1. Solving Higher Degree Equations

Question:

Solve the following equations:

    I. $3x - 2 = 0$

    II. $x^2 + x + 1 = 0$

    III. $2x^2 - 7x + 6 = 0$

    IV. $x^3 - x^2 + 4x + 6 = 0$

    V. $x^{12} - 6 = 0$

Aim:

To write a MATLAB program to solve higher degree equations.

Procedure:

- Click on MATLAB icon in the desktop.
- Select the command window.
- Enter the solving equation in command window.
- Then execute the equation in the command window.
- Then we get the output in the command window.

Program:

```
>> syms x;
>> equ1 = 3*x - 2 == 0

equ1 =

3*x - 2 == 0

>> ans1 = solve(equ1,x)

ans1 =

2/3

>> pretty(ans1)
2
-
3

>> equ2 = x^2 + x + 1 == 0;
>> ans2= solve(equ2,x)

ans2 =

 - (3^(1/2)*1i)/2 - 1/2
   (3^(1/2)*1i)/2 - 1/2

>> pretty(ans2)
/   sqrt(3) 1i    1 \
| - ---------- - - |
|        2        2 |
|                   |
|   sqrt(3) 1i    1  |
|   ---------- - -  |
\        2        2  /

>> equ3= 2*x^2 - 7*x + 6 ==0;
>> ans3= solve(equ3,x)

ans3 =

 3/2
   2

>> pretty(ans3)
/ 3 \
| - |
| 2 |
|   |
\ 2 /

>> equ4 = x^3 - x^2 + 4*x + 6 == 0
```

```
equ4 =

x^3 - x^2 + 4*x + 6 == 0

>> ans4= solve(equ4,x)

ans4 =

             -1
 1 - 5^(1/2)*1i
 1 + 5^(1/2)*1i

>> pretty(ans4)
/       -1        \
|                 |
| 1 - sqrt(5) 1i  |
|                 |
\ 1 + sqrt(5) 1i  /

>> equ5 = x^12 - 6 ==0

equ5 =

x^12 - 6 == 0

>> ans5= solve(equ5,x)

ans5 =

                          6^(1/12)
                         -6^(1/12)
                      -6^(1/12)*1i
                       6^(1/12)*1i
      6^(1/12)*(3^(1/2)/2 - 1i/2)
      6^(1/12)*(3^(1/2)/2 + 1i/2)
  6^(1/12)*((3^(1/2)*1i)/2 - 1/2)
  6^(1/12)*((3^(1/2)*1i)/2 + 1/2)
     -6^(1/12)*(3^(1/2)/2 - 1i/2)
     -6^(1/12)*(3^(1/2)/2 + 1i/2)
 -6^(1/12)*((3^(1/2)*1i)/2 - 1/2)
 -6^(1/12)*((3^(1/2)*1i)/2 + 1/2)

>> pretty(ans5)
/     1/12    \
|     6       |
|             |
|       1/12  |
|    - 6      |
|             |
|     1/12    |
```

```
|  - 6      1i  |
|               |
|    1/12       |
|   6      1i   |
|               |
|       #3      |
|               |
|       #4      |
|               |
|       #1      |
|               |
|       #2      |
|               |
|      -#3      |
|               |
|      -#4      |
|               |
|      -#1      |
|               |
\      -#2      /
```

where

```
          1/12 / sqrt(3) 1i    1 \
   #1 == 6     | ---------- - - |
              \      2        2 /


          1/12 / sqrt(3) 1i    1 \
   #2 == 6     | ---------- + - |
              \      2        2 /


          1/12 / sqrt(3)    1  \
   #3 == 6     | ------- - -i |
              \     2       2  /


          1/12 / sqrt(3)    1  \
   #4 == 6     | ------- + -i |
              \     2       2  /
```


>>

Conclusion:

Thus, the MATLAB program for solving higher degree equations was implemented successfully.

