3.09906 & Error : 1.09906  
x(2) : 3.49478 & Error : 0.395715  
x(3) : 3.94393 & Error : 0.449156  
x(4) : 7.0874 & Error : 3.14347  
x(5) : 6.33325 & Error : 0.754146  
x(6) : 5.70755 & Error : 0.625708  
x(7) : 4.83227 & Error : 0.875278  
x(8) : 3.74006 & Error : 1.09221  
x(9) : 4.49884 & Error : 0.758786  
x(10) : 3.37396 & Error : 1.12488  
x(11) : 3.78303 & Error : 0.409069  
x(12) : 4.68533 & Error : 0.9023  
x(13) : 3.59716 & Error : 1.08818  
x(14) : 4.11767 & Error : 0.520515  
x(15) : 1.23625 & Error : 2.88142  
x(16) : 0.14243 & Error : 1.09382  
x(17) : 0.310957 & Error : 0.168527  
x(18) : 0.296586 & Error : 0.0143708  
x(19) : 0.296464 & Error : 0.000122458  
x(20) : 0.296464 & Error : 2.98023e-08  
  
Press Enter to return to Quincy...
```

## ***Numerical solution of a system of linear equations by Gauss' elimination method***

**Problem 14.** Write a program in C++ for finding solution of the following system of linear equations correct to four decimal places by using Gauss' elimination method:

$$AX=B,$$

$$A = \begin{pmatrix} 1.162+k & 0.684 & 0.822 & 0.382 \\ 0.785 & 1.192+k & 1.396 & 0.492 \\ 0.735 & 1.602 & 1.108+k & 0.456 \\ 0.443 & 0.573 & 0.338 & 0.899+k \end{pmatrix},$$

where

$$B = [-4.825 \quad 7.106 \quad 7.235 \quad -5.215]^T, \quad X = [x_1 \quad x_2 \quad x_3 \quad x_4]^T \text{ and } k=1.$$

Run your program and write the output.

//Gauss-Elimination Method

```
#include<iostream>
using namespace std;
int main()
{
    int n;
    float Ab[100][100], m[100][100], x[100];

    cout<<"Enter n : "<<endl;
    cin>>n;

    cout<<"Enter Augmented Matrix : "
    " << endl; for(int i=1;i<n+1;i++) {

        for(int j=1;j<n+1;j++)
        {
            cout<<"Enter A"<<i<<j<<" : ";
            cin>>Ab[i][j];
        }
        cout<<"Enter B"<<i<<" : ";
        cin>>Ab[i][n+1];
    }
```

```

//Triangulation...
for(int k=2;k<n+1;k++)
{
cout<<(k-1)<<"th Triangulation..."<<endl<<endl;

for(int i=1;i<n+1;i++)
{

if(k<=i)
{
m[i][k-1]=Ab[i][k-1]/Ab[k-1][k-1];
cout<<"A"<<i<<(k-1)<<" : "<<Ab[i][k-1]<<endl;
cout<<"A"<<(k-1)<<(k-1)<<" : "<<Ab[k-1][k-
1]<<endl;
cout<<"m"<<i<<(k-1)<<" : "<<m[i][k-
1]<<endl; }

for(int j=1;j<n+2;j++)
{

if(k<=i && k<=j+1)
{
        Ab[i][j]=Ab[i][j]-m[i][k-1]*Ab[k-1][j];
}

}

cout<<endl;
}

for(int i=1;i<n+1;i++)
{
for(int j=1;j<n+2;j++)
{
cout<<Ab[i][j]<<" ";
}
cout<<endl;
}
//k loop
}
//Triangulation Complete//

for(int i=n;0<i;i--)
{

x[i]=Ab[i][n+1];
}

```

```

if(i<n)
{
    for(int j=i+1;j<n+1;j++)
    {
        x[i]=x[i]-Ab[i][j]*x[j];
    }
}

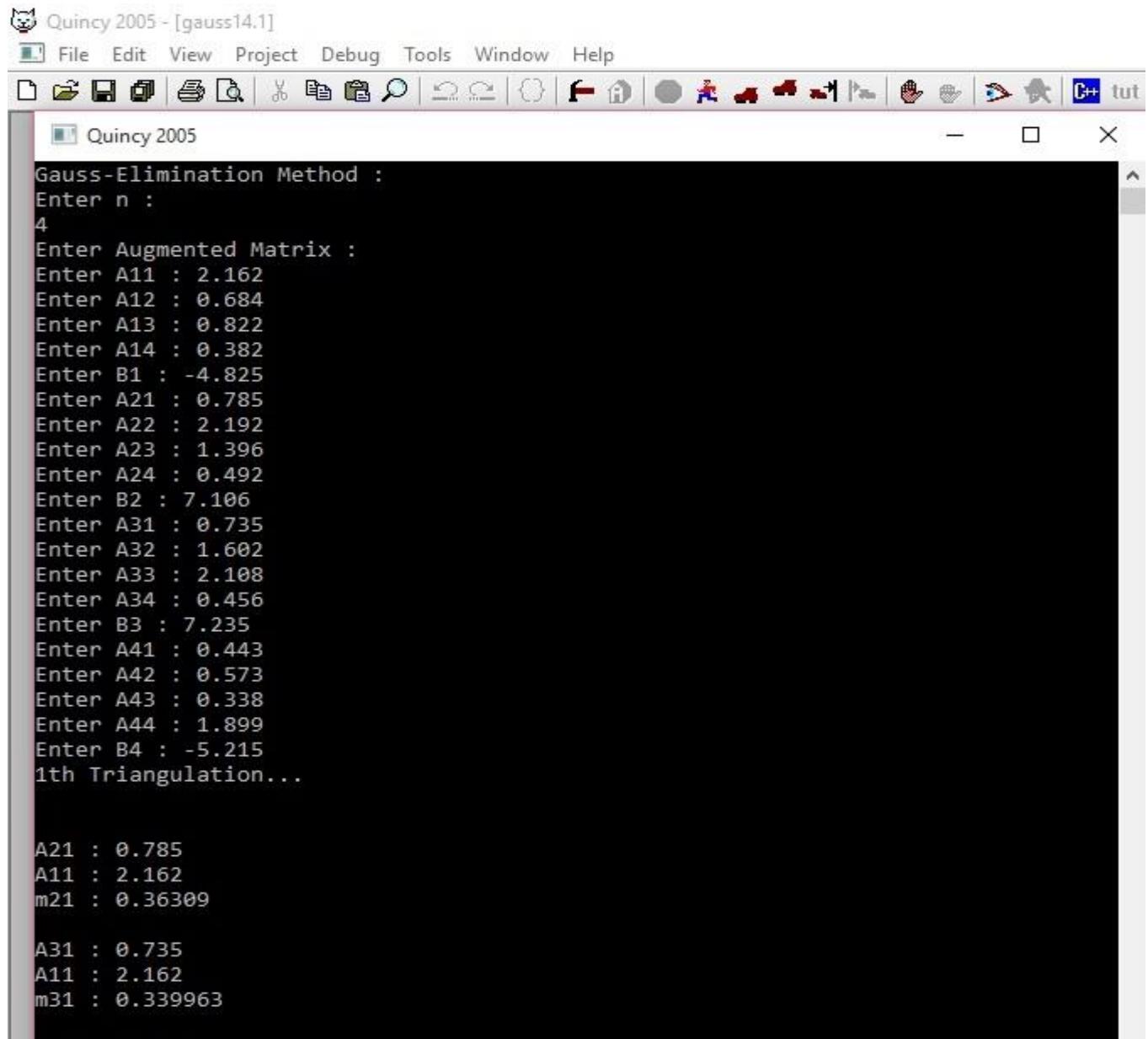
x[i]=x[i]/Ab[i][i]; cout<<"x("<<i<<")"
: "<<x[i]<<endl; }

return 0;
}

```

Page No.: 32

## Out Put:



The screenshot shows the Quincy 2005 IDE interface with the title bar "Quincy 2005 - [gauss14.1]". The menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. Below the menu is a toolbar with various icons. The main window displays the following text:

```

Quincy 2005
Gauss-Elimination Method :
Enter n :
4
Enter Augmented Matrix :
Enter A11 : 2.162
Enter A12 : 0.684
Enter A13 : 0.822
Enter A14 : 0.382
Enter B1 : -4.825
Enter A21 : 0.785
Enter A22 : 2.192
Enter A23 : 1.396
Enter A24 : 0.492
Enter B2 : 7.106
Enter A31 : 0.735
Enter A32 : 1.602
Enter A33 : 2.108
Enter A34 : 0.456
Enter B3 : 7.235
Enter A41 : 0.443
Enter A42 : 0.573
Enter A43 : 0.338
Enter A44 : 1.899
Enter B4 : -5.215
1th Triangulation...

A21 : 0.785
A11 : 2.162
m21 : 0.36309

A31 : 0.735
A11 : 2.162
m31 : 0.339963

```

Quincy 2005 - [gauss14.1]

File Edit View Project Debug Tools Window Help

Quincy 2005

1th Triangulation...

```
A21 : 0.785
A11 : 2.162
m21 : 0.36309

A31 : 0.735
A11 : 2.162
m31 : 0.339963

A41 : 0.443
A11 : 2.162
m41 : 0.204903

2.162 0.684 0.822 0.382 -4.825
2.92173e-08 1.94365 1.09754 0.3533 8.85791
-1.24429e-08 1.36947 1.82855 0.326134 8.87532
-9.31241e-09 0.432846 0.16957 1.82073 -4.22634
2th Triangulation...
```

A32 : 1.36947
A22 : 1.94365
m32 : 0.704586

A42 : 0.432846
A22 : 1.94365
m42 : 0.222698

2.162 0.684 0.822 0.382 -4.825
2.92173e-08 1.94365 1.09754 0.3533 8.85791
-1.24429e-08 -4.5974e-08 1.05524 0.0772042 2.63417
-9.31241e-09 -9.26409e-09 -0.0748503 1.74205 -6.19898
3th Triangulation...

Quincy 2005 - [gauss14.1]

File Edit View Project Debug Tools Window Help

Quincy 2005

```
-9.31241e-09 0.432846 0.16957 1.82073 -4.22634
2th Triangulation...
```

A32 : 1.36947
A22 : 1.94365
m32 : 0.704586

A42 : 0.432846
A22 : 1.94365
m42 : 0.222698

2.162 0.684 0.822 0.382 -4.825
2.92173e-08 1.94365 1.09754 0.3533 8.85791
-1.24429e-08 -4.5974e-08 1.05524 0.0772042 2.63417
-9.31241e-09 -9.26409e-09 -0.0748503 1.74205 -6.19898
3th Triangulation...

A43 : -0.0748503
A33 : 1.05524
m43 : -0.0709321

2.162 0.684 0.822 0.382 -4.825
2.92173e-08 1.94365 1.09754 0.3533 8.85791
-1.24429e-08 -4.5974e-08 1.05524 0.0772042 2.63417
-9.31241e-09 -9.26409e-09 3.44029e-09 1.74752 -6.01214
x(4) : -3.44037
x(3) : 2.74798
x(2) : 3.631
x(1) : -3.8174

Press Enter to return to Quincy....

***Numerical solution of an ordinary differential equation by  
fourth order Runge-Kutta method***

**Problem 15.** Given the differential equation

$$\frac{dy}{dx} = \frac{xy+1}{11y^2+k+2}, \quad y(0)=2,$$

write a program in C++ to tabulate its solution in [0, 0.4] with step length 0.1 correct to five decimal places by the fourth order Runge-Kutta method, where k = 1 . Run your program and write the output.

//Solve ODE by Range\_Kutta Method

```
#include<iostream>
using namespace std;
float f(float x, float y);
int main()
{
    int n;
    float h, x, y, k1, k2, k3, k4, k0;
    cout<<"Enter n : "; cin>>n;

    cout<<"Enter h : ";
    cin>>h;

    cout<<"Enter initial point (x0, y0) : ";
    cin>>x>>y;

    for(int i=0; i<n; i++)
    {
        k1=h*f(x, y);
        k2=h*f(x+0.5*h, y+0.5*k1);
        k3=h*f(x+0.5*h, y+0.5*k2);
        k4=h*f(x+h, y+k3);

        k0=(k1+2*k2+2*k3+k4)/6;

        x=x+h;
        y=y+k0;
    }
}
```

```
cout<<"y ("<<x<<") : "<<y<<endl; }

return 0;
}

float f(float x, float y)
{
float p;

p=(1+x*y) / (11*y*y +3);

return p;
}
```

## Out Put:

The screenshot shows the Quincy 2005 IDE interface. The title bar says "Quincy 2005 - [RK15]". The menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. The toolbar has various icons for file operations like Open, Save, Print, and search. A status bar at the bottom right shows "C++ tut". The main window displays the following text:

```
Quincy 2005

Range-Kutta Method :
Enter n : 5
Enter h : 0.1
Enter initial point (x0,y0): 0 2
y(0.1) : 2
y(0.2) : 2
y(0.3) : 2
y(0.4) : 2
y(0.5) : 2

Press Enter to return to Quincy....
```

***Statistical data analysis  
Sample characteristics***

**Problem 16.** The number of telephone calls received daily in a certain house in Kolkata was recorded for 92 days from 1st May to 31st July 1962, and the following data were obtained.

No. of calls	Frequency
3	2
4	5
5	10
6	8
7	12
8	$12+k$
9	$15-k$
10	11
11	7
12	4
13	4
14	2
<i>Total</i>	92

Write a program in C++ to compute the (i) mean, (ii) median, (iii) mode, (iv) variance, (v) standard deviation, (vi) coefficient of skewness, (vii) coefficient of kurtosis of the sample, where  $k = 1$ . Run your program and write the output.

```
#include<iostream>
#include<cmath>
using namespace std;
int main()
{
```

```

        int n,l;
float d,N,x[100],f[100],F[100],mean,median,Q[3],Sx,mx[4],max;

cout<<"Enter n : ";
cin>>n;

//Taking Data Input
for(int i=1;i<n+1;i++)
{
    cout<<"Enter x("<<i<<") : ";
    cin>>x[i];

    cout<<"Enter f("<<i<<") : ";
    cin>>f[i];

    cout<<endl;
}

//Total freq
N=0;
for(int i=1;i<n+1;i++)
{
    N=N+f[i];
}
//Mean
mean=0;
for(int i=1;i<n+1;i++)
{
    mean=mean+f[i]*x[i];
}
mean=mean/N;
cout<<"Mean : "<<mean<<endl;
//Variance
Sx=0;
for(int i=1;i<n+1;i++)
{
    Sx=Sx+f[i]*(x[i]-mean)*(x[i]-mean);
}
Sx=Sx/N;
cout<<"Variance : "<<Sx<<endl;
Sx=sqrt(Sx);

//Moments about Mean//
```

```

for(int j=1;j<=4;j++)
{
    mx[j]=0;
    for(int i=1;i<n+1;i++)
    {
        mx[j]=mx[j]+f[i]*pow((x[i]-mean),j);
    }
    mx[j]=mx[j]/N;
    cout<<j<<"th Moment about mean : "<<mx[j]<<endl;
}

//Coe of Skewness
cout<<"Coe. of Skewness : "<<(mx[3]/pow(Sx,3))<<endl;
//Coe of Kurtosis
cout<<"Coe. of Kurtosis : "<<(mx[4]/pow(Sx,4))<<endl;
//Coe of Excess
cout<<"Coe. of Excess : "<<((mx[4]/pow(Sx,4))-3)<<endl;

//Comulative frequency
F[1]=f[1];
cout<<"F(1) : "<<F[1]<<endl;
for(int i=2;i<n+1;i++)
{
    F[i]=F[i-1]+f[i]; cout<<"F("<<i<<") : "<<F[i]<<endl;
}
//Quartiles
for(int r=1;r<4;r++)
{
    d=r*(N+1)/4;
    for(int i=1;i<n;i++)
    {
        if(F[i]<= d && d<=F[i+1])
        {
            Q[r]=x[i+1];
        }
    }
    cout<<"Q("<<r<<") : "<<Q[r]<<endl;
}

//Median
cout<<"Median : "<<Q[2]<<endl;
//Quartile Range
cout<<"Quartile Range : "<<(Q[3]-Q[1])<<endl;
//Mode: Corresponding to the Max freq

```

```

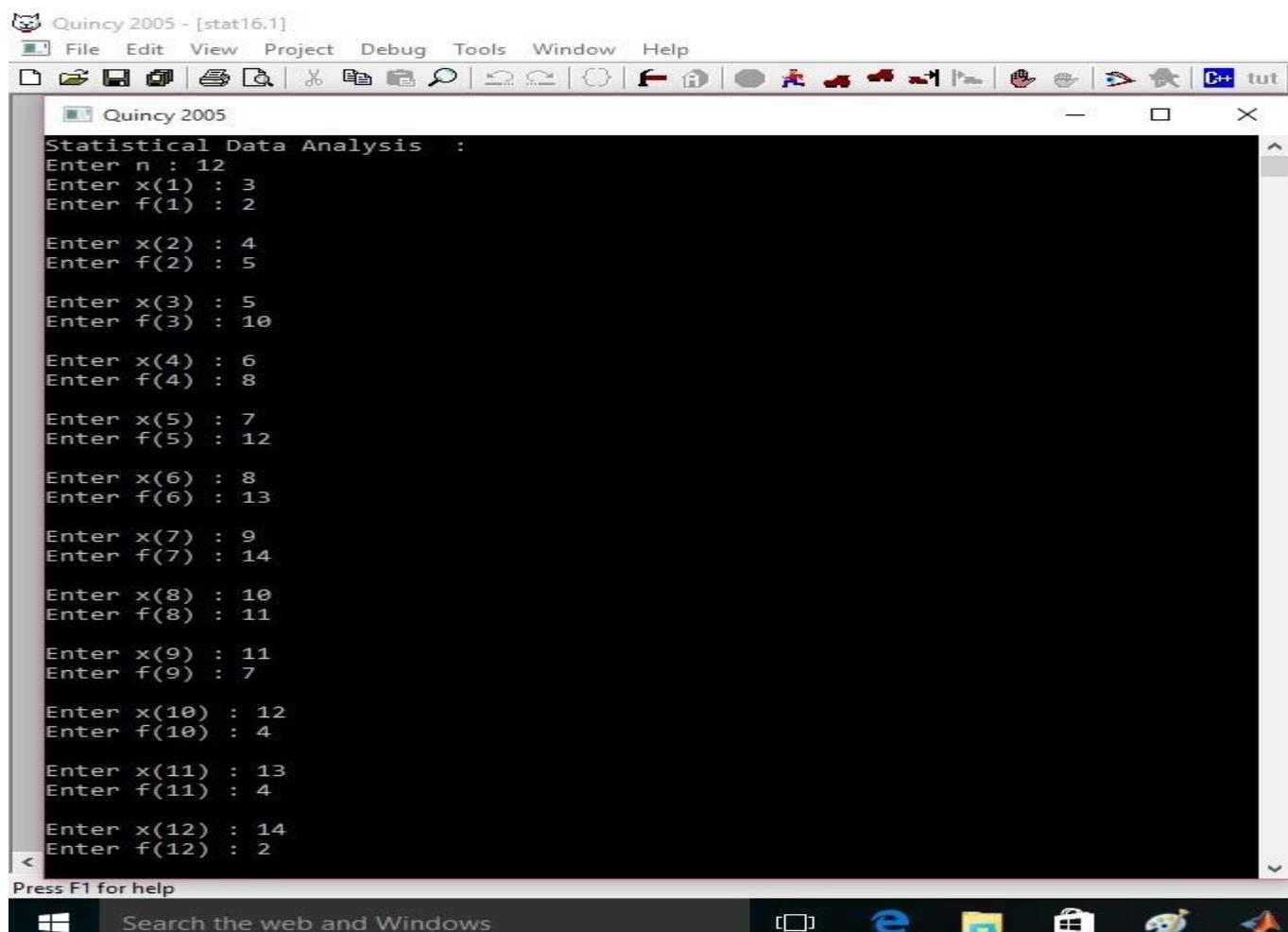
max=f[1];
for(int i=1;i<n;i++)
{
if(f[i]<=f[i+1])
max=f[i+1];
else
max=f[i];
}

for(int i=1;i<n;i++)
{
if(f[i]==max)
{
l=i;
break;
}
}
cout<<"Mode : "<<x[l]<<endl;

return 0;
}

```

## Out Put:



The screenshot shows the Quincy 2005 IDE interface. The title bar reads "Quincy 2005 - [stat16.1]". The menu bar includes File, Edit, View, Project, Debug, Tools, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Print, and Run. The main window has a dark background and displays the following text:

```

Quincy 2005
Statistical Data Analysis :
Enter n : 12
Enter x(1) : 3
Enter f(1) : 2

Enter x(2) : 4
Enter f(2) : 5

Enter x(3) : 5
Enter f(3) : 10

Enter x(4) : 6
Enter f(4) : 8

Enter x(5) : 7
Enter f(5) : 12

Enter x(6) : 8
Enter f(6) : 13

Enter x(7) : 9
Enter f(7) : 14

Enter x(8) : 10
Enter f(8) : 11

Enter x(9) : 11
Enter f(9) : 7

Enter x(10) : 12
Enter f(10) : 4

Enter x(11) : 13
Enter f(11) : 4

Enter x(12) : 14
Enter f(12) : 2

```

At the bottom of the window, it says "Press F1 for help". The Windows taskbar at the bottom shows the Start button, a search bar with "Search the web and Windows", and several pinned icons.

Quincy 2005 - [stat16.1]

File Edit View Project Debug Tools Window Help

Quincy 2005

```
Enter f(7) : 14
Enter x(8) : 10
Enter f(8) : 11
Enter x(9) : 11
Enter f(9) : 7
Enter x(10) : 12
Enter f(10) : 4
Enter x(11) : 13
Enter f(11) : 4
Enter x(12) : 14
Enter f(12) : 2
Mean : 8.18478
Variance : 6.71585
1th Moment about mean : -3.93909e-07
2th Moment about mean : 6.71585
3th Moment about mean : 2.39033
4th Moment about mean : 109.389
Coe. of Skewness : 0.137343
Coe. of Kurtosis : 2.42533
Coe. of Excess : -0.574673
F(1) : 2
F(2) : 7
F(3) : 17
F(4) : 25
F(5) : 37
F(6) : 50
F(7) : 64
F(8) : 75
F(9) : 82
F(10) : 86
F(11) : 90
F(12) : 92
Q(1) : 6
Q(2) : 8
Q(3) : 10
Median : 8
Quartile Range : 4
Mode : 12
Press any key to continue . . . =
```

***Bivariate sample***  
***Correlation coefficient and regression lines***

**Problem 17.** The marks (%) obtained in Numerical Analysis Theoretical and Practical at the M.Sc. examination by 10 candidates are given by the following table:

Theoretical Marks (x)	84	76	72	70+k	64	54	52	48	46	40
Practical Marks (y)	92-k	56	84	94	88	66	90	78	82	30-k

Write a program in C++ to compute the correlation coefficient (between the theoretical and practical marks) and the lines of regression, where  $k = 1$ . Run your program and write the output.