2. Solving System of Equations by Matrix Method and and Finding the Eigen Values and Eigen Vector of a Matrix of Order 4 x 4

Question:

Solve the following system of equation:

- 2w + y + z = - 3

 x + 2y − z = 2

- 3w + 2x + 4y + z = - 2

- w + x − 4y − 7z = - 19

Aim:

To write a MATLAB program to solve system of equation by matrix method and also find the eigen values and eigen vectors of the given system.

Procedure:

- Click the MATLAB icon in the desktop.
- From the MATLAB desktop click on **New Script** and then define the function on it and then save the file as **functionname** and with **m** extension by using **Save** button.
- In command window give the values of the matrices A and B.
- Call the function with required inputs.
- Then we get the output in the command window.

Program:

In Script Window:

```
%Solving System of Equation and Finding its Eigen values and Eigen Vectors

function [X,V, D] = Xval(A, B)
X = linsolve(A,B)      %Solve the matrix equation AX = B where B is a column vector
[V, D] = eig(A)     %Return matrix D of eigen values of matrix A and its eigen vectors V
end
```

In MATLAB Command Window

```
>> A = [-2 0 1 1; 0 1 2 -1; -3 2 4 1; -1 1 -4 -7];
```

```
>> B= [-3; 2; -2; -19];
>> A

A =

    -2     0     1     1
     0     1     2    -1
    -3     2     4     1
    -1     1    -4    -7

>> B

B =

    -3
     2
    -2
   -19

>> Xval(A, B);

X =

    3.0000
   -1.0000
    2.0000
    1.0000


V =

   -0.0863   -0.1984   -0.4659   -0.0271
   -0.5152    0.1984    0.4659    0.7937
   -0.8171   -0.1984   -0.4659   -0.4476
    0.2438    0.9391    0.5907    0.4110


D =

    4.6458         0         0         0
         0   -5.7321         0         0
         0         0   -2.2679         0
         0         0         0   -0.6458

>>
```

Conclusion:

   Thus, the MATLAB program for solving system of equation by matrix method and also for finding the eigen values and eigen vectors was implemented successfully.

3. Solving System of Non-Linear Equations

Question:

Solve the following system of equations:

$x\,y - 5y + 10 = 0$

$x^3 - y^2 = 2$

with initial condition x = 1, y = 1.

Aim:

To write a MATLAB program to solve system of non-linear equations.

Procedure:

- Click the MATLAB icon in the desktop.
- From the MATLAB desktop click on **New Script** and then define the function on it and then save the file as **functionname** and with **m** extension by using **Save** button.
- In command window, create a handle for the function by using @functionname and then give initial values for the independent variables.
- Use fsolve function to solve the given system of non-linear equation with required inputs.
- Then we get the output in the command window.

Program:

In Script window:

```
function fun = nonlinear(x)
fun(1) = x(1)*x(2)-5*x(2)+10;
fun(2) = x(1)^3-x(2)^2-2;
end
```

MATLAB Command Window

```
>> x =[1 1];
>> F = @nonlinear

F =

    @nonlinear

>> [xval, fval]= fsolve(F,x) %Solve the non linear equations

Equation solved.

fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.

<stopping criteria details>


xval =
```

```
     3.0000     5.0000


 fval =

     1.0e-08 *

     0.1116     0.0637

>> F([3 5]) %checking the solution that whether we get F(1) = 0 and F(2) = 0 for  x(1) ↙
               3 and x(2) =5


 ans =

      0     0

>>
```

Conclusion:

Thus, the MATLAB program for solving system of non-linear equation was implemented successfully.

4. Finding Second and Third Order Derivative of Different Functions

Question:

Find the second and third order derivatives of given function with respect to their variables:

(i)   f = t sin(5x)
(ii)   g = e^t x^2

Aim:

To write the MATLAB program for finding second and third order derivatives of the given functions.

Procedure:

- Click on MATLAB icon in desktop.
- Select the command window.
- Enter the given functions in the command window.
- Use **diff** function to find the second and third order derivatives of the given function.
- After executing the **diff** function, we get the output in the command window.

Program:

In Command Window:
```
>> syms x t;
>> f = t * sin(5*x);
>> f f =

t*sin(5*x)

>> derf1x=diff(f,x)

derf1x =
```

```
5*t*cos(5*x)

>> derf2x=diff(f,x,2)

derf2x =

-25*t*sin(5*x)

>> derf3x=diff(f,x,3)

derf3x =

-125*t*cos(5*x)

>> derf1t=diff(f,t)

derf1t =

sin(5*x)

>> derf2t=diff(f,t,2)

derf2t =

0

>> g=exp(t)*x^2

g =

x^2*exp(t)

>> derg2x=diff(g,x,2)

derg2x =

2*exp(t)

>> derg3t=diff(g,t,3)

derg3t =

x^2*exp(t)

>>
```

Conclusion:

Thus, the MATLAB program for finding second and third order derivatives of the given different functions was implemented successfully.

5. Finding the Integration of Different Functions with Limits

Question:

Find the integration of given functions:

(i) $f = x^7$ with lower limit a = 0 and upper limit b =1.
(ii) g = 1/x with lower limit a = 1 and upper limit b =2.
(iii) $h = \sqrt{x} \log (x)$ with lower limit a = 0 and upper limit b =1.
(iv) $z = e^{-x^2}$ with lower limit a = 0 and upper limit b = ∞.

Aim:

To write the MATLAB program for finding integration of the given functions.

Procedure:

- Click on MATLAB icon in desktop.
- Select the command window.
- Enter the given functions in the command window.
- Use **int** function to find integration of the given function.
- After executing the **int** function, we get the output in the command window.

Program:

In Command Window:

```
>> syms x
>> f=x^7;
>> a=0;
>> b=1;
>> F=int(f,a,b)

F =

1/8

>> g= 1/x;
>> G = int(g,[1 2])

G =

log(2)

>> h= log(x)*sqrt(x);
>> H = int(h,[0 1])

H =

-4/9

>> z = exp(-x^2)

z =

exp(-x^2)

>> Z = int(z,[0 inf])
```

```
Z =

pi^(1/2)/2

>> pretty(Z)
sqrt(pi)
--------
   2

>>
```

Conclusion:

Thus, the MATLAB program for finding the integration of the given different functions was implemented successfully.

6. Evaluation of Double and Triple Integrals

Question:

Find the integration of given functions:

(i) Integrate the following function $f(x,y) = \frac{1}{(\sqrt{x+y})(1+x+y)^2}$ over the triangular region bounded by $0 \leq x \leq 1$ and $0 \leq y \leq 1 - x$.

(ii) Integrate the function $f(x,y,z) = y\ sinx + z\ cosx$ over the region $0 \leq x \leq \pi, 0 \leq y \leq 1,$ and $-1 \leq z \leq 1.$

Aim:

To write the MATLAB program for evaluating double and triple integrals.

Procedure:

- Click on MATLAB icon in desktop.
- Select the command window.
- Enter the given functions in the command window.
- Use **intgral2** and **integral3** functions to evaluate double and triple integral respectively.
- After executing the **integral2** and **integral3** functions, we get the output in the command window.

Program:

In Command Window:

```
>> fun = @(x,y) 1./(sqrt(x+y).*(1+x+y).^2)

fun =

    @(x,y)1./(sqrt(x+y).*(1+x+y).^2)

>> ymax=@(x)1-x;
>> q = integral2(fun,0,1,0,ymax)

q =
```

```
    0.2854

>> fun1=@(x,y,z) y.*sin(x)+z.*cos(x)

fun1 =

    @(x,y,z)y.*sin(x)+z.*cos(x)

>> q = integral3(fun1,0,pi,0,1,-1,1)

q =

    2.0000

>>
```

Conclusion:

Thus, the MATLAB program for evaluating the double and triple integrals was implemented successfully.