```
#include<iostream>
#include<cmath>
using namespace std;
int main()
{
    int n;
    float x[100], y[100], mx, my, mxy, Sx, Sy, Sxy, rxy, bxy, byx;

    cout<<"Enter n: ";
    cin>>n;

    for(int i=1;i<n+1;i++)
    {
        cout<<"Enter x("<<i<<") : ";
        cin>>x[i];

        cout<<"Enter y("<<i<<") : ";
        cin>>y[i];

        cout<<endl;
    }
    //Mean
    mx=0;
    my=0;
```

```

for(int i=1;i<n+1;i++)
{
    mx=mx+x[i];
    my=my+y[i];
}
mx=mx/n;
my=my/n;
cout<<"Means : m(x)=""<<mx<<" & m(y)=""<<my<<endl;

//Variances
Sx=0;
Sy=0;
Sxy=0;
for(int i=1;i<n+1;i++)
{
    Sx=Sx+(x[i]-mx)*(x[i]-mx);
    Sy=Sy+(y[i]-my)*(y[i]-my);
    Sxy=Sxy+(x[i]-mx)*(y[i]-my);
}
Sx=Sx/n;
Sy=Sy/n;
Sxy=Sxy/n;
cout<<"Variances : var(x)=""<<Sx<<" &
var(y)=""<<Sy<<endl;
Sx=sqrt(Sx);
Sy=sqrt(Sy);
//Correlation Co-efficient
rxy=Sxy/(Sx*Sy);
cout<<"Correlation Co-efficient : "<<rxy<<endl;
//Regression Lines
bxy=rxy*Sx/Sy;
byx=rxy*Sy/Sx;
cout<<"Reg. Co-efficients : bxy=""<<bxy<<" &
byx=""<<byx<<endl;

cout<<"Reg. Line of x on y : "<<endl; cout<<"(X - "
"<<mx<<")=""<<bxy<<" (Y - " <<my<<") "<<endl;

cout<<"Reg. Line of y on x : "<<endl; cout<<"(Y - "
"<<my<<")=""<<byx<<" (X - " <<mx<<") "<<endl;

return 0;
}

```

## Out Put:

```
Quincy 2005
Bi-Variate Data  :
Enter n: 10
Enter x(1) : 84
Enter y(1) : 91

Enter x(2) : 76
Enter y(2) : 56

Enter x(3) : 72
Enter y(3) : 84

Enter x(4) : 71
Enter y(4) : 94

Enter x(5) : 64
Enter y(5) : 88

Enter x(6) : 54
Enter y(6) : 66

Enter x(7) : 52
Enter y(7) : 90

Enter x(8) : 48
Enter y(8) : 78

Enter x(9) : 46
Enter y(9) : 82

Enter x(10) : 40
Enter y(10) : 29

Means : m(x)=60.7 & m(y)=75.8
Variances : var(x)=194.81 & var(y)=370.16
Correlation Co-efficient : 0.43324
Reg. Co-efficients : bxy=0.314297 & byx=0.597197
Reg. Line of x on y :
(X - 60.7)=0.314297(Y - 75.8)
Reg. Line of y on x :
(Y - 75.8)=0.597197(X - 60.7)

< Press Enter to return to Quincy...
Pre
```

# *Programming in MATLAB*

## *Matrices and matrix operations*

**Problem 1.** Let

$$A = \begin{pmatrix} 2.4+k & 3.2 & 2.8 & 1.4 \\ 1.8 & 2.2+k & 2.4 & 1.1 \\ 1.2 & 2.1 & 3.4+k & 1.4 \\ 1.2 & 1.5 & 1.3 & 5.8+k \end{pmatrix},$$

$$B = \begin{pmatrix} 4.4+k & 2.2 & 1.8 & 2.4 \\ 2.8 & 3.2+k & 1.4 & 2.1 \\ 2.2 & 2.2 & 4.4+k & 2.4 \\ 2.2 & 2.5 & 2.3 & 6.8+k \end{pmatrix},$$

where  $k = 1$ . Use MATLAB to answer the following questions.

- (a) Find the transpose of the matrix A.
- (b) Show the diagonal elements of A.
- (c) Determine whether A is singular or not.
- (d) Find the inverse of the matrix A, if it exists.
- (e) Write an identity matrix of order 6.
- (f) Write a  $5 \times 4$  matrix each of whose element is 1.
- (g) Write a zero matrix of order  $5 \times 3$ .
- (h) Find the matrices  $A+B$  and  $AB$ .
- (i) Find the matrix each of whose element is the sine of the corresponding elements of A.

### **OUTPUT COMMAND:**

The screenshot shows the MATLAB interface with the following details:

- Command Window:** Displays the following code and matrix definitions:
 

```
>> A=[3.4 3.2 2.8 1.4;1.8 3.2 2.4 1.1;1.2 2.1 4.4 1.4;1.2 1.5 1.3 6.8]
A =
    3.4000    3.2000    2.8000    1.4000
    1.8000    3.2000    2.4000    1.1000
    1.2000    2.1000    4.4000    1.4000
    1.2000    1.5000    1.3000    6.8000

>> B=[5.4 2.2 1.8 2.4;2.8 4.2 1.4 2.1;2.2 2.2 5.4 2.4;2.2 2.5 2.3 7.8]
B =
    5.4000    2.2000    1.8000    2.4000
    2.8000    4.2000    1.4000    2.1000
    2.2000    2.2000    5.4000    2.4000
    2.2000    2.5000    2.3000    7.8000

>> Transpose_A=A'
Transpose_A =
    3.4000    1.8000    1.2000    1.2000
    3.2000    3.2000    2.1000    1.5000
    2.8000    2.4000    4.4000    1.3000
    1.4000    1.1000    1.4000    6.8000

>> Diagonal_A=diag(A)
Diagonal_A =
    3.4000
    1.8000
    2.8000
    1.4000
```
- Workspace:** Shows variables and their values:
 

Name	Value
A	<4x4 double>
B	<4x4 double>
Determinant_A	89.4850
Diagonal_A	[3.4000;3.2000;4.4000;...]
Each_element_1_...	<5x6 double>
Identity	<6x6 double>
Inverse_A	<4x4 double>
Transpose_A	<4x4 double>
product_AB	<4x4 double>
sine_A	<4x4 double>
sum	<4x4 double>
zero_matrix	<5x3 double>
- Command History:** Displays the command history with the executed code.

MATLAB 7.9.0 (R2009b)

File Edit Debug Parallel Desktop Window Help

Current Folder: C:\Users\DELL\Documents\MATLAB

Shortcuts How to Add What's New

Command Window

```
>> Diagonal_A=diag(A)

Diagonal_A =
    3.4000
    3.2000
    4.4000
    6.8000

>> Determinant_A=det(A)

Determinant_A =
    89.4850

>> Inverse_A=inv(A)

Inverse_A =
    0.6306   -0.5632   -0.0880   -0.0206
   -0.3457    0.8067   -0.2156   -0.0149
    0.0044   -0.2198    0.3557   -0.0386
   -0.0359   -0.0365   -0.0049    0.1614

>> Identity=eye(6)

Identity =
    1     0     0     0     0     0
    0     1     0     0     0     0
    0     0     1     0     0     0
    0     0     0     1     0     0
    0     0     0     0     1     0
    0     0     0     0     0     1
```

Workspace

Name	Value
A	<4x4 double>
B	<4x4 double>
Determinant_A	89.4850
Diagonal_A	[3.4000;3.2000;4.4000;...]
Each_element_1...	<5x4 double>
Identity	<6x6 double>
Inverse_A	<4x4 double>
Transpose_A	<4x4 double>
product_AB	<4x4 double>
sine_A	<4x4 double>
sum	<4x4 double>
zero_matrix	<5x3 double>

Command History

```
clc
A=[3.4 3.2 2.8 1.4;1.8 3.1
B=[5.4 2.2 1.8 2.4;2.8 4.1
Transpose_A=A'
Diagonal_A=diag(A)
Determinant_A=det(A)
Inverse_A=inv(A)
Identity=eye(6)
Each_element_1_matrix=ones(5,4)
zero_matrix=zeros(5,3)
sum=A+B
product_AB=A*B
sine_A=sin(A)
```

MATLAB 7.9.0 (R2009b)

File Edit Debug Parallel Desktop Window Help

Current Folder: C:\Users\DELL\Documents\MATLAB

Shortcuts How to Add What's New

Command Window

```
>> Identity=eye(6)

Identity =
    1     0     0     0     0     0
    0     1     0     0     0     0
    0     0     1     0     0     0
    0     0     0     1     0     0
    0     0     0     0     1     0
    0     0     0     0     0     1

>> Each_element_1_matrix=ones(5,4)

Each_element_1_matrix =
    1     1     1     1
    1     1     1     1
    1     1     1     1
    1     1     1     1
    1     1     1     1

>> zero_matrix=zeros(5,3)

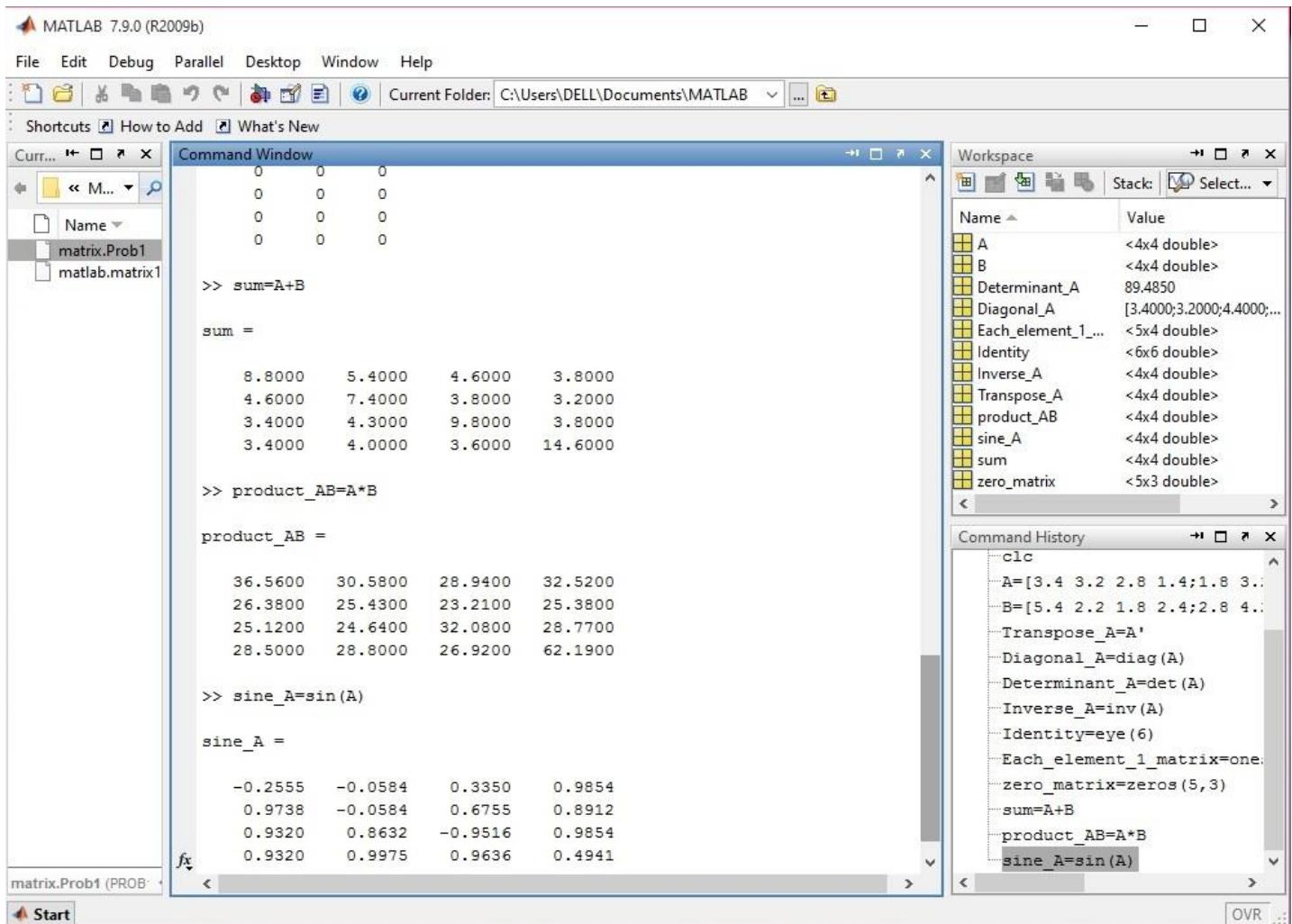
zero_matrix =
    0     0     0
    0     0     0
    0     0     0
    0     0     0
    0     0     0
```

Workspace

Name	Value
A	<4x4 double>
B	<4x4 double>
Determinant_A	89.4850
Diagonal_A	[3.4000;3.2000;4.4000;...]
Each_element_1...	<5x4 double>
Identity	<6x6 double>
Inverse_A	<4x4 double>
Transpose_A	<4x4 double>
product_AB	<4x4 double>
sine_A	<4x4 double>
sum	<4x4 double>
zero_matrix	<5x3 double>

Command History

```
clc
A=[3.4 3.2 2.8 1.4;1.8 3.1
B=[5.4 2.2 1.8 2.4;2.8 4.1
Transpose_A=A'
Diagonal_A=diag(A)
Determinant_A=det(A)
Inverse_A=inv(A)
Identity=eye(6)
Each_element_1_matrix=ones(5,4)
zero_matrix=zeros(5,3)
sum=A+B
product_AB=A*B
sine_A=sin(A)
```



### m FILES :

The screenshot shows the MATLAB Editor window displaying a script file named `matlab.matrix1*`. The code in the editor is:

```

1 A=[3.4 3.2 2.8 1.4;1.8 3.2 2.4 1.1;1.2 2.1 4.4 1.4;1.2 1.5 1.3 6.8]
2 B=[5.4 2.2 1.8 2.4;2.8 4.2 1.4 2.1;2.2 2.2 5.4 2.4;2.2 2.5 2.3 7.8]
3 Transpose_A=A'
4 Diagonal_A=diag(A)
5 Determinant_A=det(A)
6 Inverse_A=inv(A)
7 Identity=eye(6)
8 Each_element_1_matrix=ones(5,4)
9 zero_matrix=zeros(5,3)
10 sum=A+B
11 product_AB=A*B
12 sine_A=sin(A)

```

The status bar at the bottom indicates it is a plain text file with line 12 and column 14.

## *Eigenvalues, eigenvectors and decompositions of matrices*

**Problem 2.** Let

$$A = \begin{pmatrix} 4+k & 3 & 8 & 4 \\ 8 & 2+k & 4 & 1 \\ 2 & 4 & 5+k & 2 \\ 3 & 5 & 3 & 6+k \end{pmatrix},$$

where  $k = 1$ . Use MATLAB to answer the following questions.

- (a) Find the eigenvalues and the corresponding eigenvectors of the matrix  $A$ .
- (b) Find a lower triangular matrix  $L$  and an upper triangular matrix  $U$  such that  $LU = A$ .
- (c) Show the singular value decomposition of  $A$ .

### **OUTPUT COMMAND:**

The screenshot shows the MATLAB interface with the following details:

- File Menu:** File, Edit, Debug, Parallel, Desktop, Window, Help.
- Current Folder:** C:\Users\DELL\Abdul Halim\MATLAE
- Command Window:**

```
>> A=[5 3 8 4;8 10 4 1;2 4 6 2;3 5 3 7]
A =
    5     3     8     4
    8    10     4     1
    2     4     6     2
    3     5     3     7

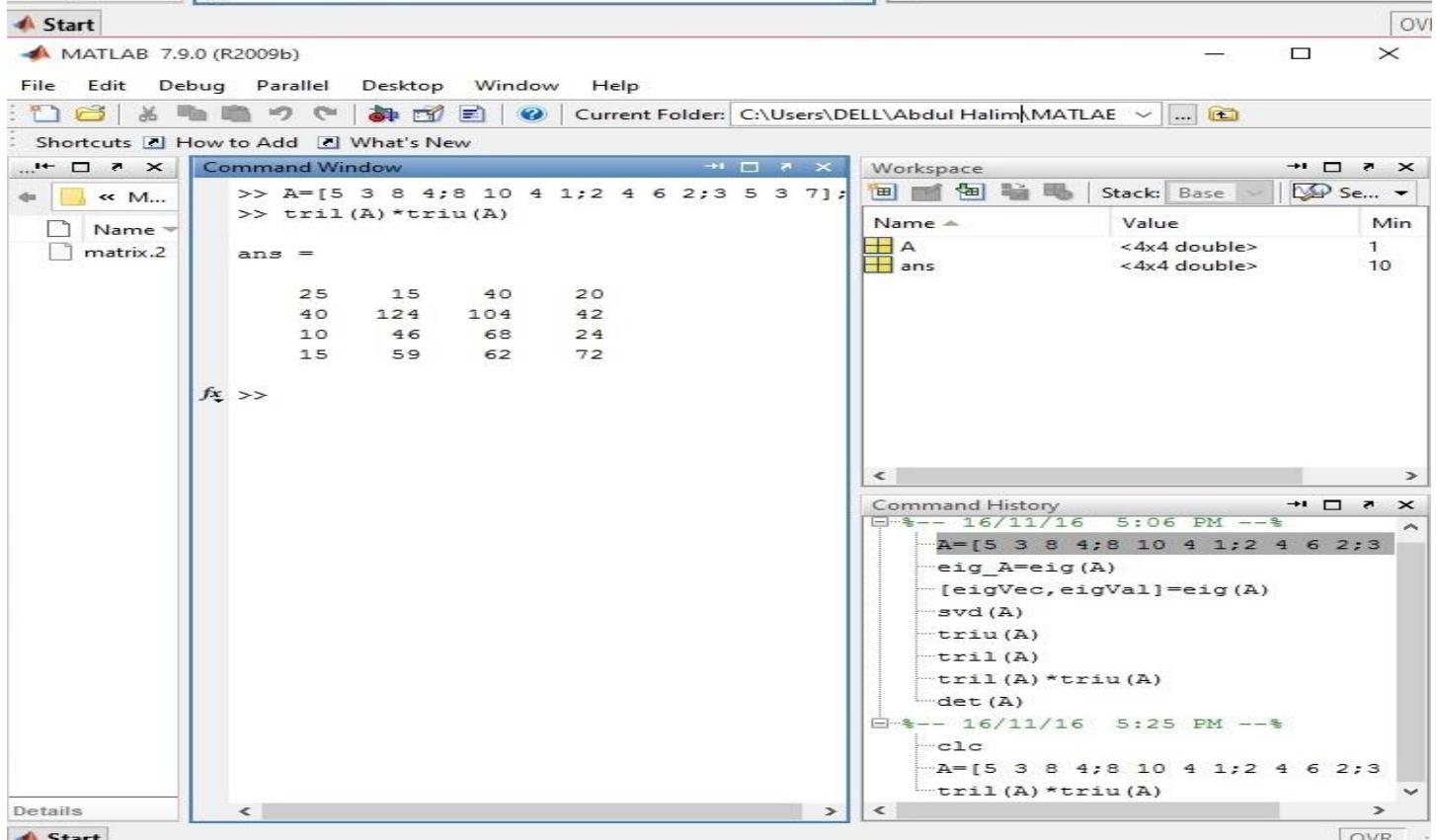
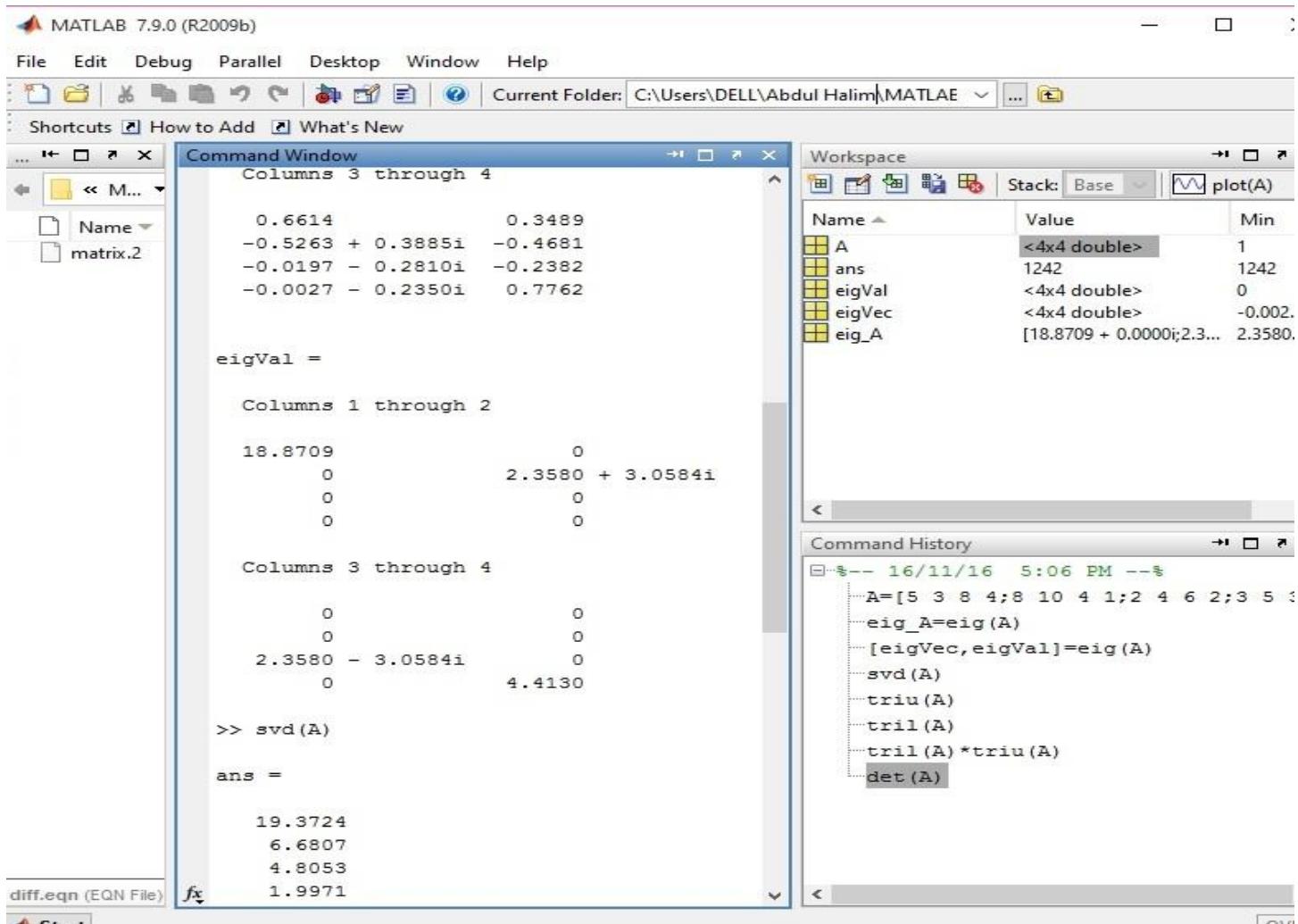
>> eig_A=eig(A)
eig_A =
    18.8709
    2.3580 + 3.0584i
    2.3580 - 3.0584i
    4.4130

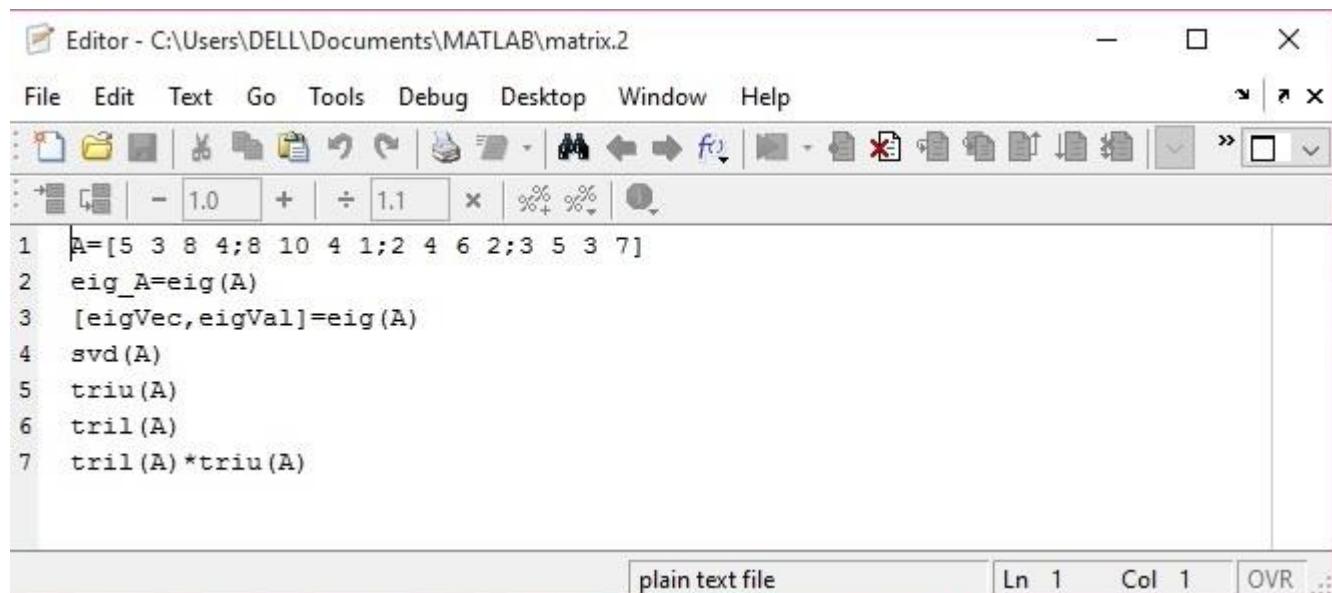
>> [eigVec,eigVal]=eig(A)
eigVec =
    Columns 1 through 2
    -0.4799         0.6614
    -0.6448        -0.5263 - 0.3885i
    -0.3497        -0.0197 + 0.2810i
    -0.4813        -0.0027 + 0.2350i

    Columns 3 through 4
    0.6614         0.3489
```
- Workspace:**

Name	Type	Value	Min
A	<4x4 double>	1	
ans		1242	1242
eigVal	<4x4 double>	0	
eigVec	<4x4 double>	-0.002	
eig_A		[18.8709 + 0.0000i; 2.3... 2.3580]	
- Command History:**

```
16/11/16 5:06 PM --%
A=[5 3 8 4;8 10 4 1;2 4 6 2;3 5 7
eig_A=eig(A)
[eigVec,eigVal]=eig(A)
svd(A)
triu(A)
tril(A)
tril(A)*triu(A)
det(A)
```



**m FILES:**

The screenshot shows the MATLAB Editor window with the title "Editor - C:\Users\DELL\Documents\MATLAB\matrix.2". The menu bar includes File, Edit, Text, Go, Tools, Debug, Desktop, Window, and Help. The toolbar contains various icons for file operations like Open, Save, Print, and Undo/Redo. Below the toolbar is a numeric keypad with buttons for 1.0, +, -, ÷, 1.1, ×, %, and %+. The main editor area contains the following MATLAB code:

```
1 A=[5 3 8 4;8 10 4 1;2 4 6 2;3 5 3 7]
2 eig_A=eig(A)
3 [eigVec,eigVal]=eig(A)
4 svd(A)
5 triu(A)
6 tril(A)
7 tril(A)*triu(A)
```

The status bar at the bottom indicates "plain text file", "Ln 1", "Col 1", and "OVR".

### *Solution of a system of equations*

**Problem 3. (a)** Let

$$A = \begin{pmatrix} 8+k & 3 & 8 & 4 \\ 8 & 7+k & 4 & 1 \\ 2 & 4 & 8+k & 2 \\ 3 & 5 & 3 & 6+k \end{pmatrix},$$

and  $B = (2.3 \quad 3.4 \quad 4.8 \quad 5.6)^T$ ,

where  $k = 11$  (suitable choice). Use MATLAB to solve the system  $AX=B$  by matrix inversion method and Gauss' elimination method.

(b) Solve the equations:  $x^2 + xy + y = 3$ ,  $x^2 - 4x + 3 = 0$ .

#### OUTPUT COMMAND:

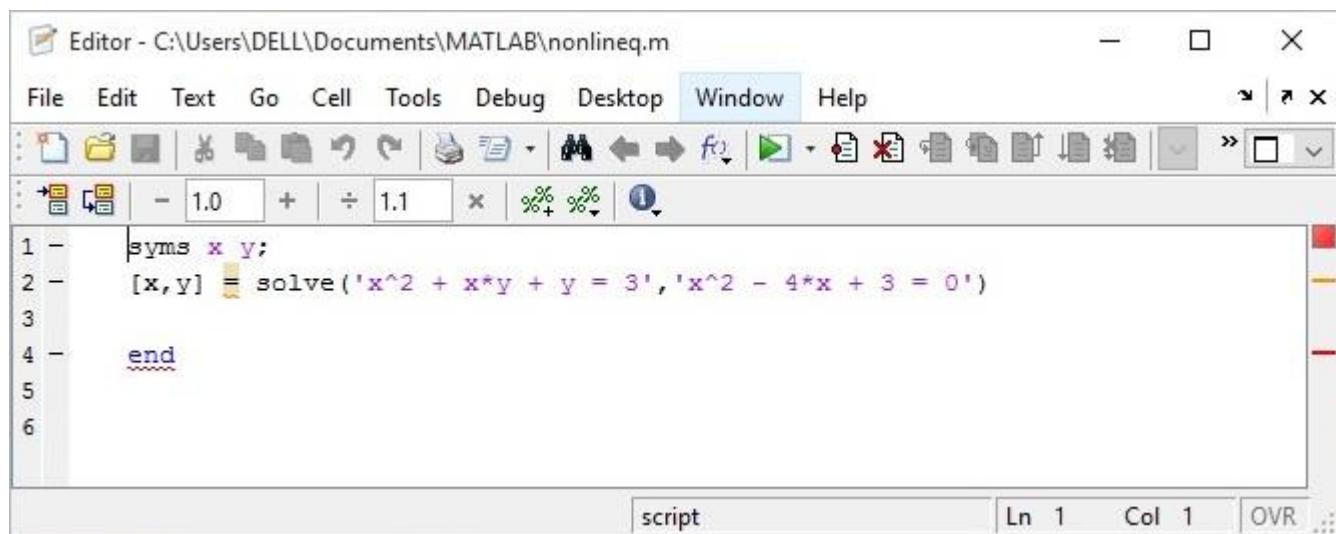
The screenshot shows the MATLAB interface with the following details:

- File Menu:** File, Edit, Debug, Parallel, Desktop, Window, Help.
- Current Folder:** C:\Users\DELL\Abdul Halim\MATLAE
- Command Window:**

```
>> syms x y;
>> [x,y] = solve('x^2 + x*y + y = 3','x^2 - 4*x + 3 = 0')
x =
1
3

y =
1
-3/2
```
- Command History:**

```
-- 13/11/16 12:08 PM --%
clc
syms x y;
[x,y] = solve('x^2 + x*y + y = 3','x^2 - 4*x + 3 = 0')
```
- Navigation:** Curr..., <> M..., Name, nonlineq.m.
- Status Bar:** nonlineq.m (M-file), Start, OVR.

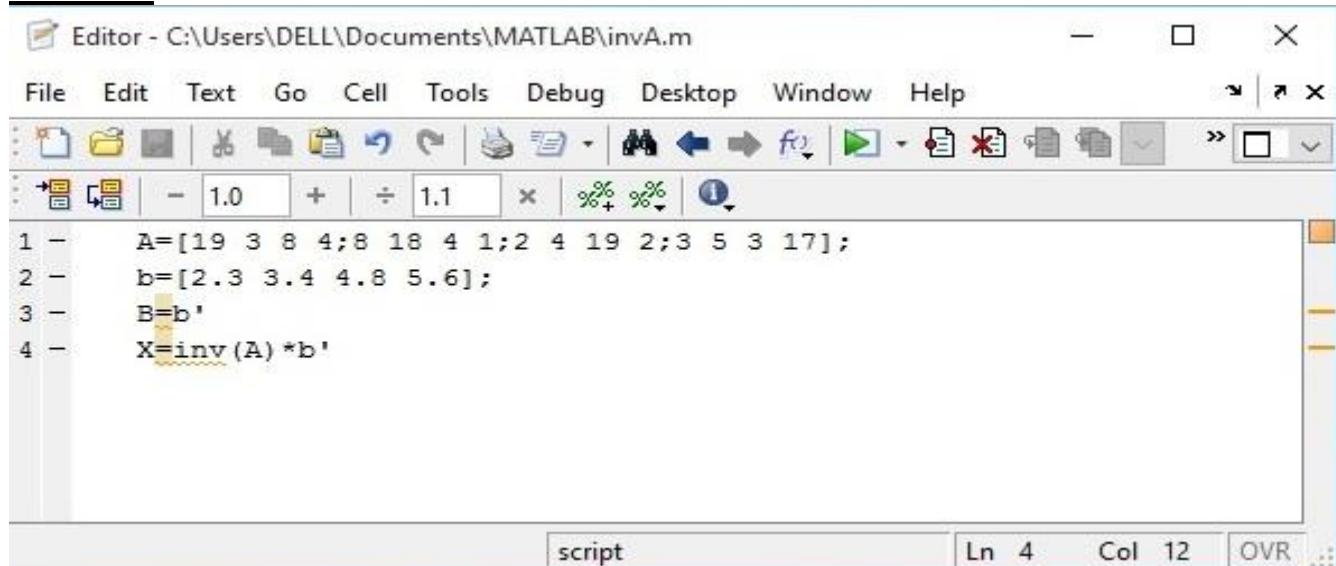
**m FILES:**


```

Editor - C:\Users\DELL\Documents\MATLAB\nonlineq.m

File Edit Text Go Cell Tools Debug Desktop Window Help
script Ln 1 Col 1 OVR ...
1 - syms x y;
2 - [x,y] = solve('x^2 + x*y + y = 3', 'x^2 - 4*x + 3 = 0')
3
4 - end
5
6

```

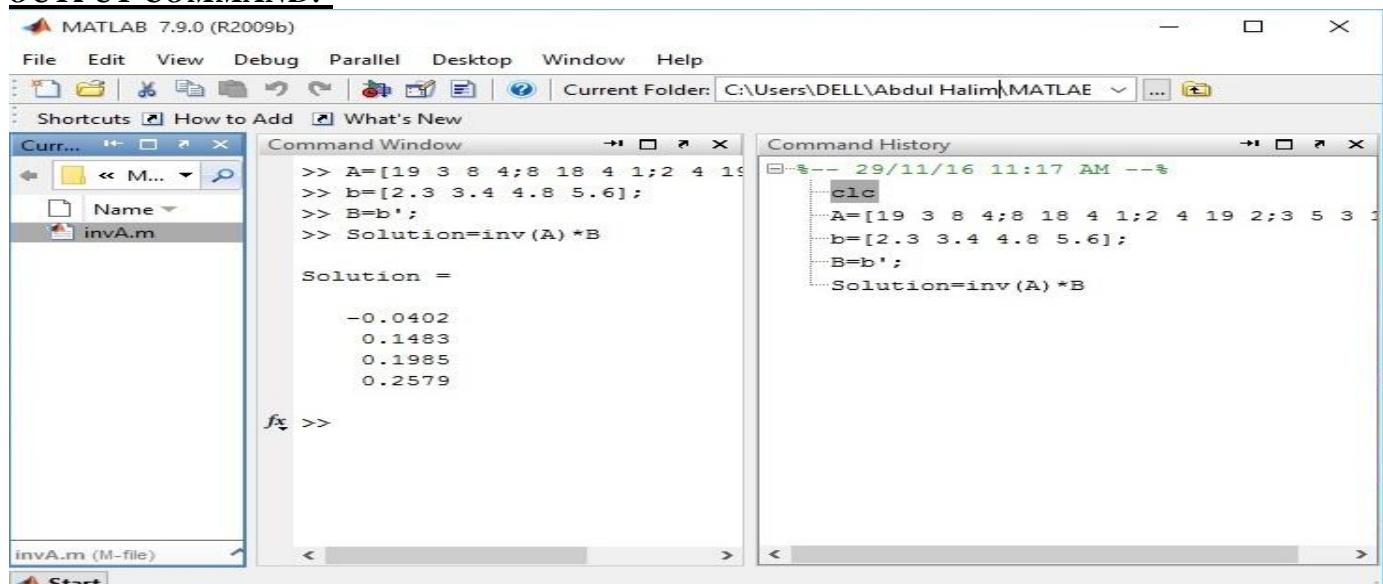
**m FILES:**


```

Editor - C:\Users\DELL\Documents\MATLAB\invA.m

File Edit Text Go Cell Tools Debug Desktop Window Help
script Ln 4 Col 12 OVR ...
1 - A=[19 3 8 4;8 18 4 1;2 4 19 2;3 5 3 17];
2 - b=[2.3 3.4 4.8 5.6];
3 - B=b';
4 - X=inv(A) *b'

```

**OUTPUT COMMAND:**


```

MATLAB 7.9.0 (R2009b)
File Edit View Debug Parallel Desktop Window Help
Current Folder: C:\Users\DELL\Abdul Halim\MATLAE ...
Shortcuts How to Add What's New
Command Window
>> A=[19 3 8 4;8 18 4 1;2 4 19 2;3 5 3 17];
>> b=[2.3 3.4 4.8 5.6];
>> B=b';
>> Solution=inv(A) *B
Solution =
-0.0402
0.1483
0.1985
0.2579
>>
Command History
-- 29/11/16 11:17 AM --
clc
A=[19 3 8 4;8 18 4 1;2 4 19 2;3 5 3 17]
b=[2.3 3.4 4.8 5.6];
B=b';
Solution=inv(A) *B

```

## m FILES:

The screenshot shows the MATLAB Editor window with the file `gauss-elimination.m` open. The code implements the Gauss elimination method to solve a system of linear equations represented by matrix `a`. It first checks if the diagonal element `a(j,j)` is zero and swaps rows if necessary. Then it performs row operations to eliminate the elements below the diagonal. Finally, it uses back-substitution to find the solution vector `x`.

```
1 - a=[19 3 8 4 2.3;8 18 4 1 3.4;2 4 19 2 4.8;3 5 3 17 5.6]
2 %Gauss elimination method [m,n]=size(a);
3 [m,n]=size(a);
4 for j=1:m-1
5     for z=2:m
6         if a(j,j)==0
7             t=a(j,:);a(j,:)=a(z,:);
8             a(z,:)=t;
9         end
10    end
11    for i=j+1:m
12        a(i,:)=a(i,:)-a(j,:)*(a(i,j)/a(j,j));
13    end
14 end
15 x=zeros(1,m);
16 for s=m:-1:1
17     c=0;
18     for k=2:m
19         c=c+a(s,k)*x(k);
20     end
21     x(s)=(a(s,n)-c)/a(s,s);
22 end
23 disp('Gauss elimination method:');
24 a
25 x'
```

## OUTPUT COMMAND:

The screenshot shows the MATLAB Command Window and History window. The user runs the script `gauss-elimination.m`. The output shows the matrix `a` and the solution vector `x'`.

Current Folder: C:\Users\DELL\Abdul Halim\MATLABE

Command Window:

```
>> disp('Gauss elimination method:');
Gauss elimination method:
>> a
a =
Columns 1 through 4
19.0000 3.0000 8.0000 4.0000
0 16.7368 0.6316 -0.6842
0 0 18.0189 1.7296
0 0 0 16.4031

Column 5
2.3000
2.4316
4.0226
4.2296

>> x'
ans =
-0.0402
0.1483
0.1985
0.2579
```

Command History:

```
%-- 14/11/16 9:45 AM --%
clc
a=[19 3 8 4 2.3;8 18 4 1 3.4;2 4 19 2 4.8;3 5 3 17 5.6]
%Gauss elimination method [m,n]=size(a);
[m,n]=size(a);
for j=1:m-1
for z=2:m
if a(j,j)==0
t=a(j,:);a(j,:)=a(z,:);
a(z,:)=t;
end
end
for i=j+1:m
a(i,:)=a(i,:)-a(j,:)*(a(i,j)/a(j,j));
end
end
x=zeros(1,m);
for s=m:-1:1
c=0;
for k=2:m
c=c+a(s,k)*x(k);
end
x(s)=(a(s,n)-c)/a(s,s);
end
disp('Gauss elimination method:');
a
x'
```

## ***Conditional structure, User-defined functions, and Loop structure***

**Problem 4. (a)** Use if-elseif-else statement to compute the the values of the following real valued function for  $x=1(1)8$ :

$$f(x)=\begin{cases} 8, & x<2 \\ \sin x, & 2 < x < 4 \\ \log x, & x \geq 4 \end{cases}$$

**(b)** Find the first 14 Fibonacci numbers by using (i) for loop and (ii) while loop.

### **OUTPUT COMMAND:**

The screenshot shows the MATLAB 7.9.0 (R2009b) desktop environment. The Command Window displays the following code and output:

```

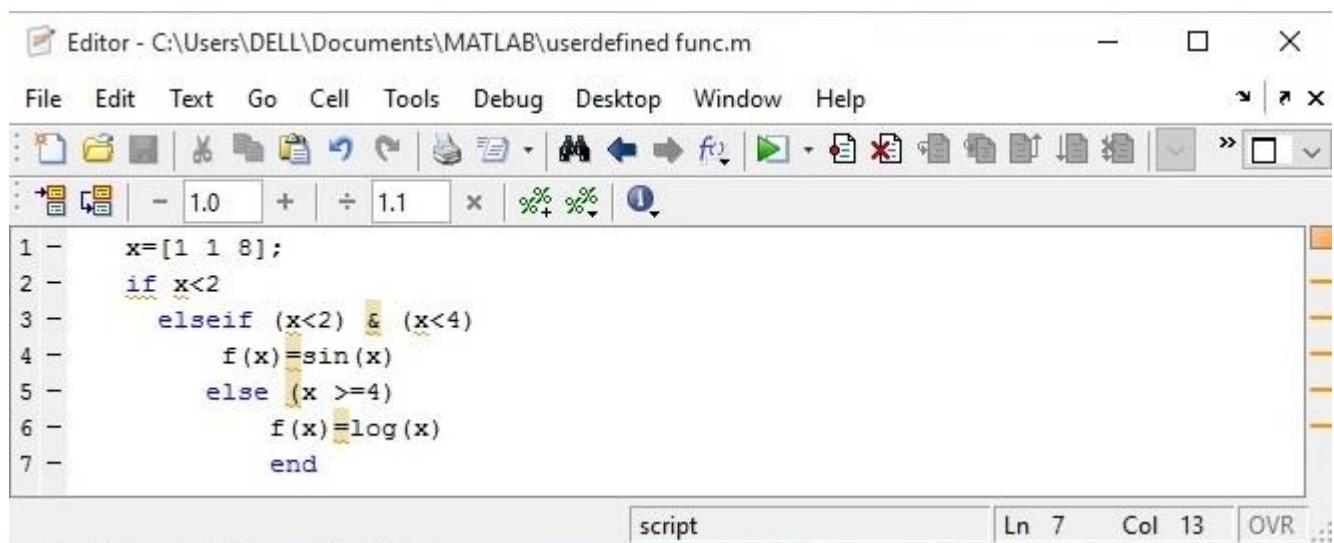
>> x=[1 1 8];
>> if x<2
f(x)=8
elseif (x<2) & (x<4)
f(x)=sin(x)
else (x >=4)
f(x)=log(x)
end

ans =
    0      0      1

f =
    Columns 1 through 4
    0      0      0      0
    Columns 5 through 8
    0      0      0     2.0794

```

The Current Folder browser shows a file named "userdefined func.m". The Command History window shows the same code as the Command Window.

m FILES:


The screenshot shows the MATLAB Editor window with the following code in a script named 'userdefined func.m':

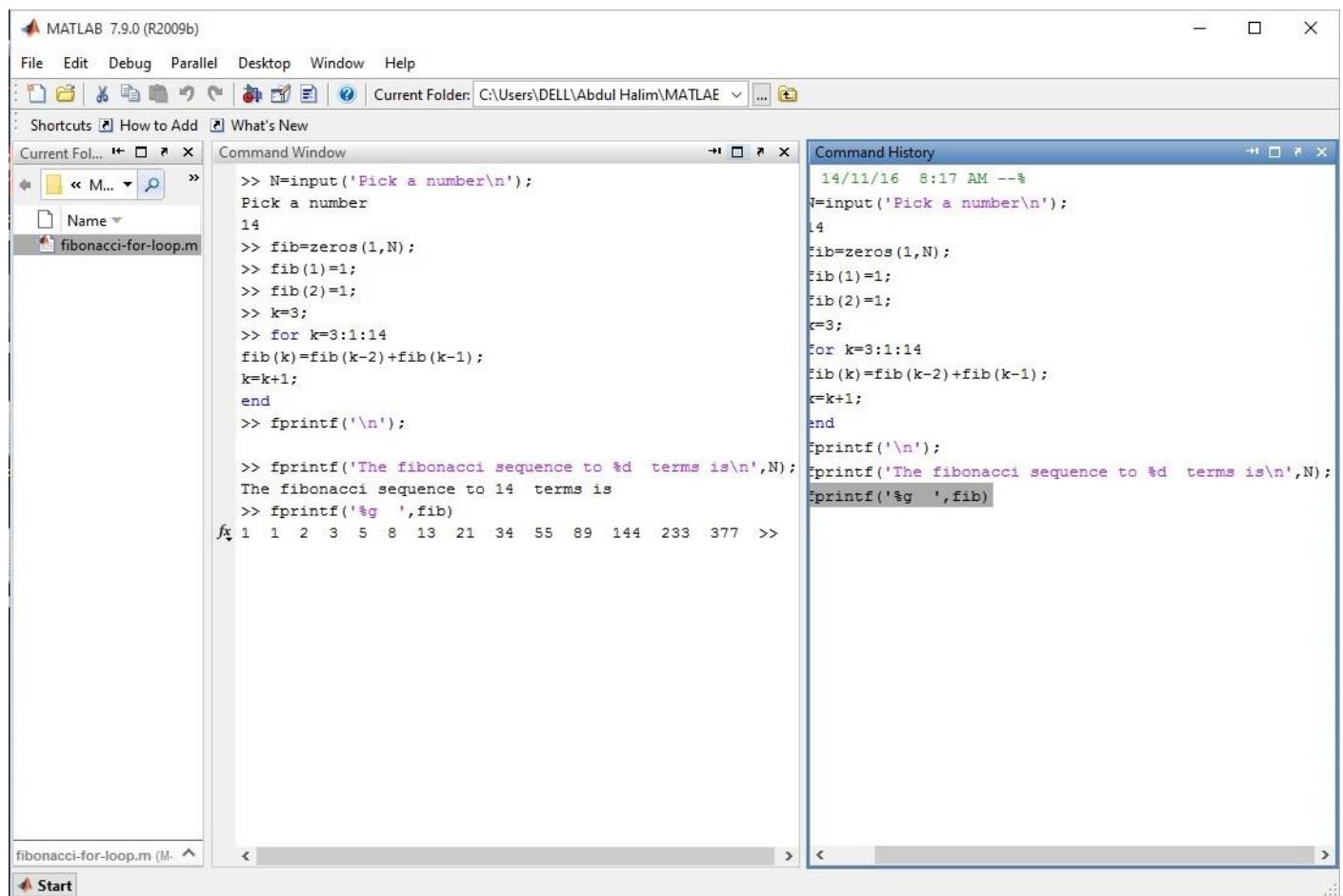
```

1 - x=[1 1 8];
2 - if x<2
3 - elseif (x<2) & (x<4)
4 -     f(x)=sin(x)
5 - else (x >=4)
6 -     f(x)=log(x)
7 - end

```

The status bar at the bottom indicates 'script' and 'Ln 7 Col 13 OVR'.

## (i) For loop



The screenshot shows the MATLAB Command Window and History window. The Command Window displays the following code and its execution:

```

>> N=input('Pick a number\n');
Pick a number
14
>> fib=zeros(1,N);
>> fib(1)=1;
>> fib(2)=1;
>> k=3;
>> for k=3:1:14
fib(k)=fib(k-2)+fib(k-1);
k=k+1;
end
>> fprintf('\n');

>> fprintf('The fibonacci sequence to %d terms is\n',N);
The fibonacci sequence to 14 terms is
>> fprintf('%g ',fib)

```

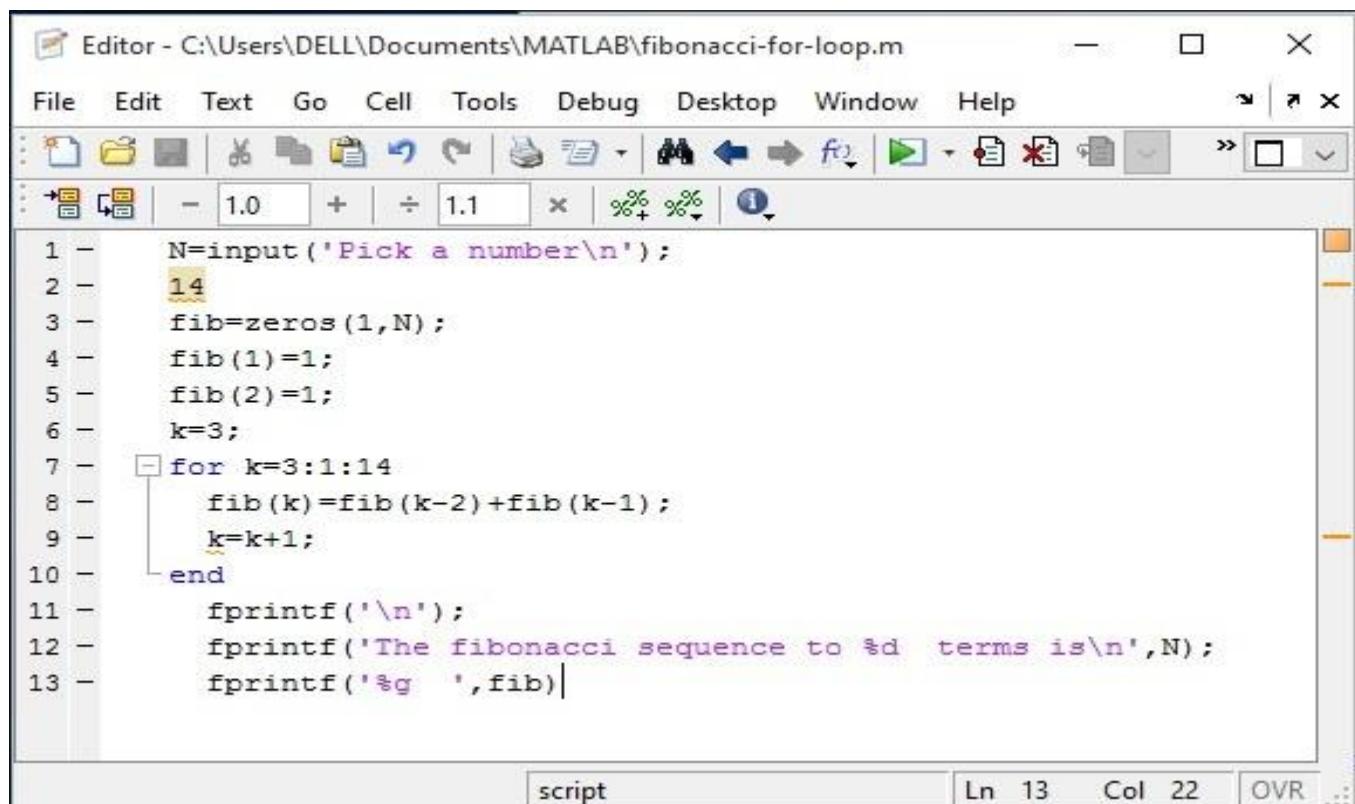
The output in the Command Window is:

```

1 1 2 3 5 8 13 21 34 55 89 144 233 377

```

The History window shows the same commands and their execution times.

m FILES:


```

Editor - C:\Users\DELL\Documents\MATLAB\fibonacci-for-loop.m

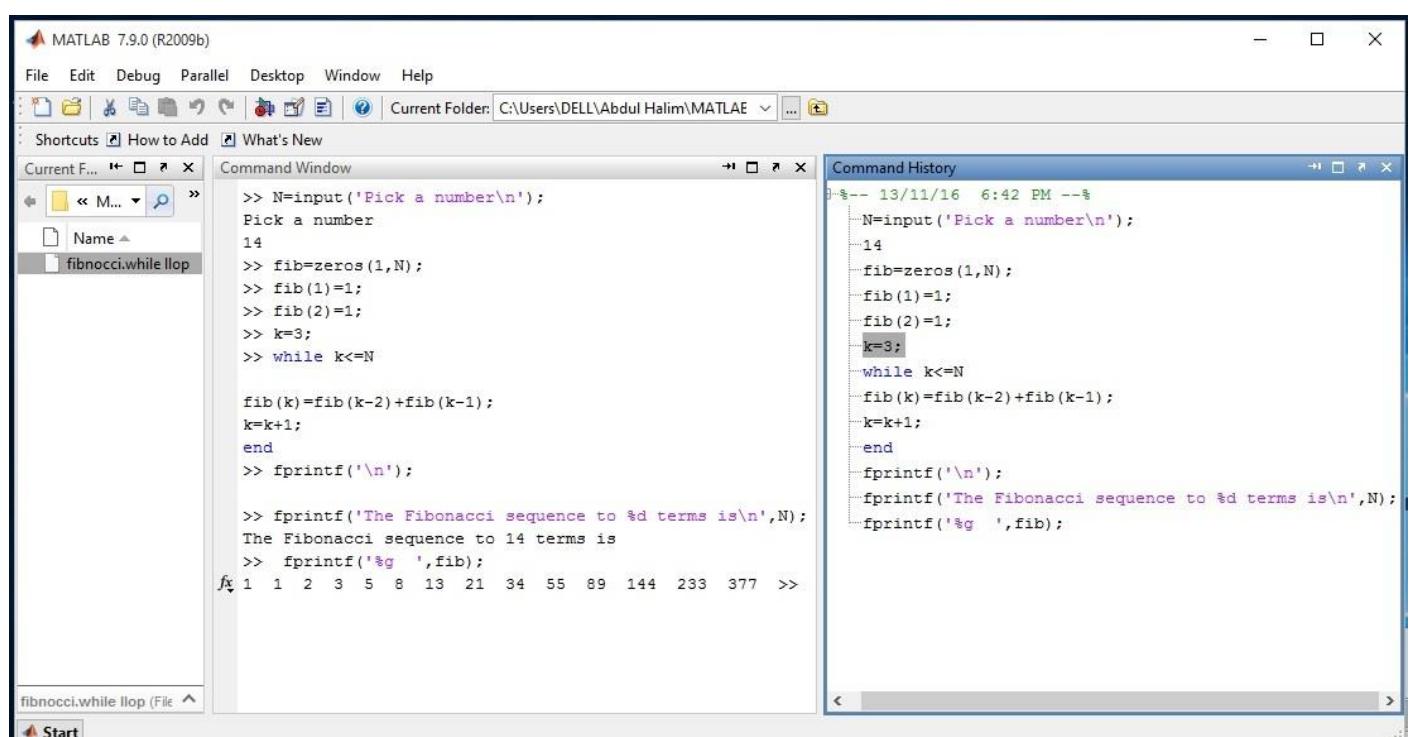
File Edit Text Go Cell Tools Debug Desktop Window Help
New Open Save All Save As Find Run Stop Close
Run Cell Run All Stop All Run Script Run Function Run Test
1 - N=input('Pick a number\n');
2 - 14
3 - fib=zeros(1,N);
4 - fib(1)=1;
5 - fib(2)=1;
6 - k=3;
7 - for k=3:1:14
8 -     fib(k)=fib(k-2)+fib(k-1);
9 -     k=k+1;
10 - end
11 - fprintf('\n');
12 - fprintf('The fibonacci sequence to %d terms is\n',N);
13 - fprintf('%g ',fib)

```

script

Ln 13 Col 22 OVR

## (ii) For While loop



MATLAB 7.9.0 (R2009b)

File Edit Debug Parallel Desktop Window Help

Shortcuts How to Add What's New

Current F... <> M... >>

Name  fibonacci.while llop

Command Window