7. Solving Ordinary Differential Equations with Initial Conditions

Question:

Solve the equation $\frac{dy}{dt} = ty$ with initial condition $y(0) = 2$.

Aim:

To write the MATLAB program for solving ODE with initial conditions.

Procedure:

- Click on MATLAB icon in desktop.
- Select the command window.
- Enter the given functions in the command window.
- Use **dsolve** functions to solve the equation with initial condition.
- After executing the **dsolve** functions, we get the output in the command window.

Program:

In Command Window:

```
>> syms y(t);
>> ode = diff(y,t)== t*y

ode(t) =

diff(y(t), t) == t*y(t)

>> ysol(t)= dsolve(ode)   %solving dy/dt without intial condition

ysol(t) =

C1*exp(t^2/2)

>> pretty(ysol(t))
      /  2 \
      | t  |
```

```
C1 exp| -- |
       \  2 /

>> cond = y(0)==2

cond =

y(0) == 2

>> ysol(t)= dsolve(ode, cond)     %solving dy/dt with intial condition y(0) = 2

ysol(t) =

2*exp(t^2/2)

>>
```

Conclusion:

Thus, the MATLAB program for solving the ordinary differential equation was implemented successfully.

8. Solving System of Ordinary Differential Equations

Question:

Solve the following system of ordinary differential equation

$$\frac{du}{dt} = 3u + 4v$$

$$\frac{dv}{dt} = -4u + 3v$$

$$with\ initial\ conditions\ u(0) = 0\ \ and\ v(0) = 1.$$

Aim:

To write the MATLAB program for solving system of ODE with initial conditions.

Procedure:

- Click on MATLAB icon in desktop.
- Select the command window.
- Enter the given functions in the command window.
- Use **dsolve** functions to solve the equation with initial condition.
- After executing the **dsolve** functions, we get the output in the command window.

Program:

In Command Window:

>> syms u(t) v(t);

>> ode1= diff(u) == 3*u +4*v;

>> ode2= diff(v) == -4*u + 3*v;

>> odes = [ode1; ode2]

odes(t) =

diff(u(t), t) == 3*u(t) + 4*v(t)

diff(v(t), t) == 3*v(t) - 4*u(t)

>> cond1 = u(0) == 0;

>> cond2 = v(0) == 1;

>> conds=[cond1; cond2]

conds =

 u(0) == 0

 v(0) == 1

>> [usol vsol]=dsolve(odes,conds)

usol =

sin(4*t)*exp(3*t)

vsol =

cos(4*t)*exp(3*t)

>>

Conclusion:

Thus, the MATLAB program for solving the system of ordinary differential equation was implemented without any error and the output is displayed in the command window successfully.

9.   Creating and Plotting 2-D and 3-D Graphs

Question:

i.   Create 2-D graph of y1 = sin(x) and y2 = cos(x) where $0 \leq x \leq 30\pi$.

ii.   Create 3-D graph of $x = \left(3 + \cos(\sqrt{32} \ t)\right) \cos t$, $y = \sin(\sqrt{32} \ t)$, and
      $z = \left(3 + \cos(\sqrt{32} \ t)\right) sint$ where $0 \leq t \leq 30\pi$.

Aim:

To write the MATLAB program for creating and plotting 2-D and 3-D graphs.