```

>> N=input('Pick a number\n');
Pick a number
14
>> fib=zeros(1,N);
>> fib(1)=1;
>> fib(2)=1;
>> k=3;
>> while k<=N

    fib(k)=fib(k-2)+fib(k-1);
    k=k+1;
end
>> fprintf('\n');

>> fprintf('The Fibonacci sequence to %d terms is\n',N);
The Fibonacci sequence to 14 terms is
>> fprintf('%g ',fib);

```

fibonacci.while llop (File)

Start

Command History

```

-- 13/11/16 6:42 PM --
N=input('Pick a number\n');
14
fib=zeros(1,N);
fib(1)=1;
fib(2)=1;
k=3;
while k<=N
fib(k)=fib(k-2)+fib(k-1);
k=k+1;
end
fprintf('\n');
fprintf('The Fibonacci sequence to %d terms is\n',N);
fprintf('%g ',fib);

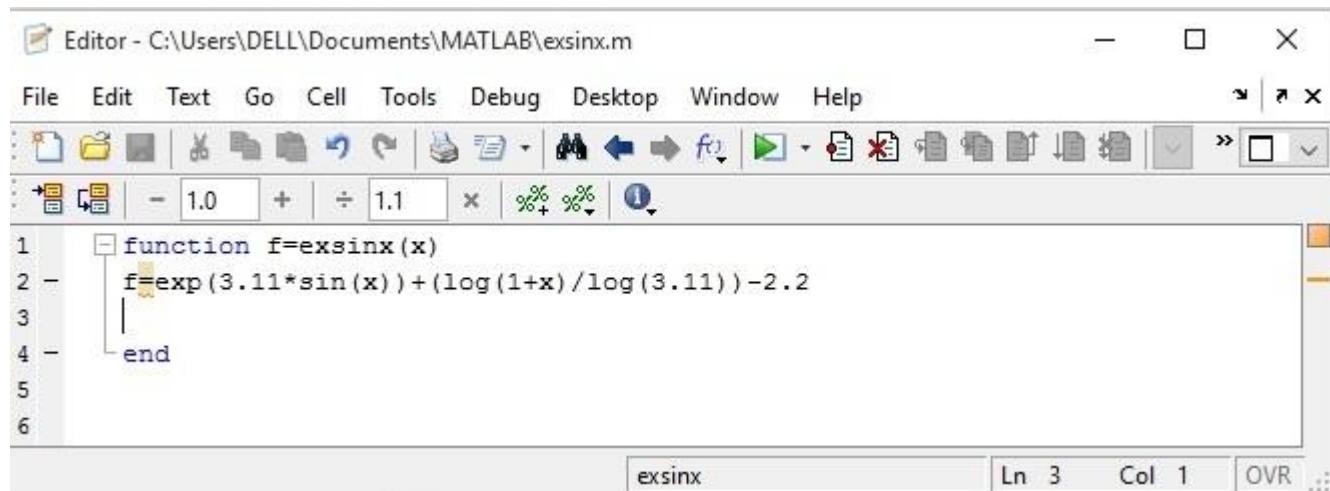
```

## *Numerical integration*

$$\int_{-2}^3 e^{-x^2} \sin x \, dx.$$

**Problem 5.** Use MATLAB to find

**m FILES (FUNCTION):**



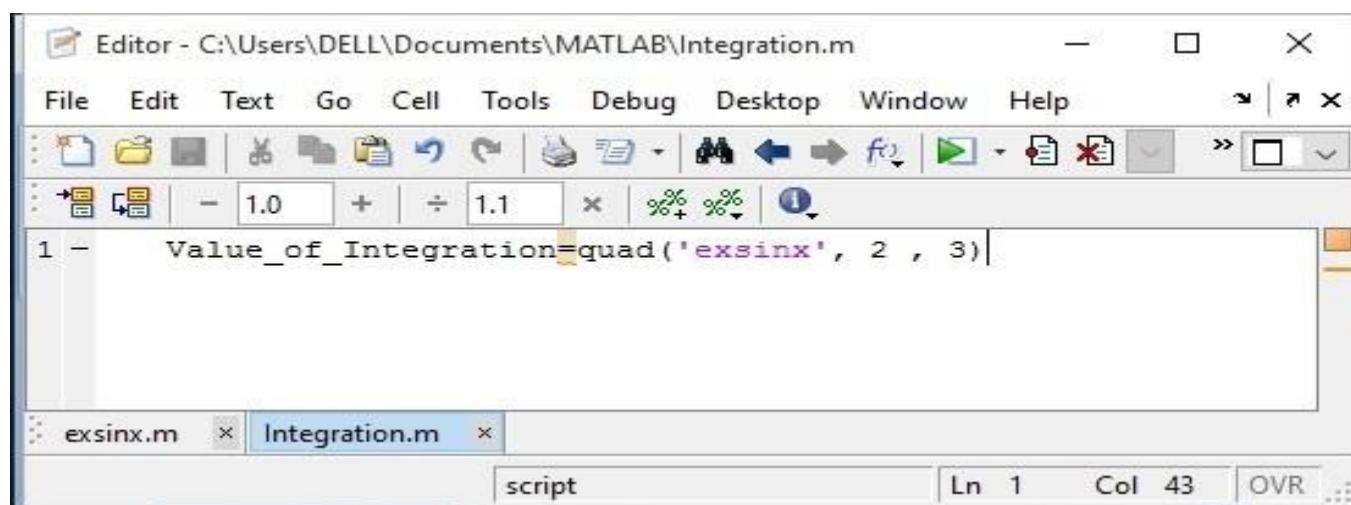
```

Editor - C:\Users\DELL\Documents\MATLAB\exsinx.m

File Edit Text Go Cell Tools Debug Desktop Window Help
exsinx.m | - 1.0 + ÷ 1.1 × %% | ①
1 function f=exsinx(x)
2 f=exp(3.11*sin(x))+(log(1+x)/log(3.11))-2.2
3
4 end
5
6
exsinx Ln 3 Col 1 OVR

```

**m FILES:**



```

Editor - C:\Users\DELL\Documents\MATLAB\Integration.m

File Edit Text Go Cell Tools Debug Desktop Window Help
Integration.m | - 1.0 + ÷ 1.1 × %% | ①
1 Value_of_Integration=quad('exsinx', 2 , 3)
exsinx.m x Integration.m x
script Ln 1 Col 43 OVR

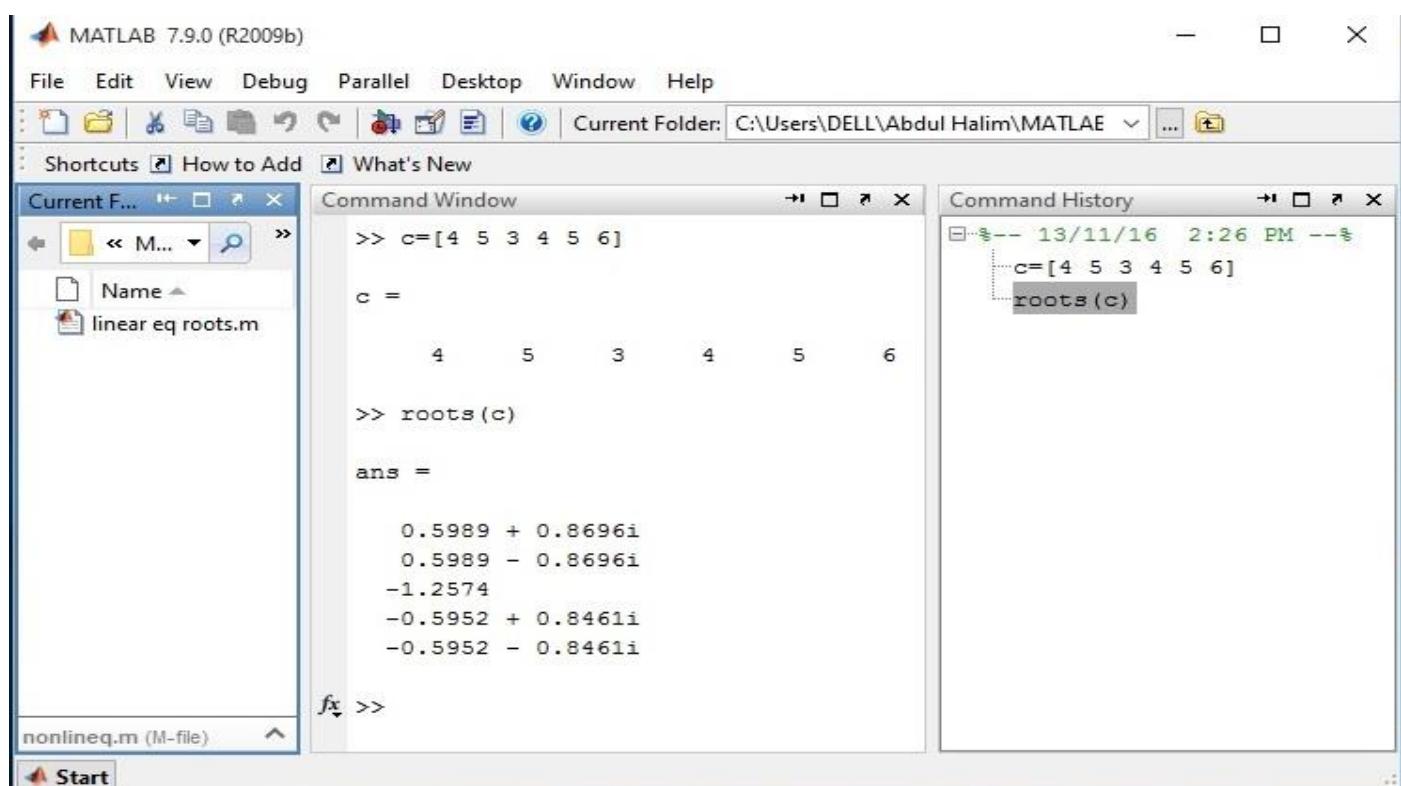
```

## *Solution of a polynomial and transcendental equations*

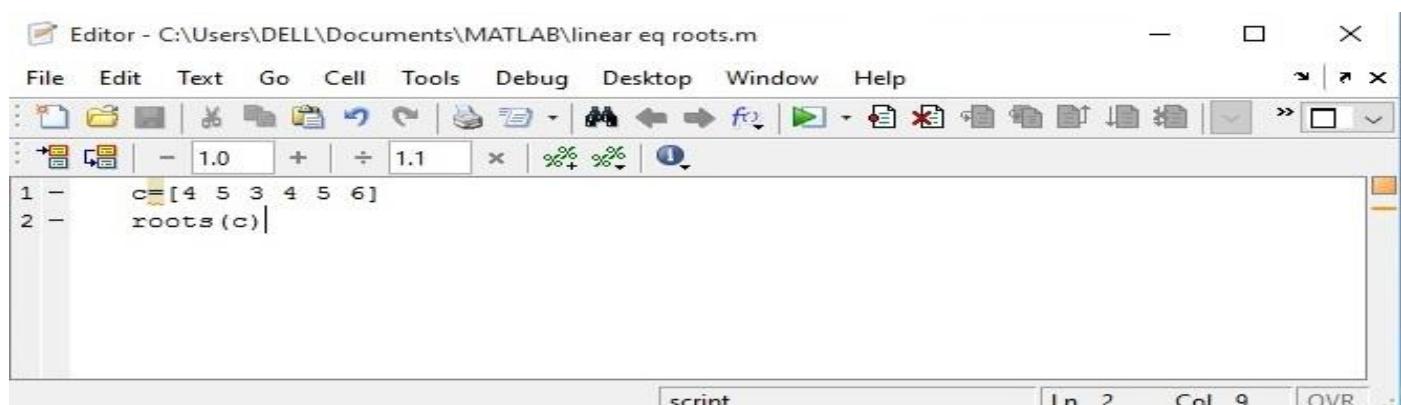
**Problem 6. (a)** Use MATLAB to find the roots of  $4x^6 + 5x^5 + 3x^4 + 4x^2 + 5x + 6 = 0$ .

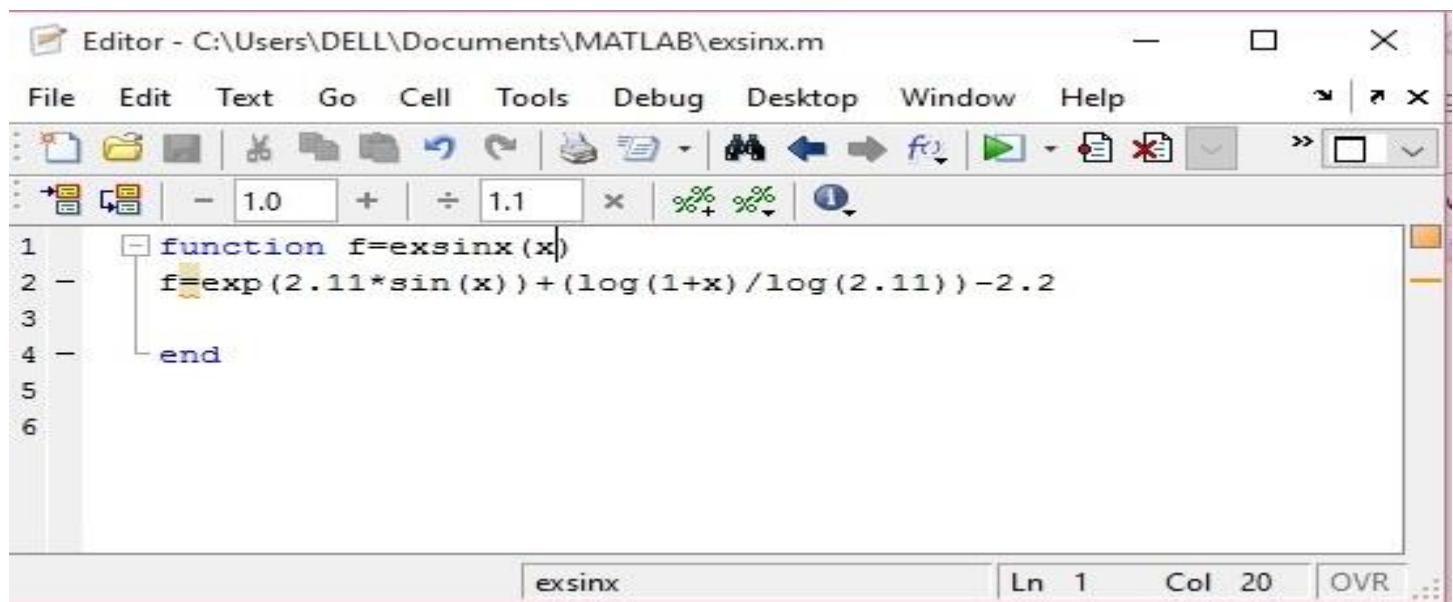
**(b)** Use MATLAB to find a root of the equation  $e^{a \sin x} + \log_a(1+x) - 2.2 = 0$ ,  
 where  $a = \frac{20.1+k}{10}$  and  $k = 1$ . so,  $a = 2.11$

### OUTPUT COMMAND:



### m FILES:



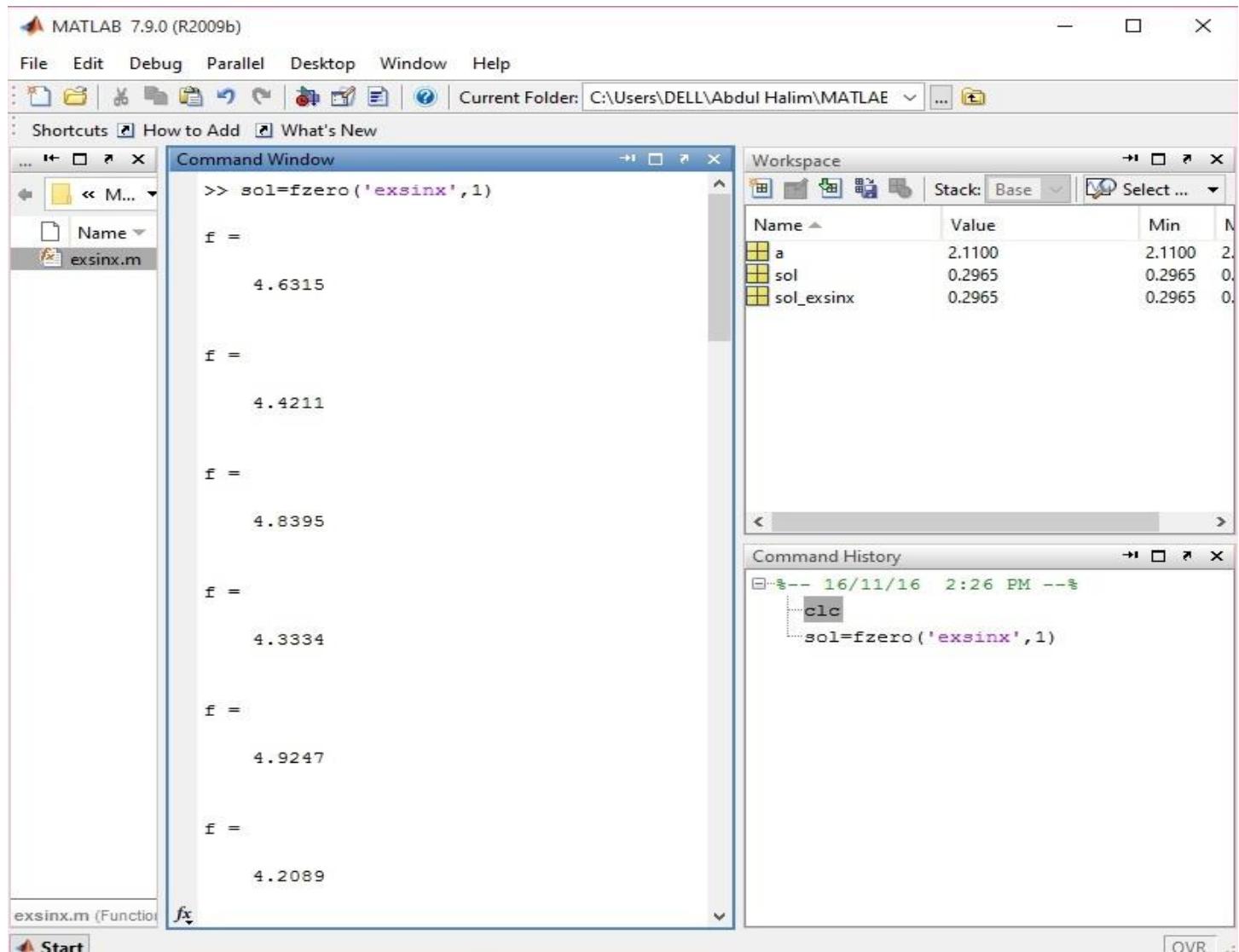
m FILES (FUNCTION) :


The screenshot shows the MATLAB Editor window with the file `exsinx.m` open. The code defines a function `f = exsinx(x)` that calculates the value of the function  $f = \exp(2.11 \sin(x)) + (\log(1+x)/\log(2.11)) - 2.2$ . The editor interface includes a toolbar at the top, a status bar at the bottom, and a vertical scroll bar on the right.

```

1 function f=exsinx(x)
2 f=exp(2.11*sin(x))+(log(1+x)/log(2.11))-2.2
3
4 end
5
6

```

OUTPUT COMMAND:


The screenshot shows the MATLAB workspace and command window. The command window displays the results of running the `fzero` function on the `exsinx` function. The workspace browser on the left shows the `exsinx.m` file is selected. The workspace table shows variables `a`, `sol`, and `sol_exsinx` with their values. The command history at the bottom shows the commands used.

Name	Value	Min	Max
<code>a</code>	2.1100	2.1100	2.1100
<code>sol</code>	0.2965	0.2965	0.2965
<code>sol_exsinx</code>	0.2965	0.2965	0.2965

```

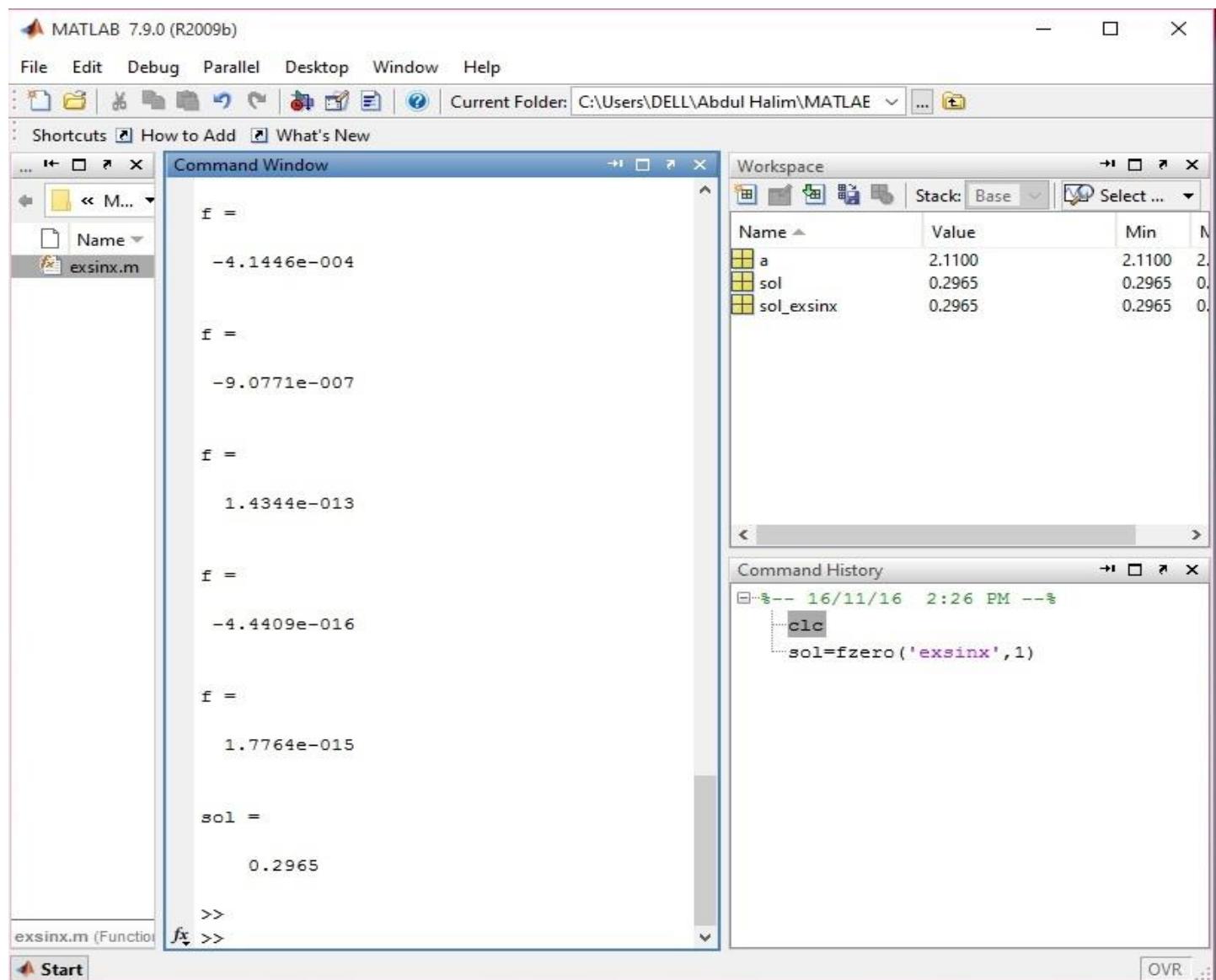
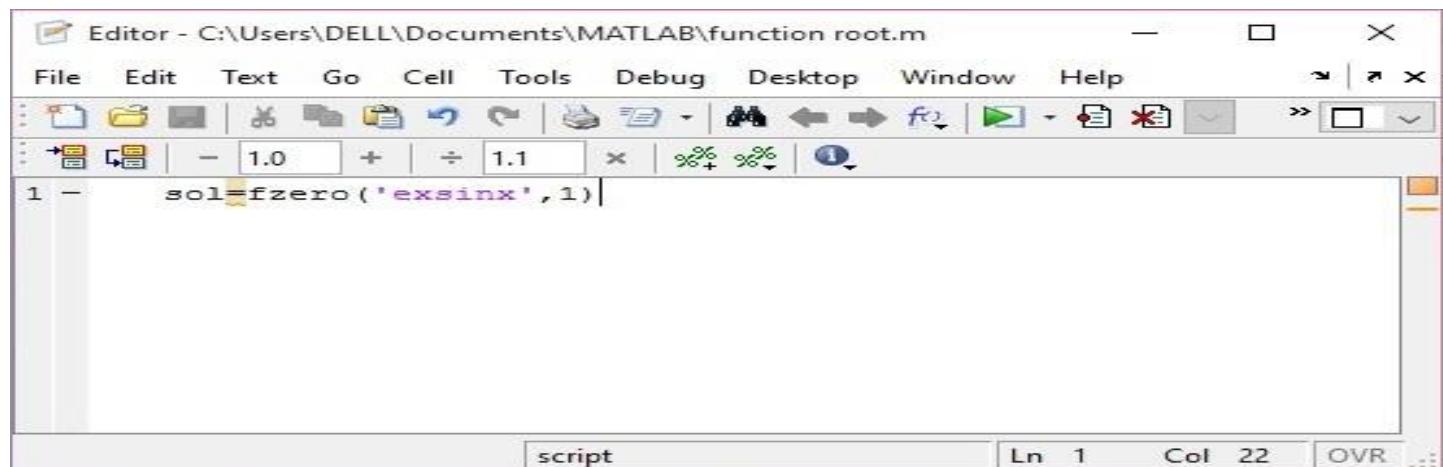
>> sol=fzero('exsinx',1)
f =
    4.6315
f =
    4.4211
f =
    4.8395
f =
    4.3334
f =
    4.9247
f =
    4.2089

```

```

Command History
-- 16/11/16 2:26 PM --
clc
sol=fzero('exsinx',1)

```

**OUTPUT COMMAND:****.m FILES:**

## ***2D and 3D graphics***

**Problem 7. (a)** Use MATLAB to draw the graph of  $y = \sin x$  for  $x \in [0, 4\pi]$ .

**(b)** Use MATLAB to draw the graphs of  $y = \sin x$  and  $y = x^2 \sin x$  in the same figure for  $x \in [0, 4\pi]$ .

**(c)** Plot  $x$ ,  $y$  and  $z$ , where  $x = t$ ,  $x = t^2$ ,  $x = \sin(e^t)$  for  $t \in [-5, 5]$ .

**(d)** Draw the surface  $z = \cos x \cos y$  ( $x, y \in [-5, 5]$ ) using both surf and surfc.

**(e)** Draw the bifurcation diagram for the logistic map by running the following:

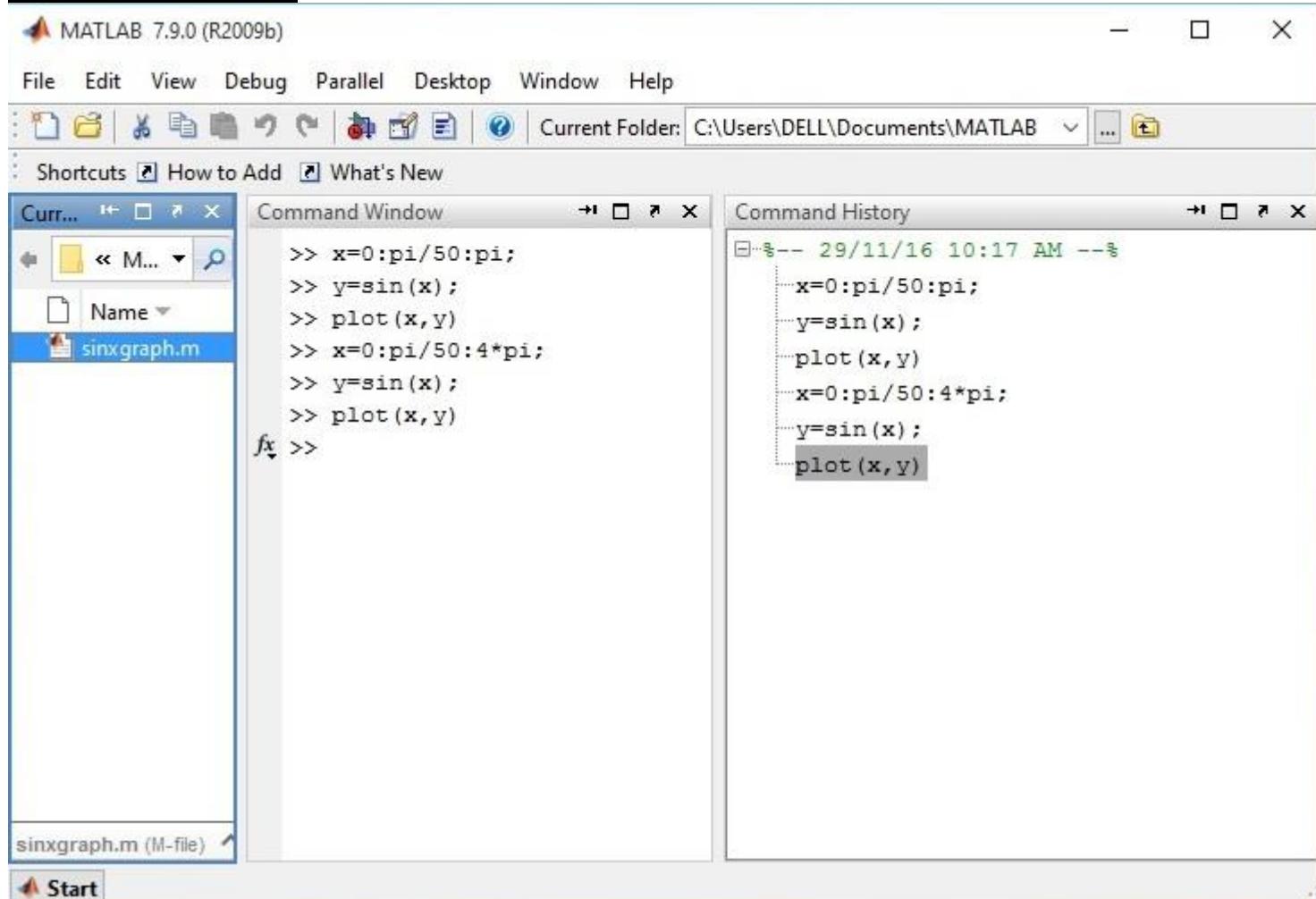
```

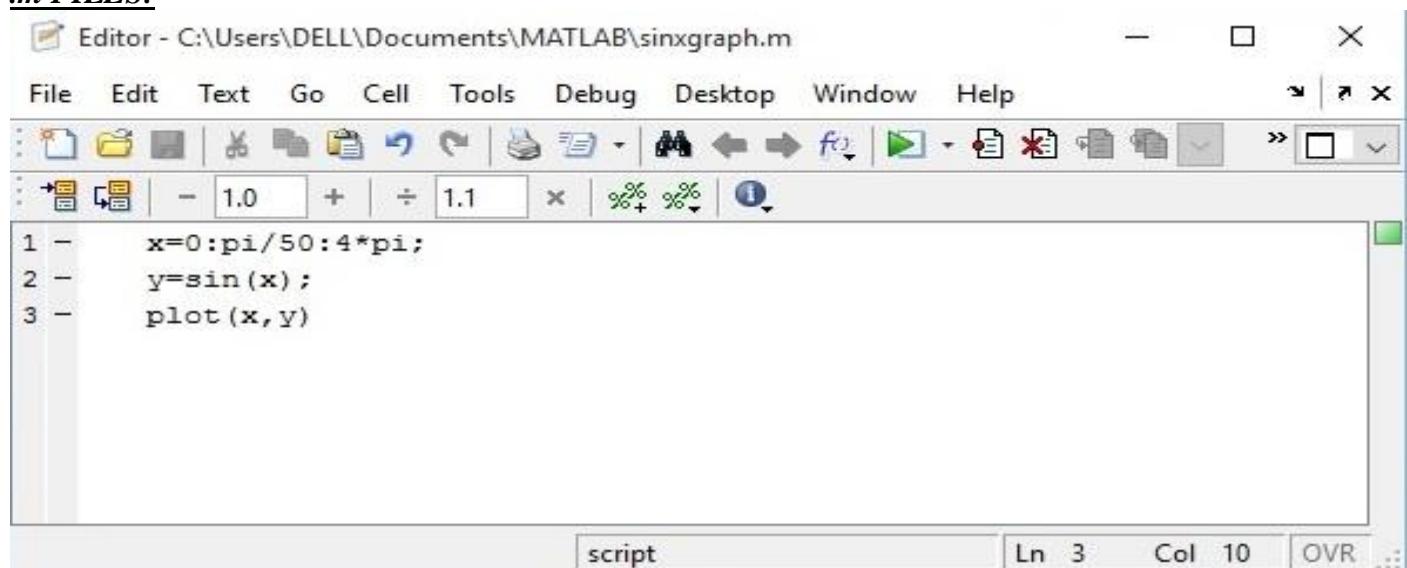
x=0.5;
for a=2.5:0.0001:4
    y=a*x*(1-x);
    plot(a,y)
    hold on
    x=y;
end
hold off

```

Use while loop to draw the same figure.

### **OUTPUT COMMAND:**



**m FILES:**


Editor - C:\Users\DELL\Documents\MATLAB\sinxgraph.m

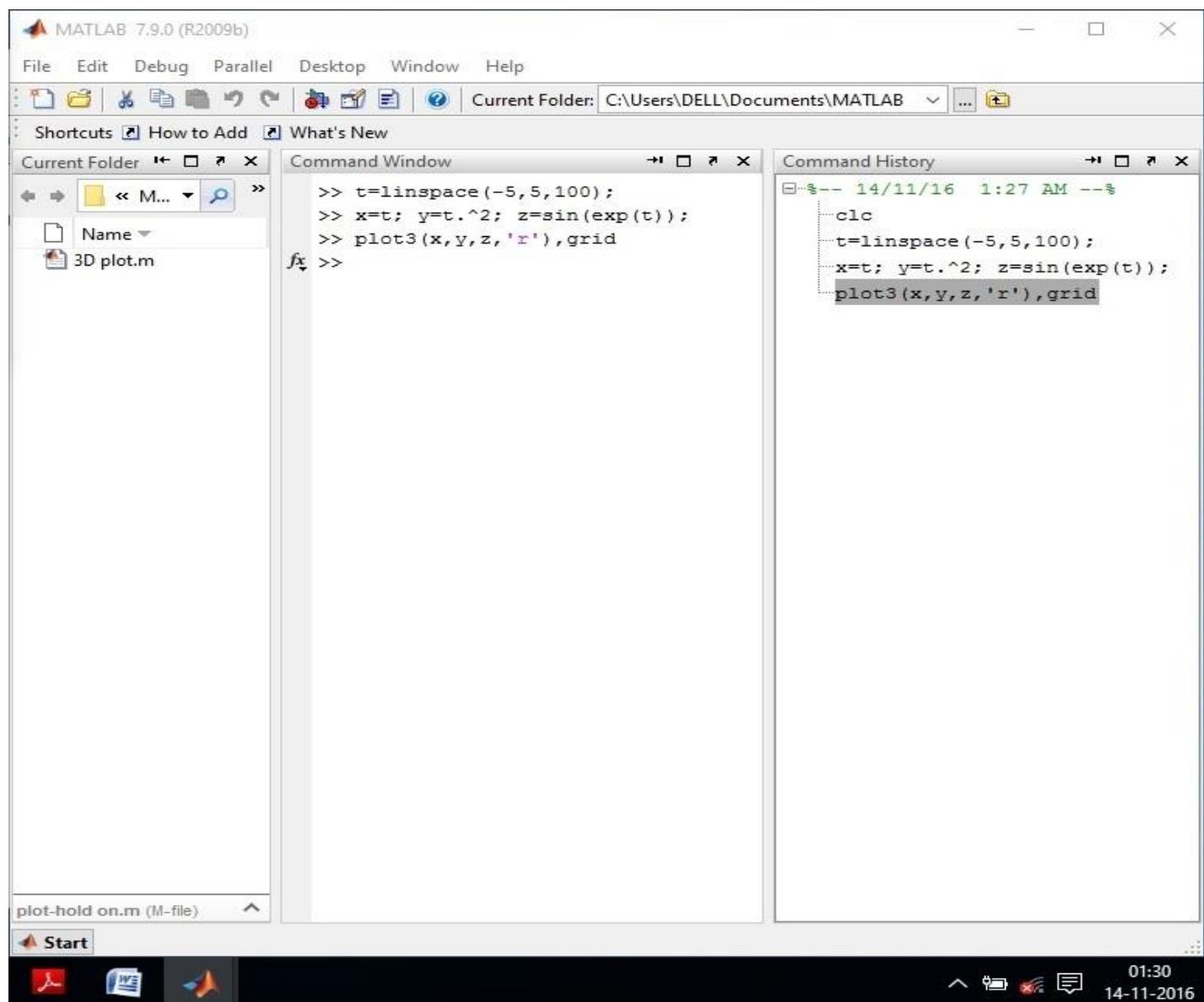
File Edit Text Go Cell Tools Debug Desktop Window Help

```

1 - x=0:pi/50:4*pi;
2 - y=sin(x);
3 - plot(x,y)

```

script Ln 3 Col 10 OVR

**OUTPUT COMMAND:**


MATLAB 7.9.0 (R2009b)

File Edit Debug Parallel Desktop Window Help

Current Folder: C:\Users\DELL\Documents\MATLAB

Current Folder

- Name
- 3D plot.m

Command Window

```

>> t=linspace(-5,5,100);
>> x=t; y=t.^2; z=sin(exp(t));
>> plot3(x,y,z,'r'),grid
fx >>

```

Command History

```

-- 14/11/16 1:27 AM --
clc
t=linspace(-5,5,100);
x=t; y=t.^2; z=sin(exp(t));
plot3(x,y,z,'r'),grid

```

plot-hold on.m (M-file)

Start

01:30 14-11-2016

The screenshot shows the MATLAB Editor window. The title bar says "Editor - C:\Users\DELL\Documents\MATLAB\3D plot.m". The menu bar includes File, Edit, Text, Go, Cell, Tools, Debug, Desktop, Window, Help. Below the menu is a toolbar with various icons. The code area contains the following script:

```
1 - t=linspace(-5,5,100);
2 - x=t; y=t.^2; z=sin(exp(t));
3 - plot3(x,y,z,'r'),grid
```

The status bar at the bottom shows "script" and "Ln 3 Col 22 OVR".

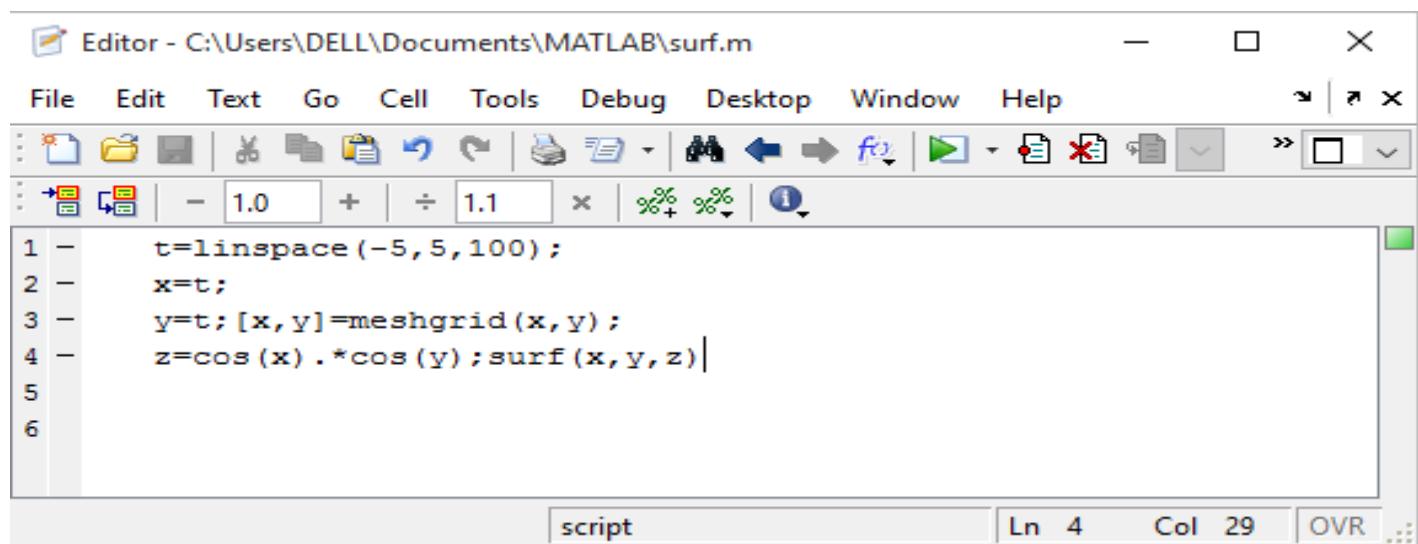
OUTPUT COMMAND:

The screenshot shows the MATLAB desktop interface. The title bar says "MATLAB 7.9.0 (R2009b)". The menu bar includes File, Edit, View, Debug, Parallel, Desktop, Window, Help. The toolbar has icons for file operations. The "Current Folder" browser on the left shows a folder named "3D plot.m". The "Command Window" in the center shows the command history:

```
>> t=linspace(-5,5,100);
>> x=t; y=t.^2; z=sin(exp(t));
>> plot3(x,y,z,'r'),grid
fx >>
```

The "Command History" browser on the right shows the same commands with the current date and time:

```
%-- 14/11/16 1:27 AM --
clc
t=linspace(-5,5,100);
x=t; y=t.^2; z=sin(exp(t));
plot3(x,y,z,'r'),grid
```

m FILES:


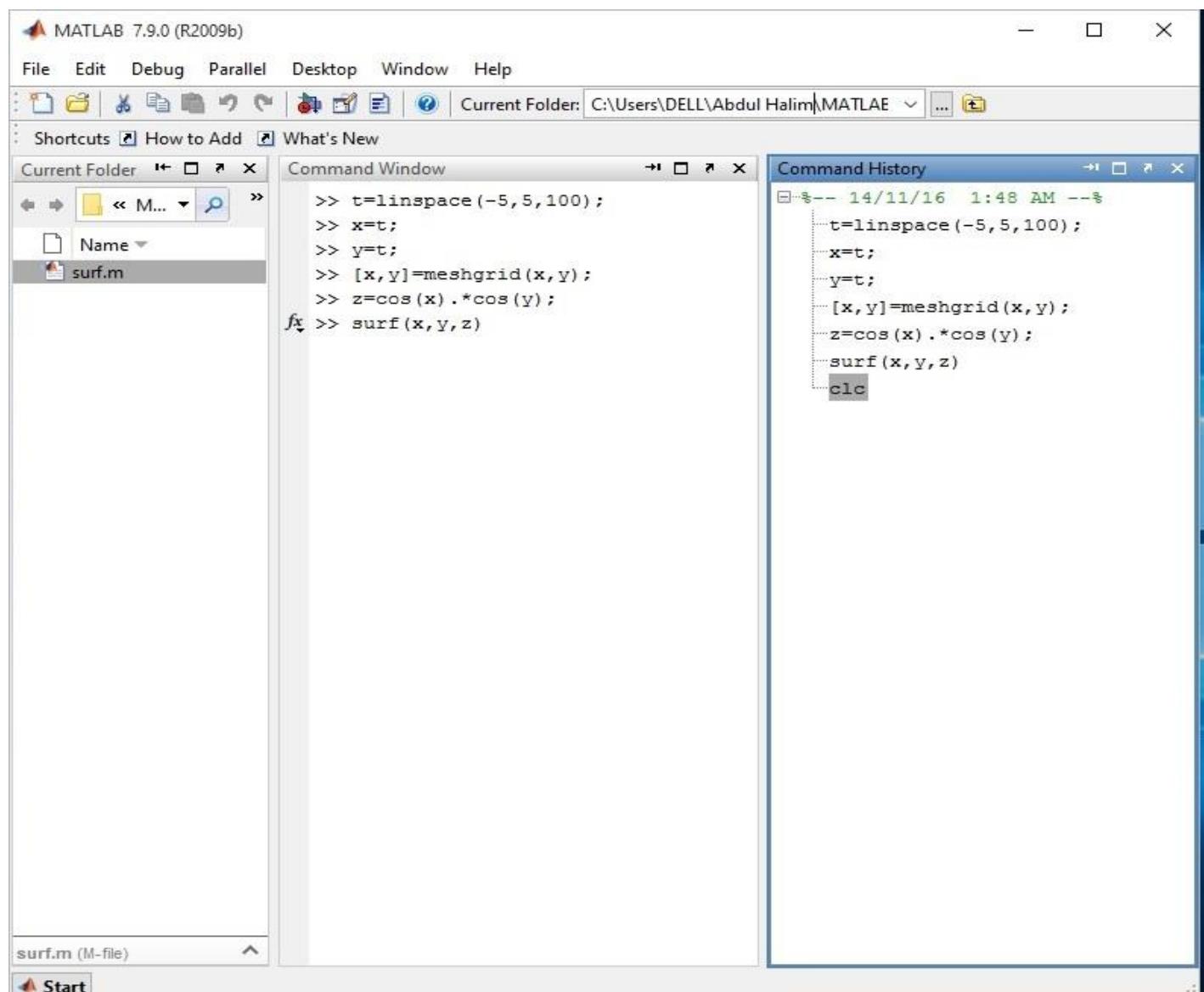
The screenshot shows the MATLAB Editor window with the file `surf.m` open. The code in the editor is:

```

1 - t=linspace (-5,5,100);
2 - x=t;
3 - y=t; [x,y]=meshgrid(x,y);
4 - z=cos (x) .*cos (y);surf (x,y,z)
5 -
6 -

```

The status bar at the bottom indicates "script" and "Ln 4 Col 29".

RESULTS COMMAND :


The screenshot shows the MATLAB interface with the Command Window and History panes visible.

**Command Window:**

```

>> t=linspace(-5,5,100);
>> x=t;
>> y=t;
>> [x,y]=meshgrid(x,y);
>> z=cos(x).*cos(y);
fx >> surf(x,y,z)

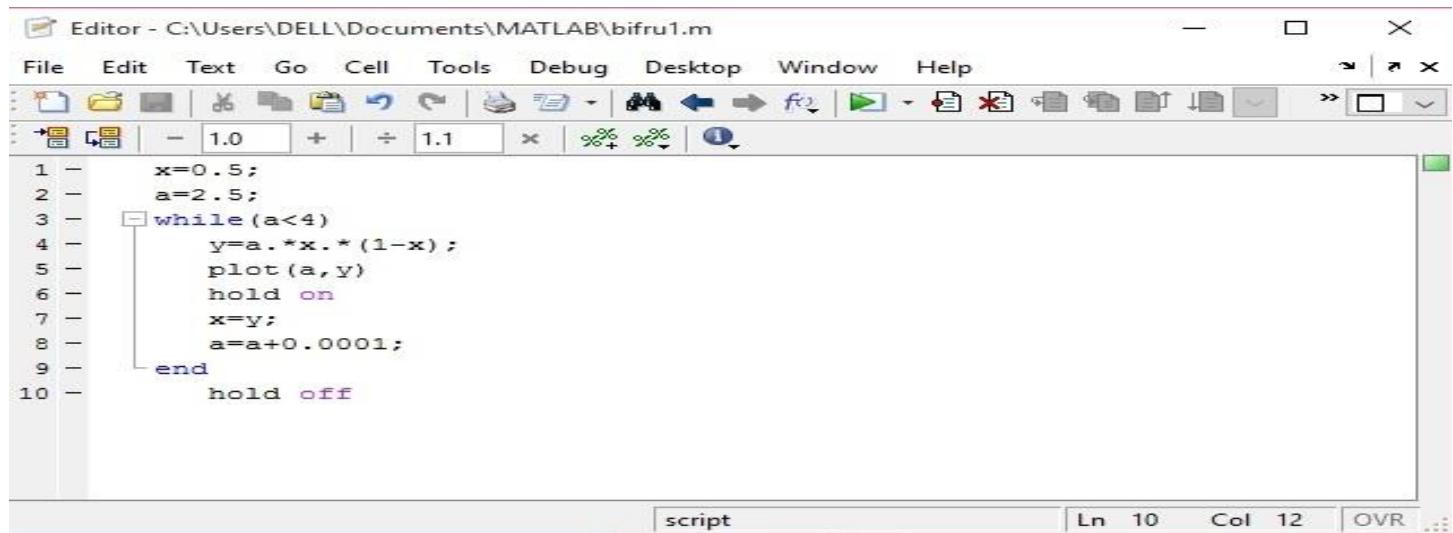
```

**Command History:**

```

14/11/16 1:48 AM --%
>> t=linspace(-5,5,100);
>> x=t;
>> y=t;
>> [x,y]=meshgrid(x,y);
>> z=cos(x).*cos(y);
>> surf(x,y,z)
>> clc

```

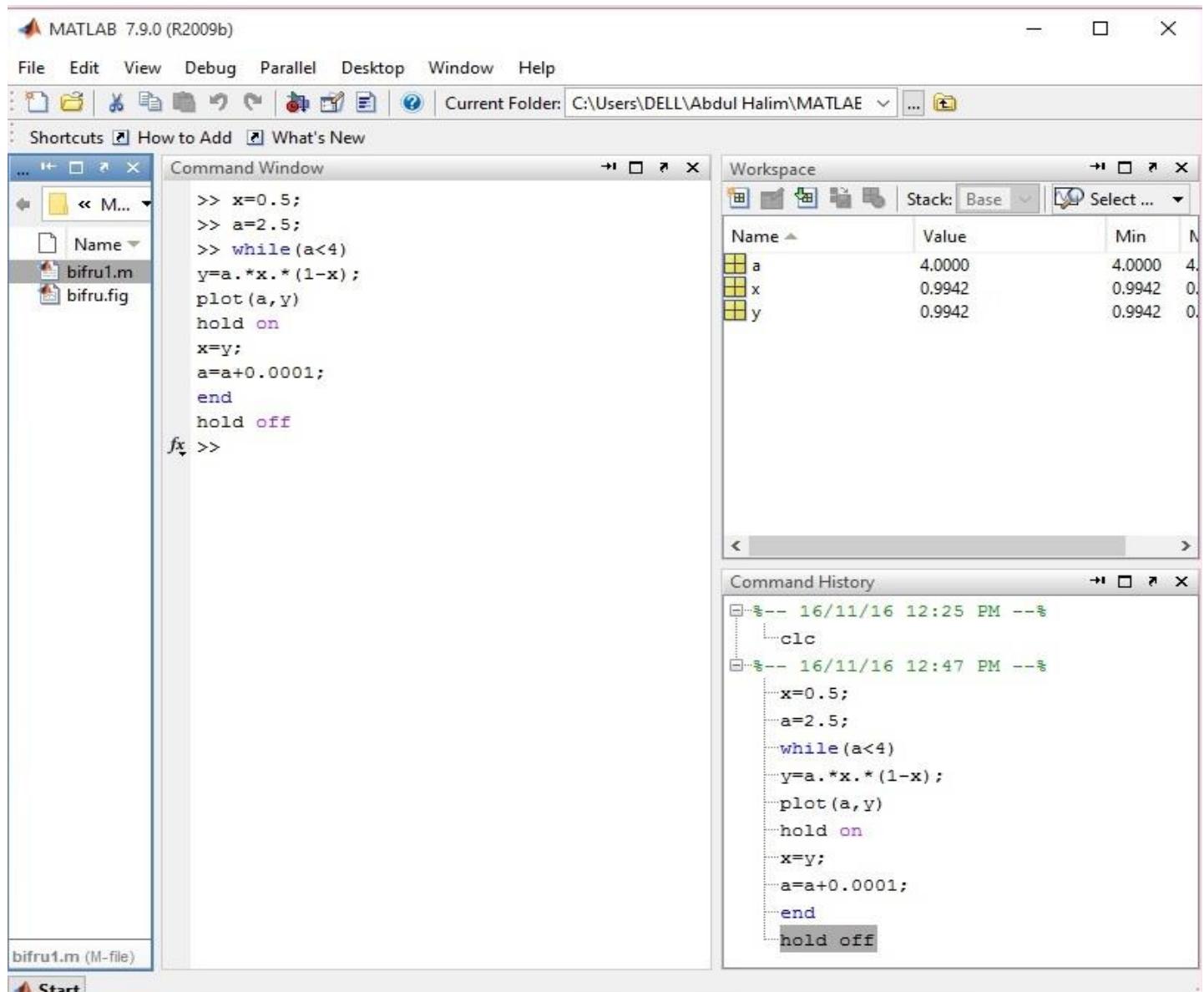
.m FILES:


The screenshot shows the MATLAB Editor window with the file `bifru1.m` open. The code in the editor is:

```

1 - x=0.5;
2 - a=2.5;
3 - while(a<4)
4 -     y=a.*x.*(1-x);
5 -     plot(a,y)
6 -     hold on
7 -     x=y;
8 -     a=a+0.0001;
9 - end
10 - hold off

```

OUTPUT COMMAND:


The screenshot shows the MATLAB interface with the following windows open:

- Command Window:** Displays the command history and the execution of the script `bifru1.m`.
- Workspace:** Shows the variables `a`, `x`, and `y` with their values.
- Command History:** Shows the commands entered and the output generated by the script.

**Command Window Output:**

```

>> x=0.5;
>> a=2.5;
>> while(a<4)
y=a.*x.*(1-x);
plot(a,y)
hold on
x=y;
a=a+0.0001;
end
hold off
fx >>

```

**Workspace Variables:**

Name	Value	Min	Max
<code>a</code>	4.0000	4.0000	4.0000
<code>x</code>	0.9942	0.9942	0.9942
<code>y</code>	0.9942	0.9942	0.9942

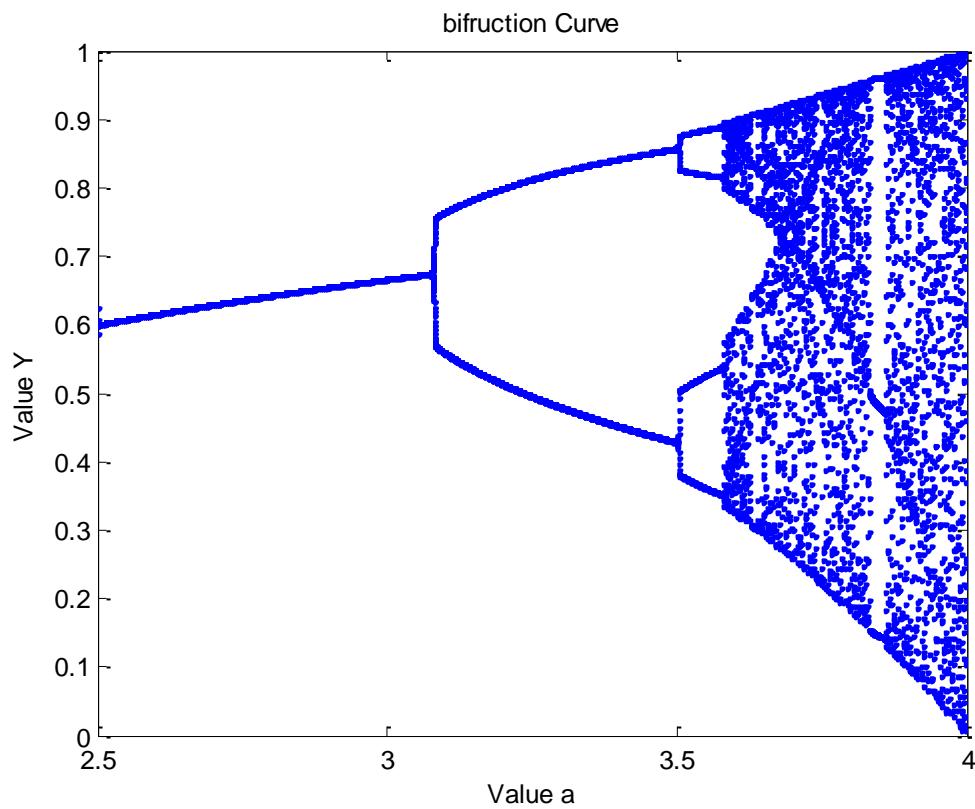
**Command History:**

```

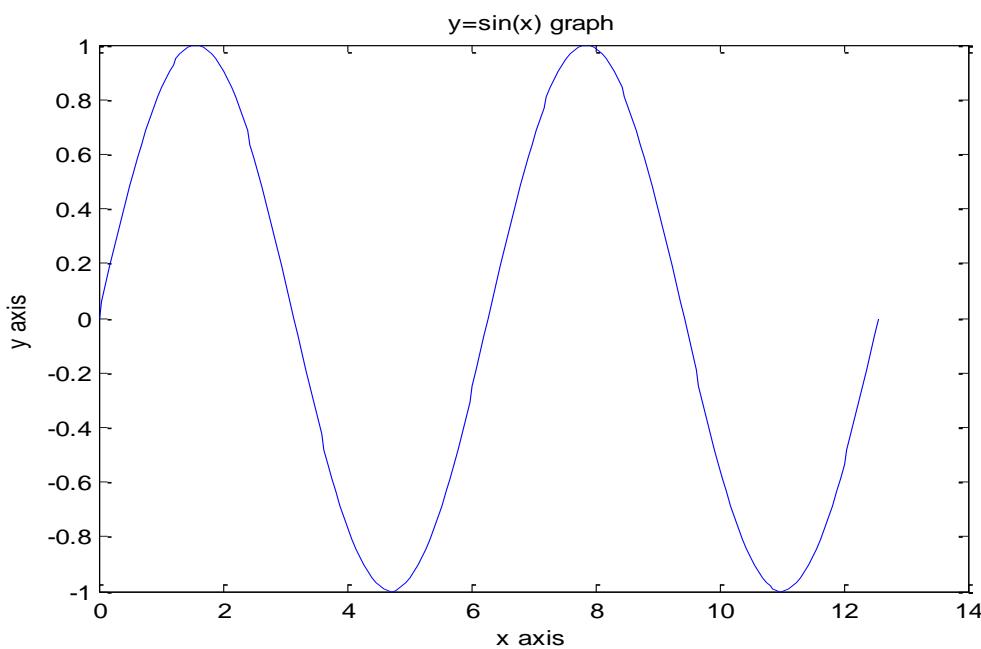
--> 16/11/16 12:25 PM --%
    clc
--> 16/11/16 12:47 PM --%
    x=0.5;
    a=2.5;
    while(a<4)
        y=a.*x.*(1-x);
        plot(a,y)
        hold on
        x=y;
        a=a+0.0001;
    end
    hold off

```

**OUTPUT PICTURE:**



**OUTPUT PICTURE:**



## *Solutions of differential equations*

**Problem 8. (a)** Use MATLAB to solve the following ODE:

$$\frac{dx}{dt} = t + x^2, \quad x(0) = 0, \text{ for } t \in [0,1]. \text{ Plot } t \text{ versus } x.$$

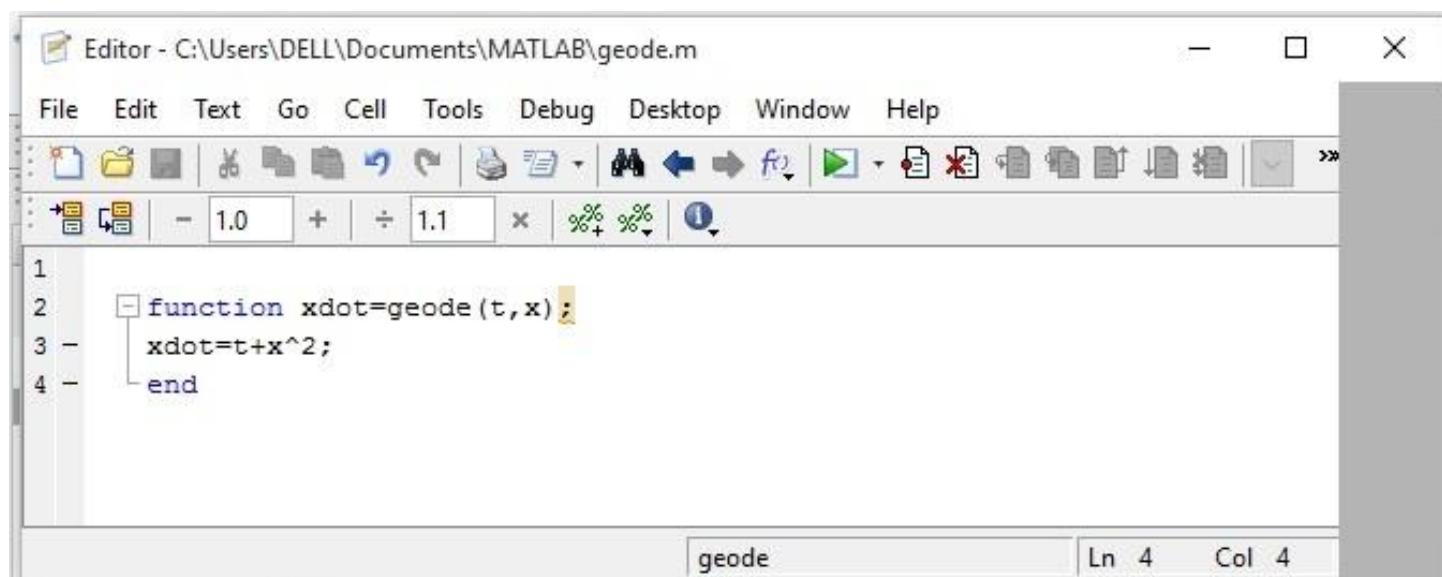
**(b)** Consider the following system of differential equations:

$$\begin{aligned}\frac{dx}{dt} &= x(1-x) - \frac{2xy}{0.5+x}, & x(0) &= 0.5, \\ \frac{dy}{dt} &= \frac{cxy}{0.5+x} - y, & y(0) &= 0.5.\end{aligned}$$

(i) Take  $c = 2$ . Use MATLAB to solve the above system for  $t = 0$  to  $t = 500$ , and plot the solution ( $x$  versus  $y$ ). Also plot  $t$  versus  $x$  and  $t$  versus  $y$  in a single figure (Use red colour to indicate  $x$  and blue to indicate  $y$ ).

(ii) Take  $c = 4$ . Use MATLAB to solve the above system for  $t = 0$  to  $t = 500$ , and plot the solution ( $x$  versus  $y$ ). Also plot  $t$  versus  $x$  and  $t$  versus  $y$  in a single figure (Use red colour to indicate  $x$  and blue to indicate  $y$ ).

### m FILES (FUNCTION):



The screenshot shows the MATLAB Editor window with the following details:

- Title Bar:** Editor - C:\Users\DELL\Documents\MATLAB\geode.m
- Menu Bar:** File Edit Text Go Cell Tools Debug Desktop Window Help
- Toolbar:** Includes icons for New, Open, Save, Run, Stop, and others.
- Command Window:** Shows numerical values: 1.0, 1.1, %, %, and a help icon.
- Code Area:**

```

1
2 function xdot=geode(t,x);
3 - xdot=t+x^2;
4 - end

```
- Status Bar:** geode Ln 4 Col 4

**OUTPUT COMMAND:**

The screenshot shows the MATLAB 7.9.0 (R2009b) interface. The Command Window displays the execution of a script named 'geode.m'. The Command History window shows the full command history, including the plot command and its arguments.

```
>> tspan=[0 1];x0=0;
>> [t,x]=ode23('geode',tspan,x0)

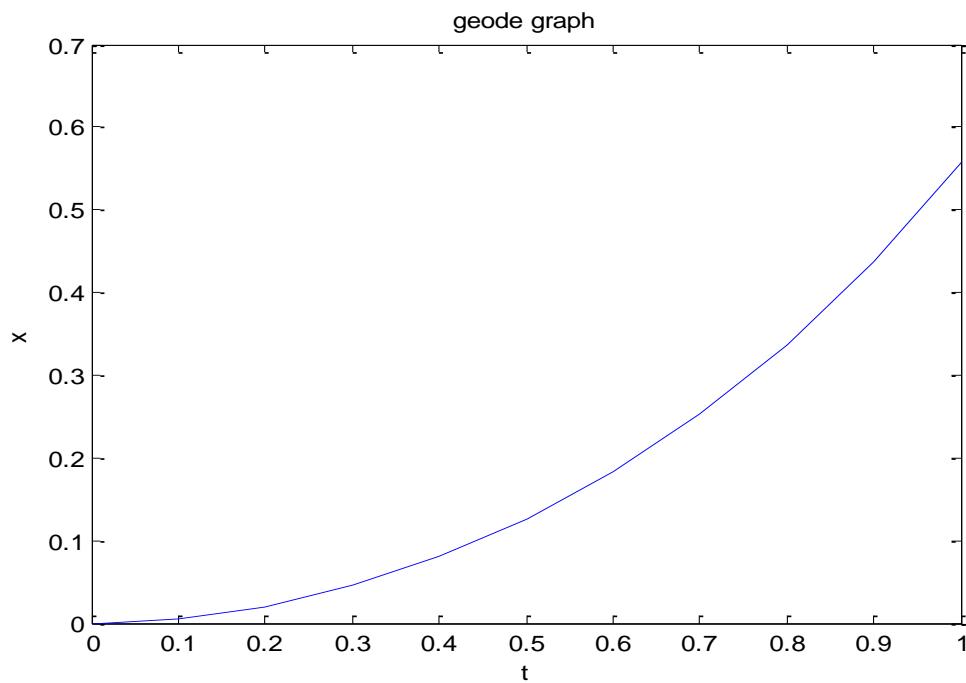
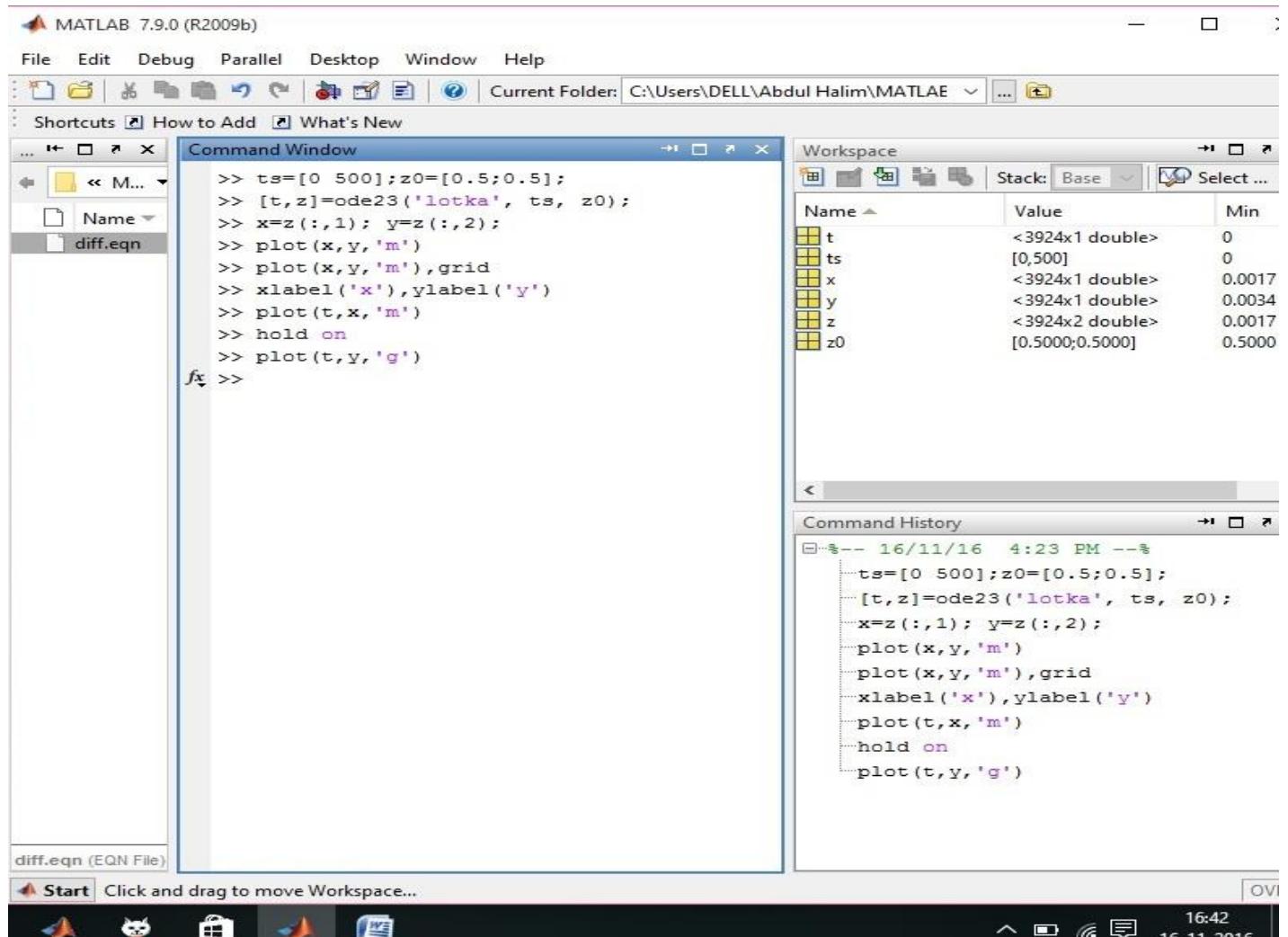
t =
    0
    0.1000
    0.2000
    0.3000
    0.4000
    0.5000
    0.6000
    0.7000
    0.8000
    0.9000
    1.0000

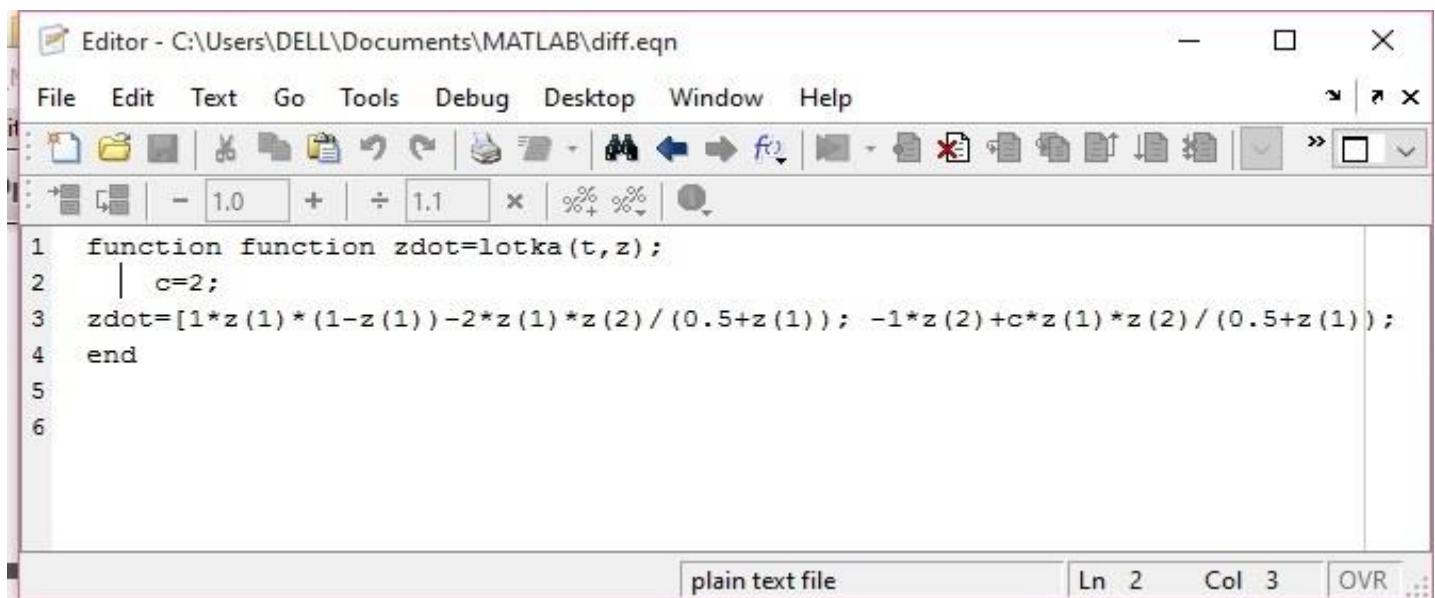
x =
    0
    0.0050
    0.0200
    0.0451
    0.0805
    0.1266
    0.1840
    0.2538
    0.3375
    0.4375
    0.5571

>> plot(t,x)
fx >> xlabel('t'), ylabel('x')
```

Command History:

```
%-- 13/11/16 11:02 AM --%
tspan=[0 1];x0=0;
[t,x]=ode23('geode',tspan,x0)
plot(t,x)
xlabel('t'), ylabel('x')
```

**OUTPUT GRAPH:****OUTPUT COMMAND:**

**m FILES(FUNCTION):**


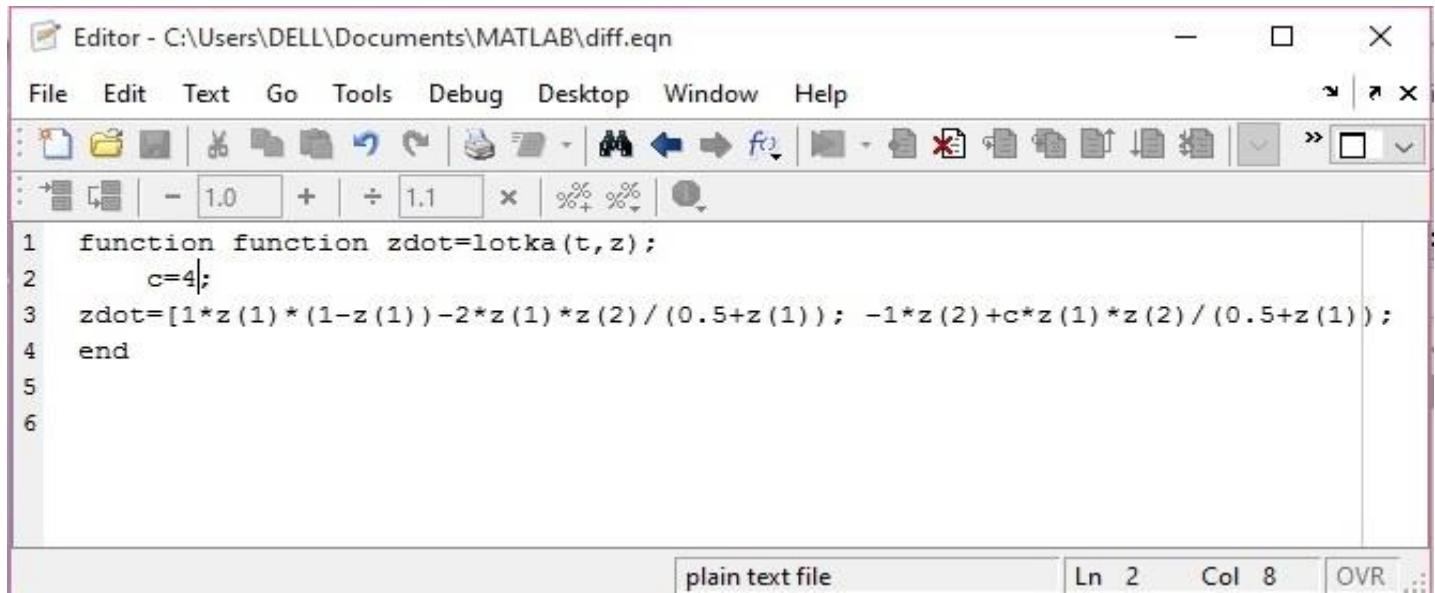
Editor - C:\Users\DELL\Documents\MATLAB\diff.eqn

```

1 function function zdot=lotka(t,z);
2     | c=2;
3 zdot=[1*z(1)*(1-z(1))-2*z(1)*z(2)/(0.5+z(1)); -1*z(2)+c*z(1)*z(2)/(0.5+z(1));
4 end
5
6

```

plain text file      Ln 2      Col 3      OVR

**m FILES(FUNCTION):**


Editor - C:\Users\DELL\Documents\MATLAB\diff.eqn

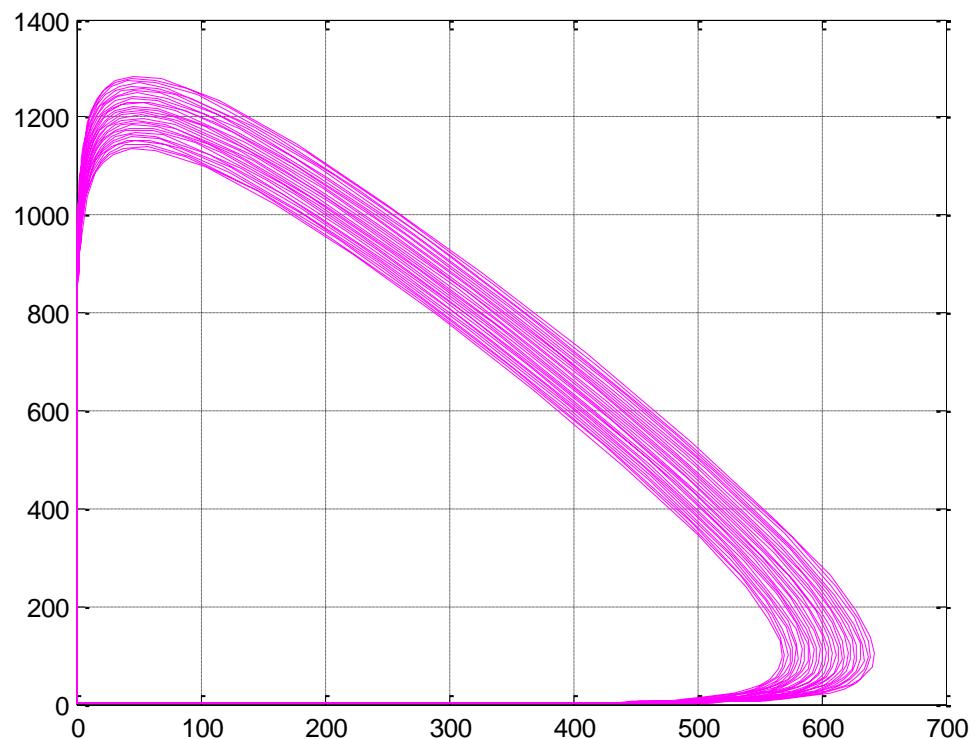
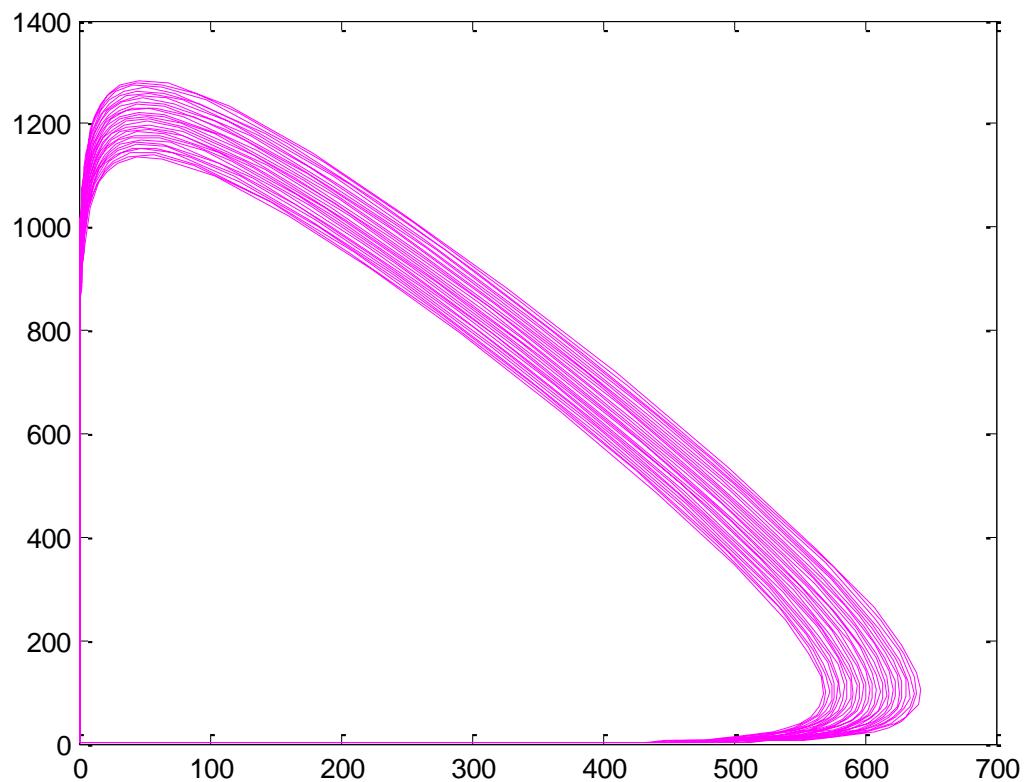
```

1 function function zdot=lotka(t,z);
2     c=4;
3 zdot=[1*z(1)*(1-z(1))-2*z(1)*z(2)/(0.5+z(1)); -1*z(2)+c*z(1)*z(2)/(0.5+z(1));
4 end
5
6

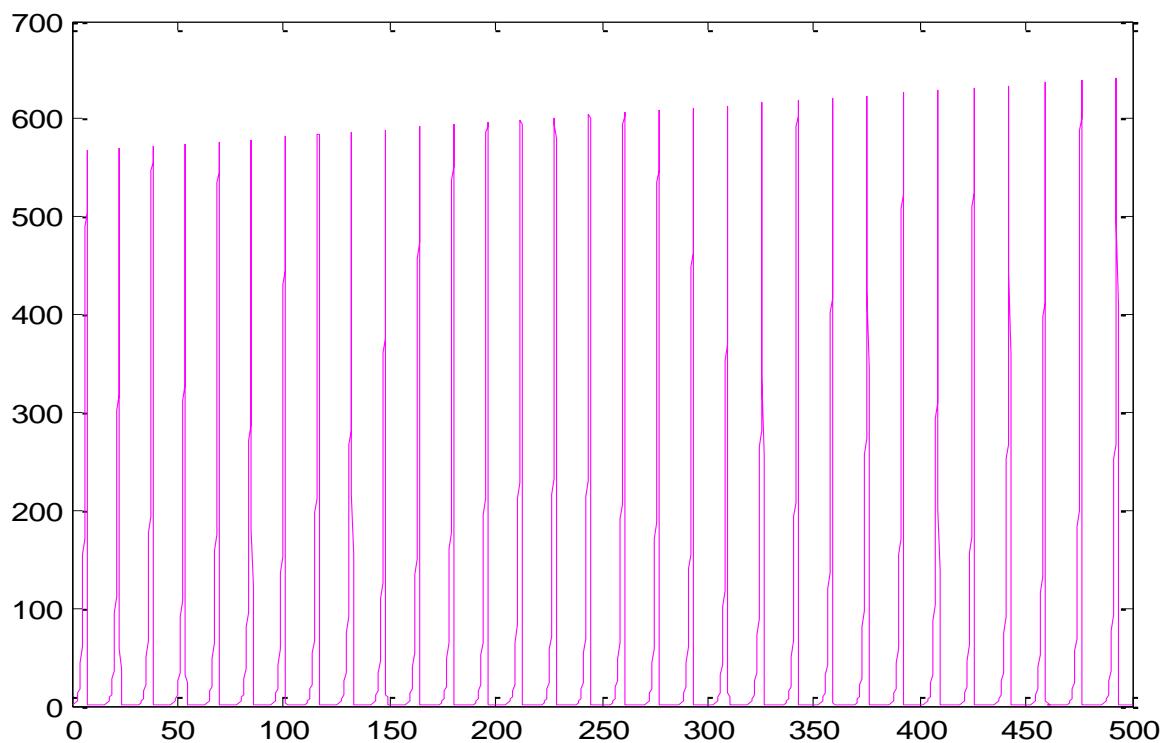
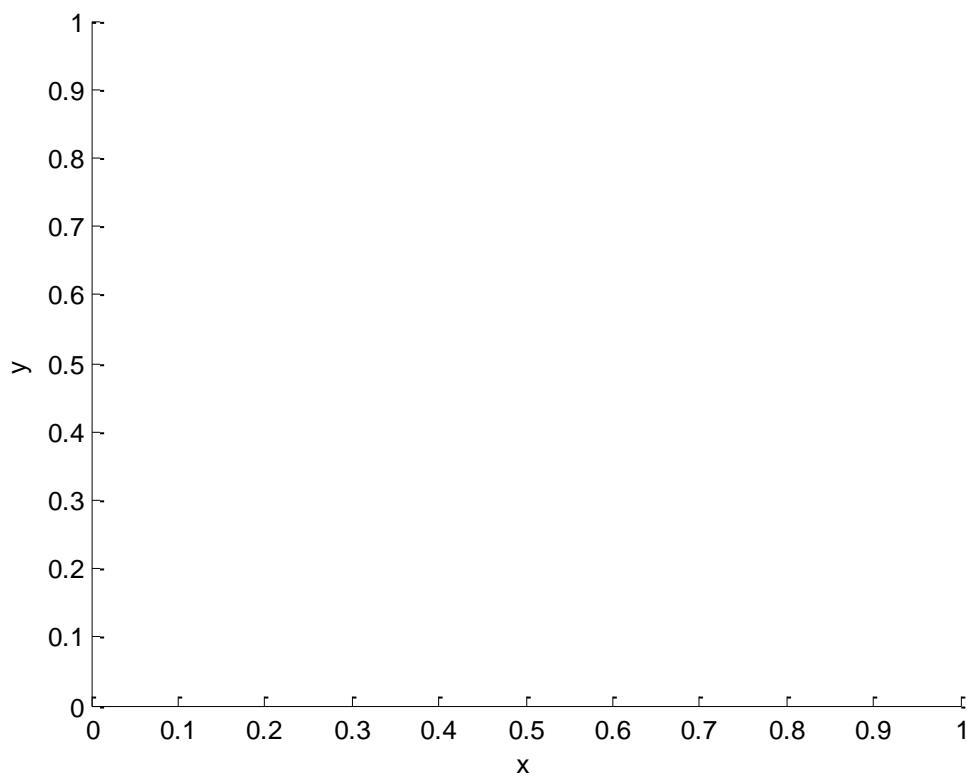
```

plain text file      Ln 2      Col 8      OVR

**OUTPUT GRAPH:**



**OUTPUT GRAPH:**



## ***Handling complex numbers***

**Problem 9.** Let  $z=2+3i$ . Use MATLAB to answer the following questions.

- (a) Find  $|z|$ .
- (b) Find the real and imaginary parts of  $z$ .
- (c) Find the conjugate of  $z$ .
- (d) Compute the value of  $e^{i\pi}$ .

### **OUTPUT COMMAND:**

The screenshot shows the MATLAB 7.9.0 (R2009b) desktop environment. The Command Window displays the following code and results:

```

>> z=2+3i;
>> mod_z=abs(z)
mod_z =
3.6056

>> re_part=real(z)
re_part =
2

>> img_part=imag(z)
img_part =
3

>> c=conj(z)
c =
2.0000 - 3.0000i

>> d=exp(i*pi)
d =
-1.0000 + 0.0000i

```

The Command History window shows the commands entered:

```

13/11/16 12:44 AM --%
clc
z=2+3i;
mod_z=abs(z)
re_part=real(z)
img_part=imag(z)
c=conj(z)
d=exp(i*pi)

```

### **.m FILES:**

The screenshot shows the MATLAB Editor displaying the file `complex.m`. The code in the editor is identical to the code shown in the Command Window above:

```

1 - z=2+3i;
2 - mod_z=abs(z)
3 - re_part=real(z)
4 - img_part=imag(z)
5 - c=conj(z)
6 - d=exp(i*pi)
7

```

## *Statistical data analysis*

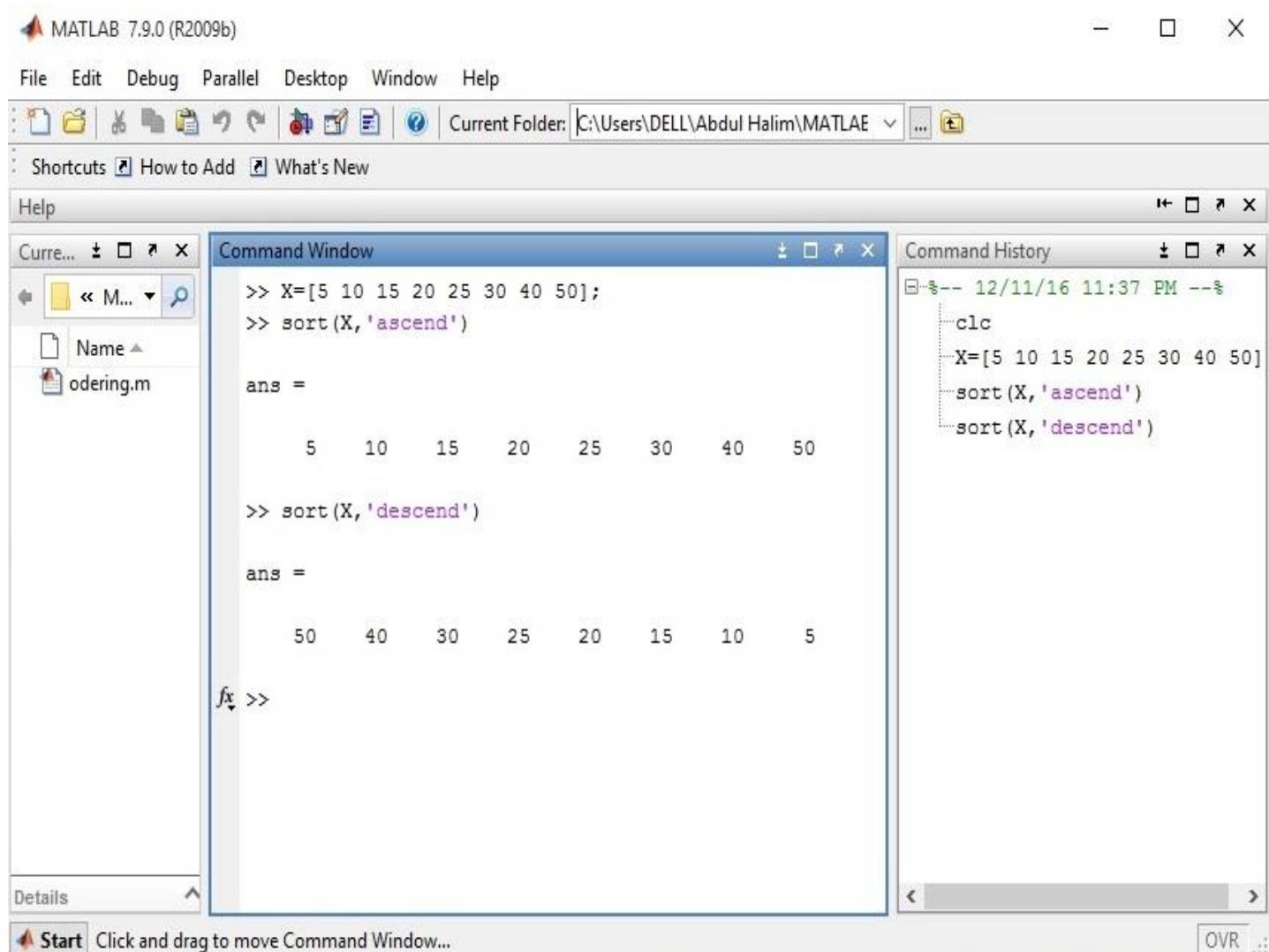
**Problem 10.** Consider the following data:

5, 10, 15, 20, 25, 30, 40, 50.

Use MATLAB to answer the following questions.

- (a) Sort the numbers in ascending and descending order.
- (b) Find (i) minimum and maximum value, (ii) mean, (iii) median, (iv) standard deviation of the data.
- (c) Draw a histogram, a bar diagram (both horizontal and vertical), and a pie chart of the data.

### **OUTPUT COMMAND:**



The screenshot shows the MATLAB 7.9.0 (R2009b) desktop environment. The menu bar includes File, Edit, Debug, Parallel, Desktop, Window, and Help. The toolbar has various icons for file operations. The current folder is set to C:\Users\DELL\Abdul Halim\MATLAE. The Command Window displays the following code and output:

```

>> X=[5 10 15 20 25 30 40 50];
>> sort(X, 'ascend')
ans =
      5     10     15     20     25     30     40     50

>> sort(X, 'descend')
ans =
      50     40     30     25     20     15     10      5

```

The Command History window shows the command history with the same entries. The Current Folder browser shows a file named 'odering.m'.

m FILES:

The screenshot shows the MATLAB Command Window interface. The title bar says "Command Window". Below it is a toolbar with various icons. The main area contains the code for 'odering.m':

```

1 - X=[5 10 15 20 25 30 40 50];
2 - sort(X, 'ascend')
3 - sort(X, 'descend')

```

At the bottom, there is a status bar with "script", "Ln 3", "Col 18", and "OVR".

OUTPUT COMMAND:

The screenshot shows the MATLAB desktop interface. The title bar says "MATLAB 7.9.0 (R2009b)". The left sidebar shows a file tree with "statVal.m" selected. The central area has two panes: "Command Window" on the left and "Command History" on the right. The Command Window pane shows the execution of a script:

```

>> X=[5 10 15 20 25 30 40 50];
>> max(X)

ans =
    50

>> min(X)

ans =
    5

>> mean(X)

ans =
    24.3750

>> median(X)

ans =
    22.5000

>> std(X)

ans =
    15.2216

```

The Command History pane shows the entire session history:

```

12/11/16 11:59 PM -->
X=[5 10 15 20 25 30 40 50];
max(X)
min(X)
mean(X)
median(X)
std(X)

```

**m FILES:**

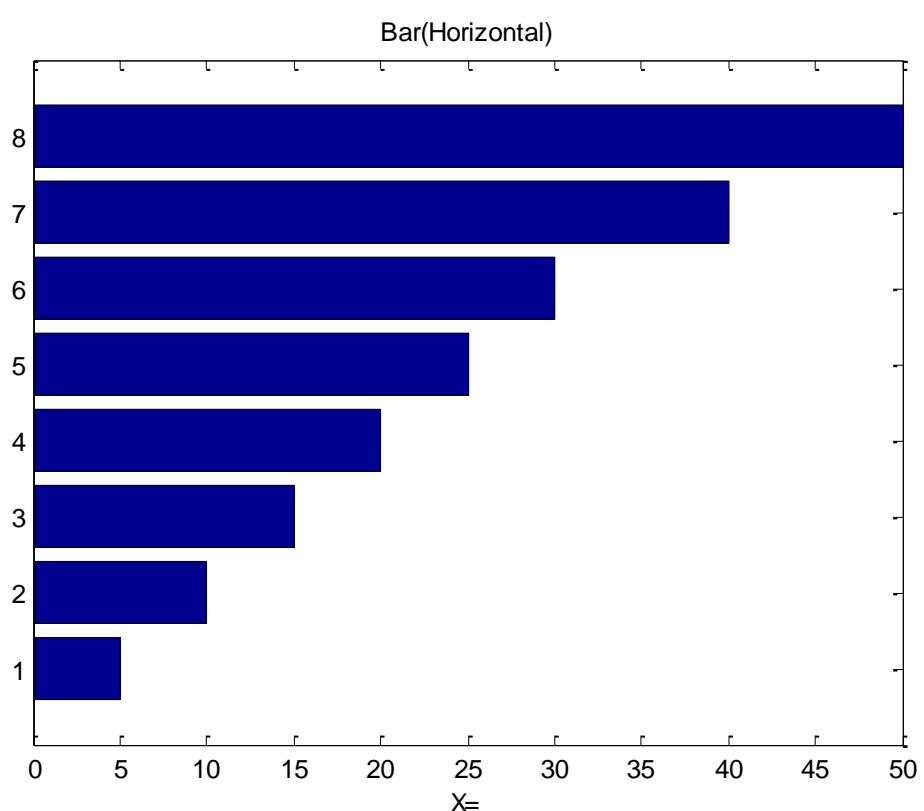
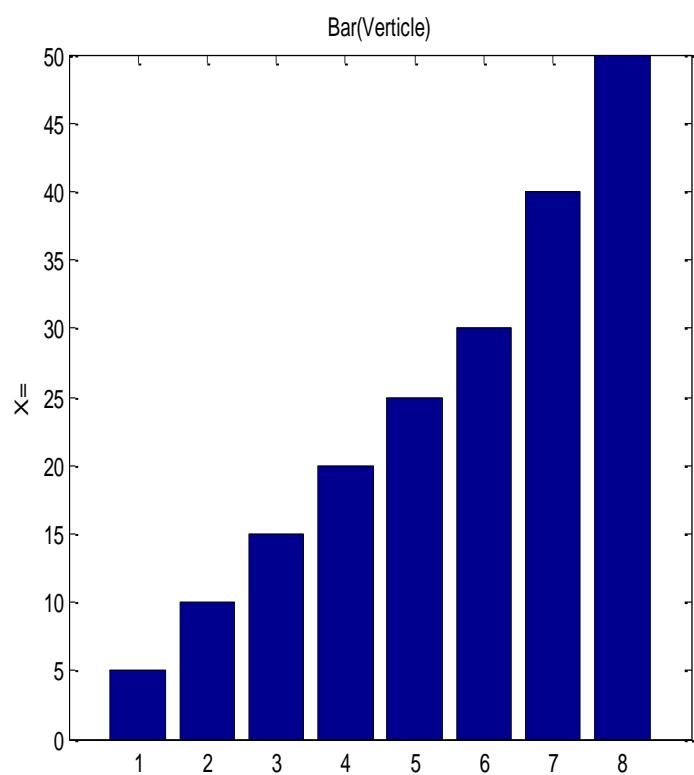
Editor - C:\Users\DELL\Documents\MATLAB\statVal.m

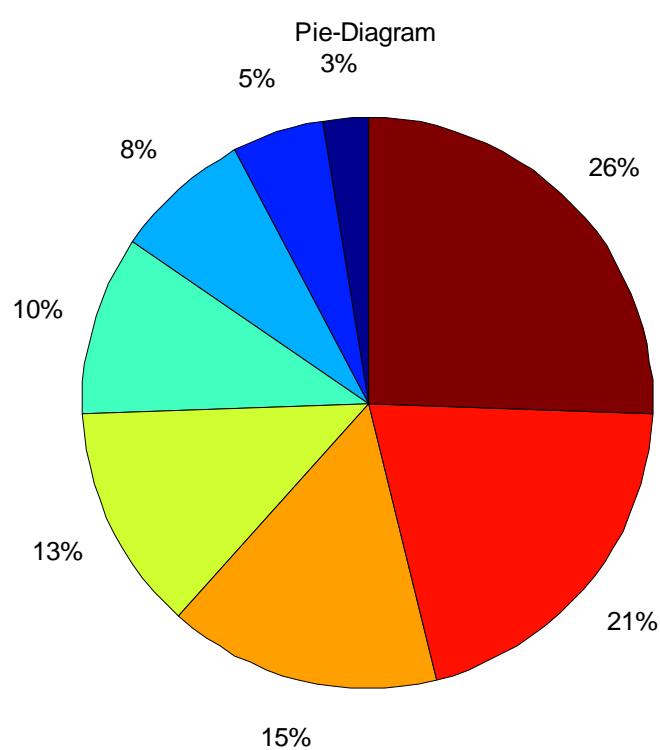
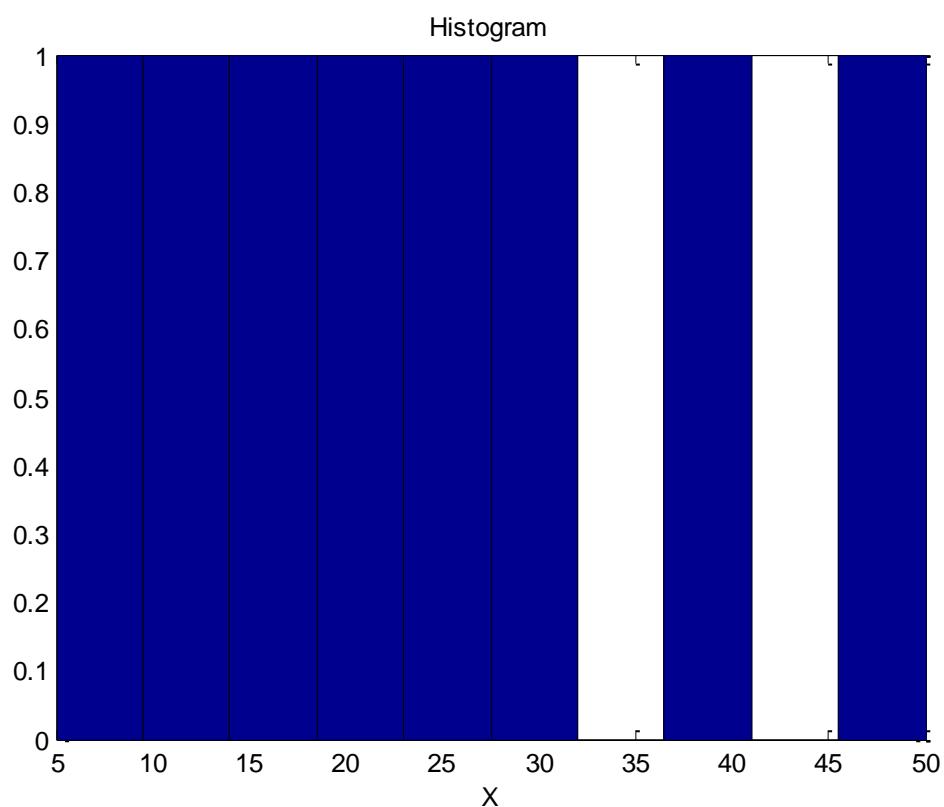
```
File Edit Text Go Cell Tools Debug Desktop Window Help
script
Ln 6 Col 7 OVR ...
1 - X=[5 10 15 20 25 30 40 50];
2 - max(X)
3 - min(X)
4 - mean(X)
5 - median(X)
6 - std(X)|
```

**OUTPUT COMMAND:**

Editor - C:\Users\DELL\Documents\MATLAB\diagram.m

```
File Edit Text Go Cell Tools Debug Desktop Window Help
script
Ln 5 Col 7 OVR ...
1 - X=[5 10 15 20 25 30 40 50];
2 - bar(X)
3 - barh(X)
4 - hist(X)
5 - pie(X)|
```

**OUTPUT GRAPH:**



## ***Correlation and curve fitting***

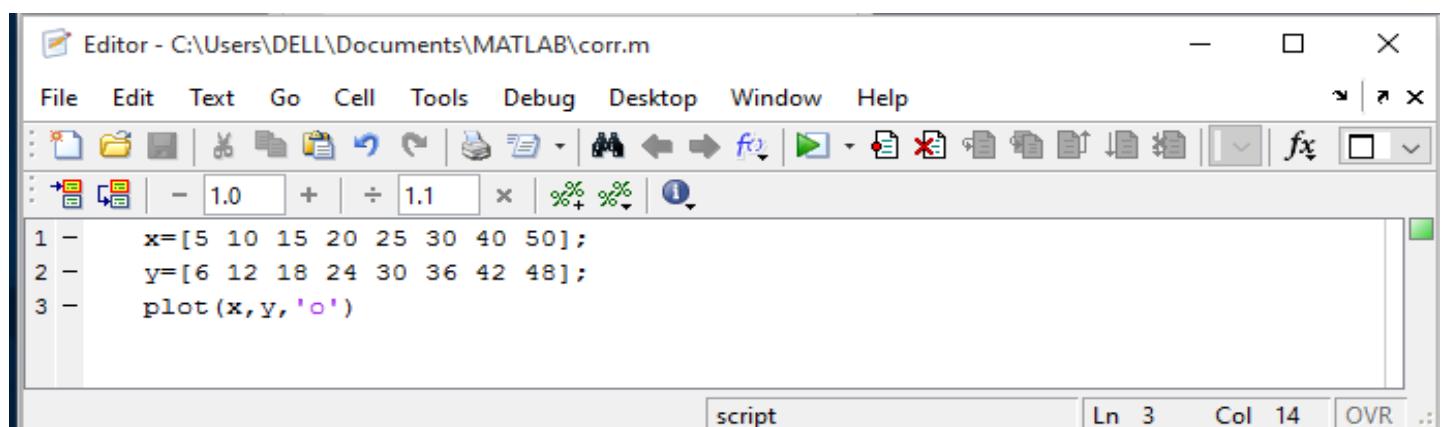
**Problem 11.** Consider the following bivariate data:

Use MATLAB to answer the following questions.

x	y
05	06
10	12
15	18
20	24
25	30
30	36
40	42
50	48

- (a) Find the correlation coefficient.
- (b) Draw a scatter diagram.
- (c) Fit polynomials of degree 2,3,4 to the data.

### **m FILES:**



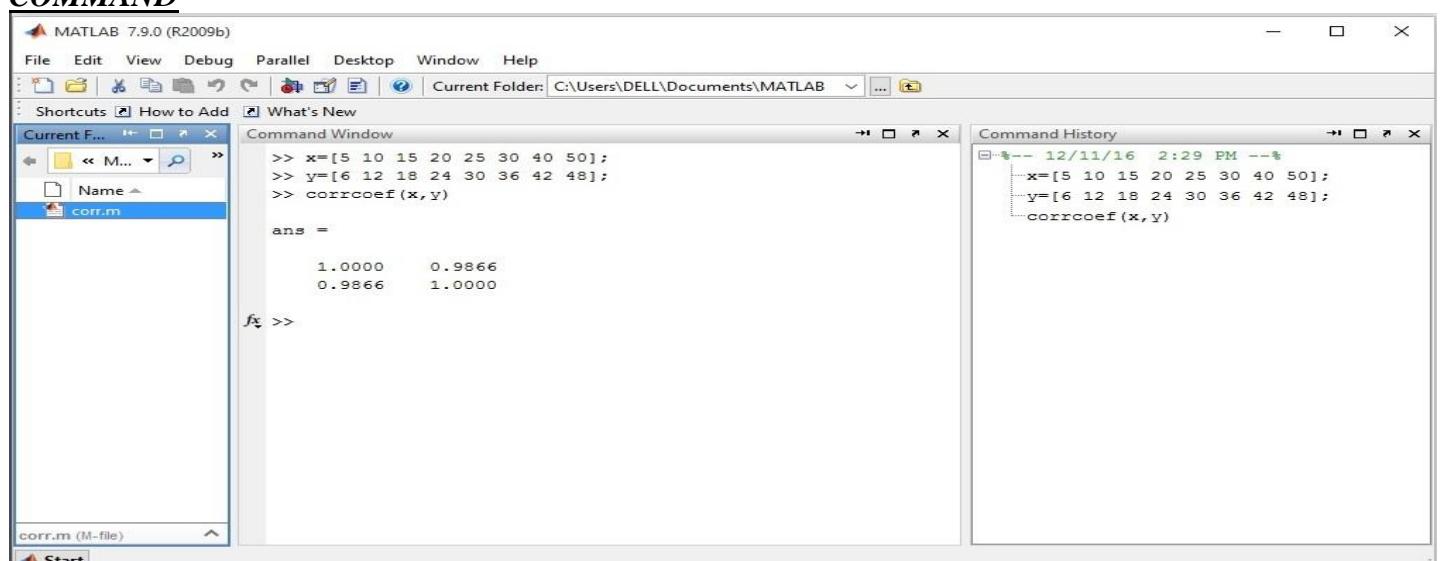
```

Editor - C:\Users\DELL\Documents\MATLAB\corr.m

File Edit Text Go Cell Tools Debug Desktop Window Help
script Ln 3 Col 14 OVR ...
1 - x=[5 10 15 20 25 30 40 50];
2 - y=[6 12 18 24 30 36 42 48];
3 - plot(x,y,'o')

```

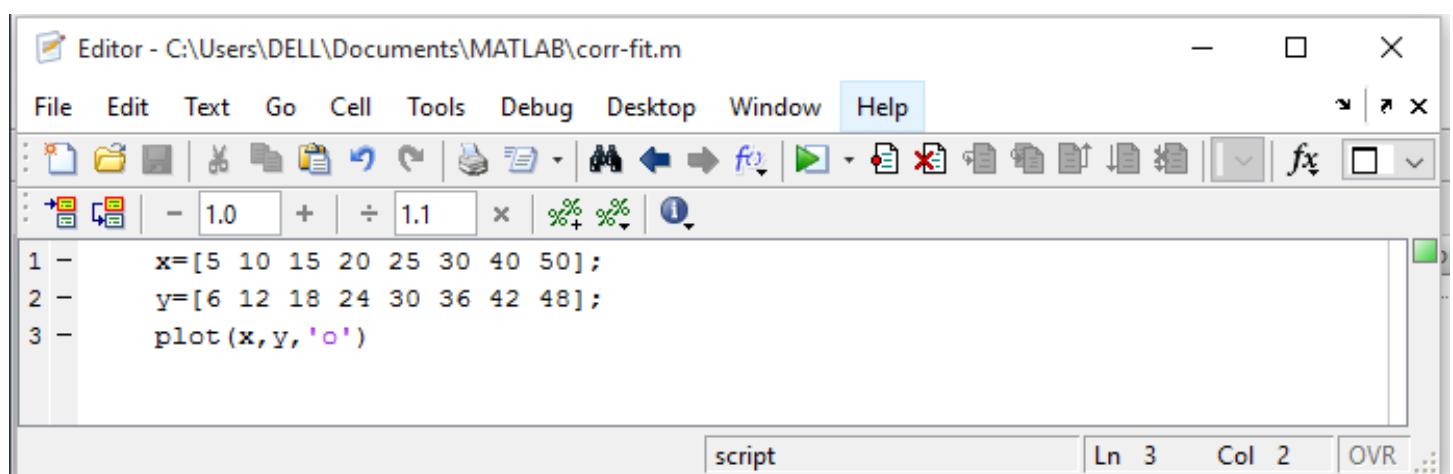
### **OUTPUT** **COMMAND**



```

MATLAB 7.9.0 (R2009b)
File Edit View Debug Parallel Desktop Window Help
Current Folder: C:\Users\DELL\Documents\MATLAB ...
Shortcuts How to Add What's New
Current F... <> M... <> Name & corr.m
>> x=[5 10 15 20 25 30 40 50];
>> y=[6 12 18 24 30 36 42 48];
>> corrcoef(x,y)
ans =
    1.0000    0.9866
    0.9866    1.0000
fxt >>

```

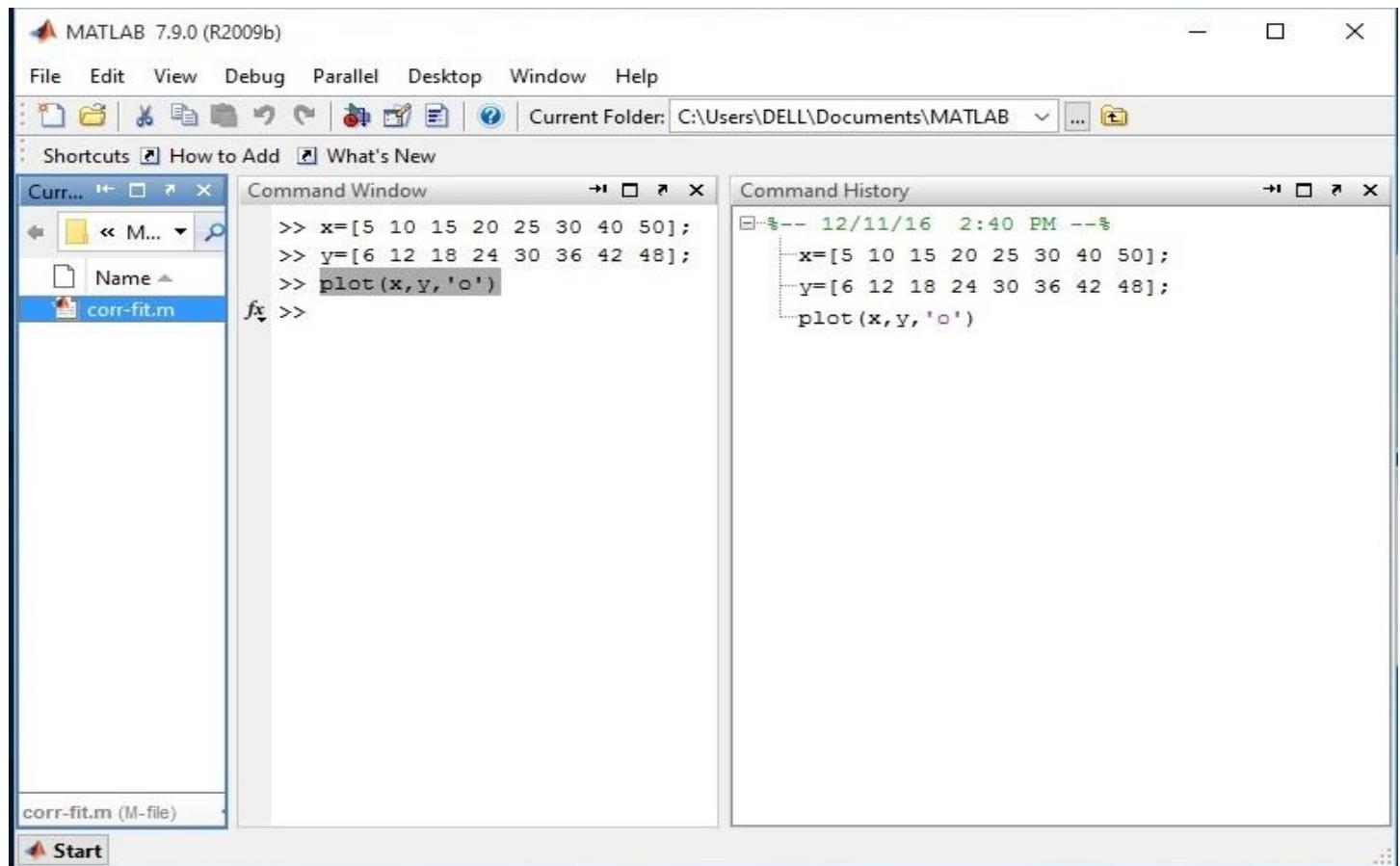
m FILES:


The screenshot shows the MATLAB Editor window with the file `corr-fit.m` open. The code in the editor is:

```

1 - x=[5 10 15 20 25 30 40 50];
2 - y=[6 12 18 24 30 36 42 48];
3 - plot(x,y,'o')

```

OUTPUT COMMAND:


The screenshot shows the MATLAB interface with the following windows open:

- Current Folder:** Shows the path `C:\Users\DELL\Documents\MATLAB`.
- Command Window:** Displays the command history for running the script.
- Command History:** Shows the commands entered and their output.

The command history in the Command Window shows:

```

>> x=[5 10 15 20 25 30 40 50];
>> y=[6 12 18 24 30 36 42 48];
>> plot(x,y,'o')
fx >>

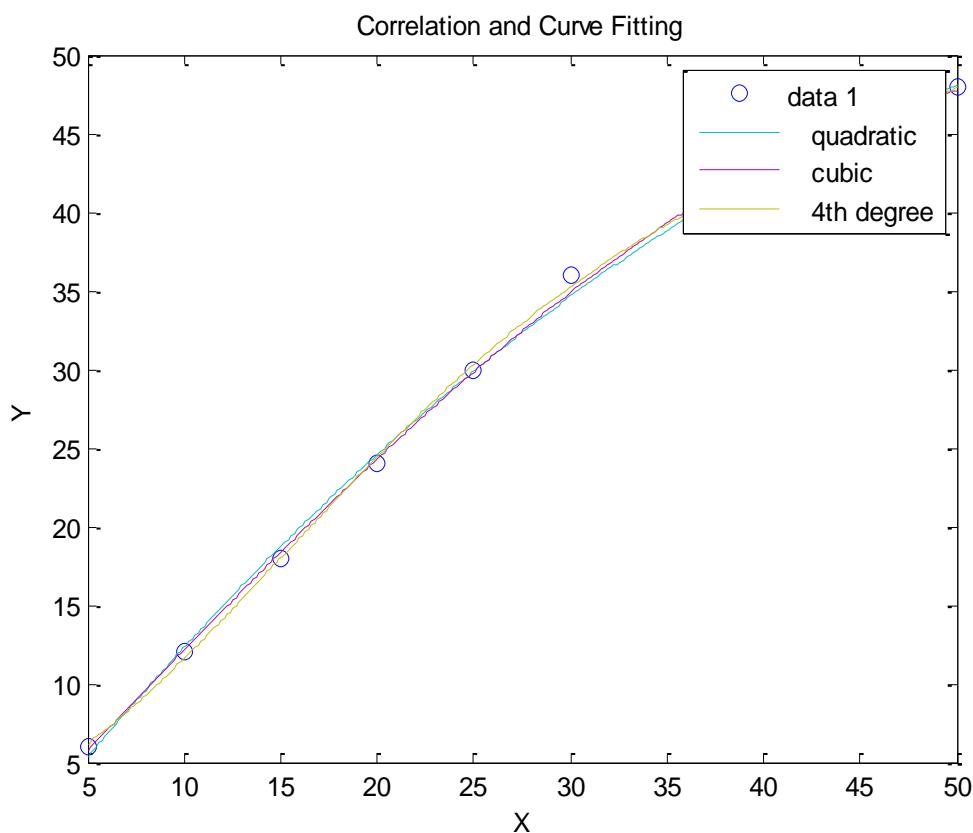
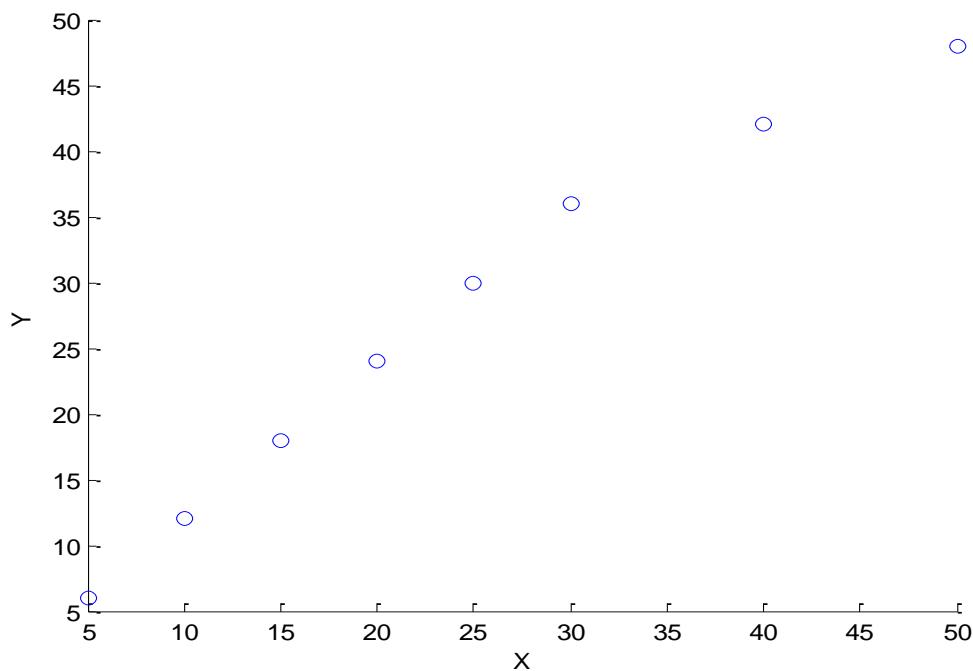
```

The Command History window shows the same commands and their output:

```

-->-- 12/11/16 2:40 PM --%
-> x=[5 10 15 20 25 30 40 50];
-> y=[6 12 18 24 30 36 42 48];
-> plot(x,y,'o')

```

**OUTPUT COMMAND:***scatter diagram*

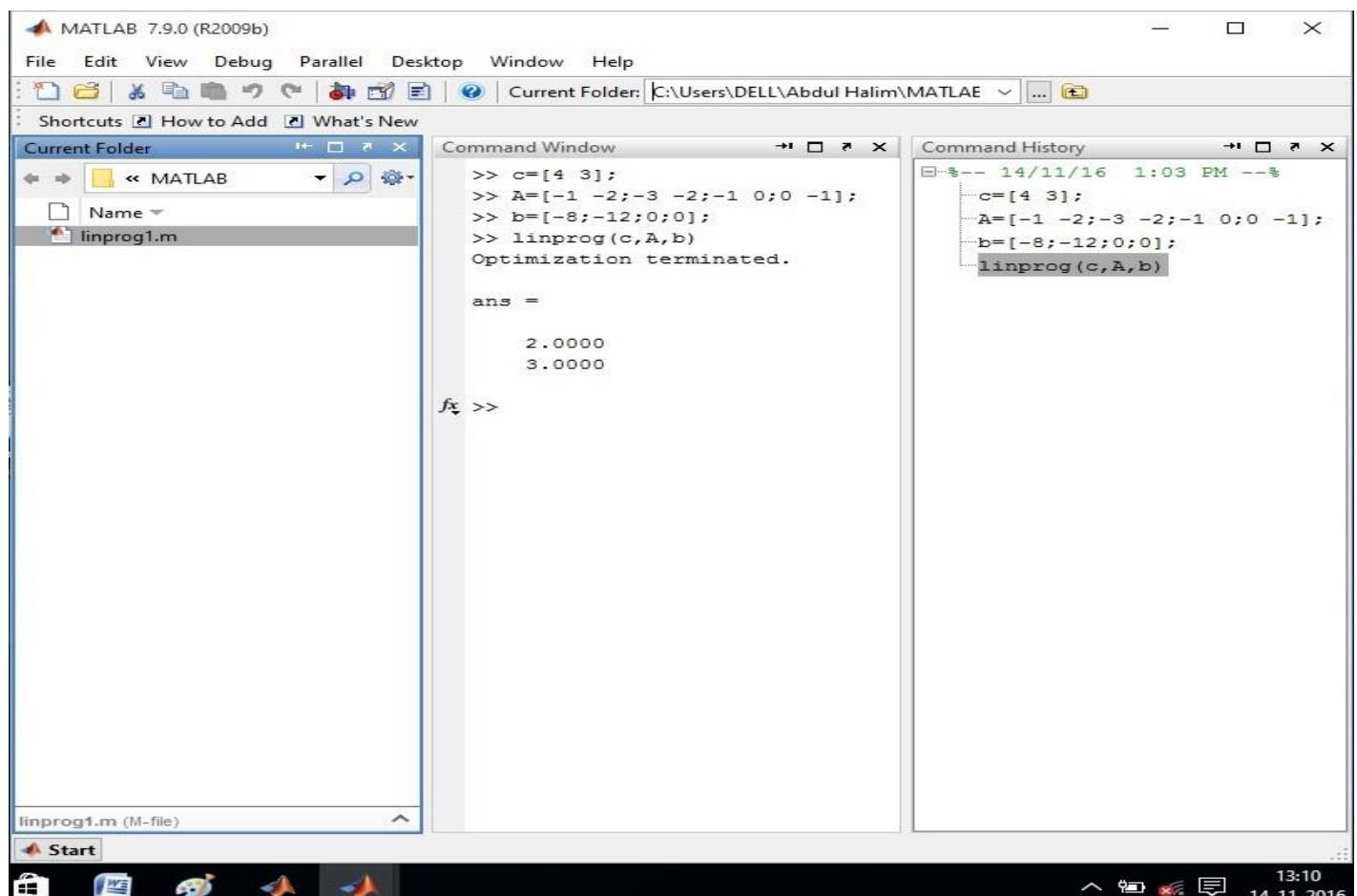
## *Solution of linear programming problems*

**Problem 12.** Use MATLAB to solve the following L.P.P:

(a) Maximize  $z = 4x_1 + 3x_2$   
 subject to  $3x_1 + x_2 \leq 15$ ,  
 $3x_1 + 4x_2 \leq 24$ ,  
 $x_1, x_2 \geq 0$ .

(b) Minimize  $z = 4x_1 + 3x_2$   
 subject to  $x_1 + 2x_2 \geq 8$ ,  
 $3x_1 + 2x_2 \leq 12$ ,  
 $x_1, x_2 \geq 0$ .

### OUTPUT COMMAND:



The screenshot shows the MATLAB 7.9.0 (R2009b) interface. The Command Window displays the following code and output:

```

>> c=[4 3];
>> A=[-1 -2;-3 -2;-1 0;0 -1];
>> b=[-8;-12;0;0];
>> linprog(c,A,b)
Optimization terminated.

ans =
    2.0000
    3.0000

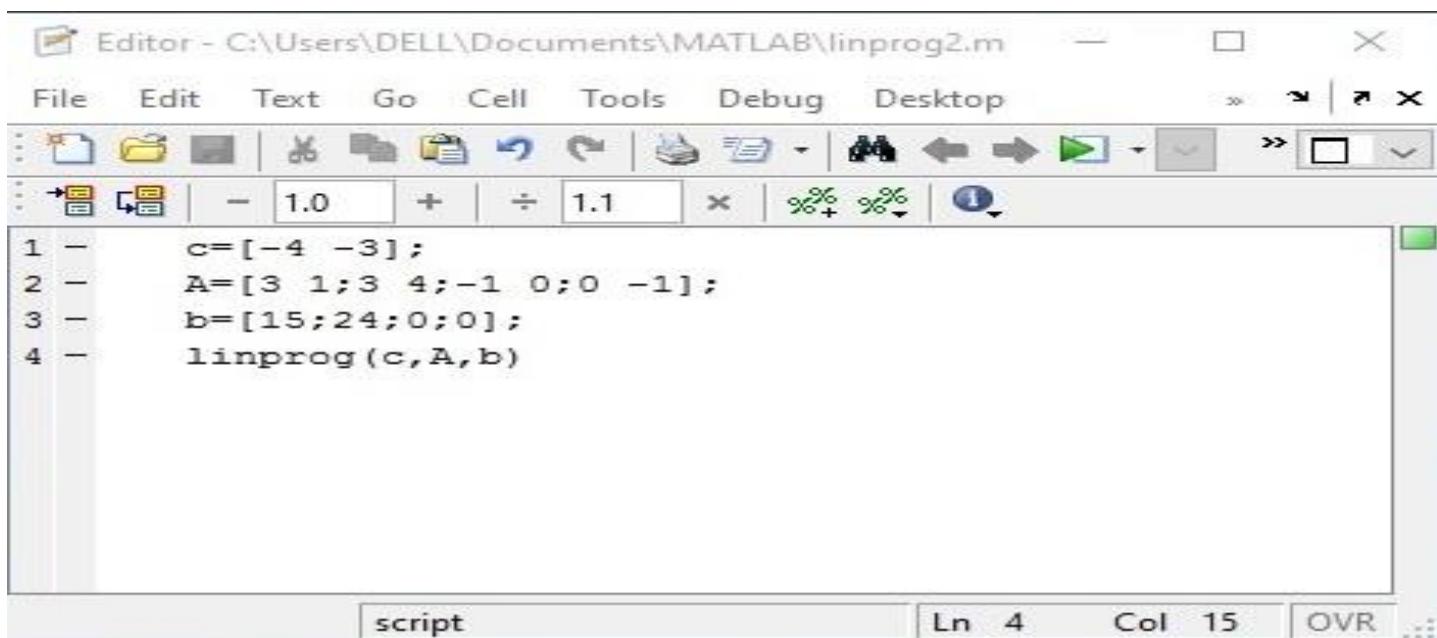
```

The Current Folder browser shows a file named "linprog1.m" selected. The Command History window shows the commands entered:

```

14/11/16  1:03 PM --%
c=[4 3];
A=[-1 -2;-3 -2;-1 0;0 -1];
b=[-8;-12;0;0];
linprog(c,A,b)

```

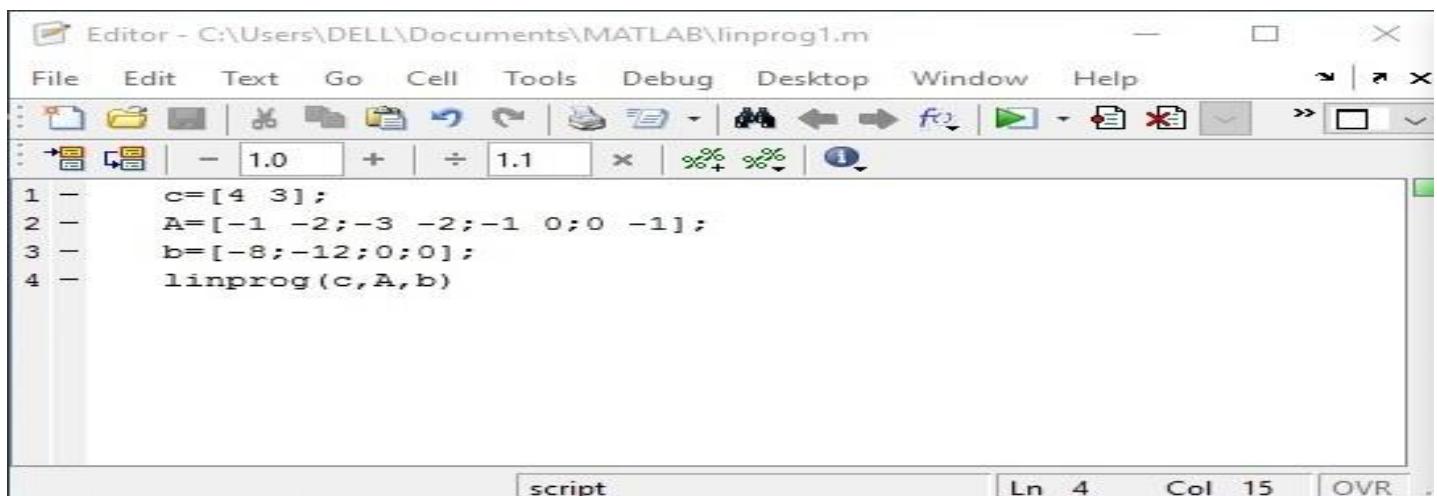
**m FILES:**


The screenshot shows the MATLAB Editor window with the file `linprog2.m` open. The code defines vectors `c`, `A`, and `b` and calls the `linprog` function.

```

1 -      c=[-4 -3];
2 -      A=[3 1;3 4;-1 0;0 -1];
3 -      b=[15;24;0;0];
4 -      linprog(c,A,b)

```

**m FILES:**


The screenshot shows the MATLAB Editor window with the file `linprog1.m` open. The code defines vectors `c`, `A`, and `b` and calls the `linprog` function.

```

1 -      c=[4 3];
2 -      A=[-1 -2;-3 -2;-1 0;0 -1];
3 -      b=[-8;-12;0;0];
4 -      linprog(c,A,b)

```

**OUTPUT COMMAND:**

The screenshot shows the MATLAB 7.9.0 (R2009b) interface. The Command Window displays the following code and output:

```
>> c=[4 3];
>> A=[-1 -2;-3 -2;-1 0;0 -1];
>> b=[-8;-12;0;0];
>> linprog(c,A,b)
Optimization terminated.

ans =
    2.0000
    3.0000

fx >>
```

The Command History window shows the commands entered:

```
14/11/16 1:03 PM --%
c=[4 3];
A=[-1 -2;-3 -2;-1 0;0 -1];
b=[-8;-12;0;0];
linprog(c,A,b)
```