Procedure:

- Click the MATLAB icon in the desktop.
- From the MATLAB desktop click on **New Script** and then define the given function on it and then use **plot** and **plot3** functions for plotting 2-D and 3-D graph respectively after that save the file as **plot2d3d** and with **m** extension by using **Save** button.
- In command window, call the file by using the file name **plot2d3d.** Then we get the output in the figures window
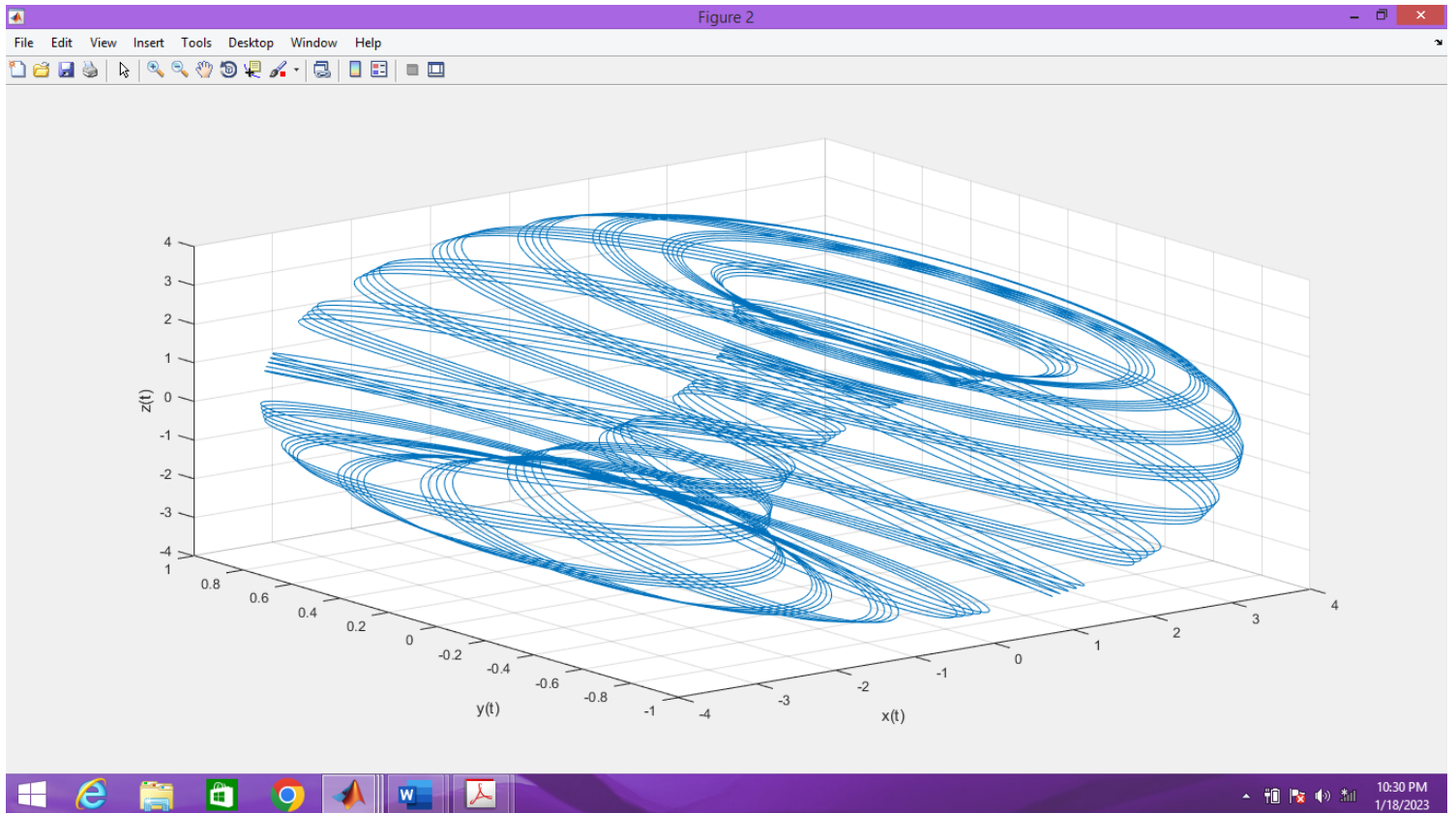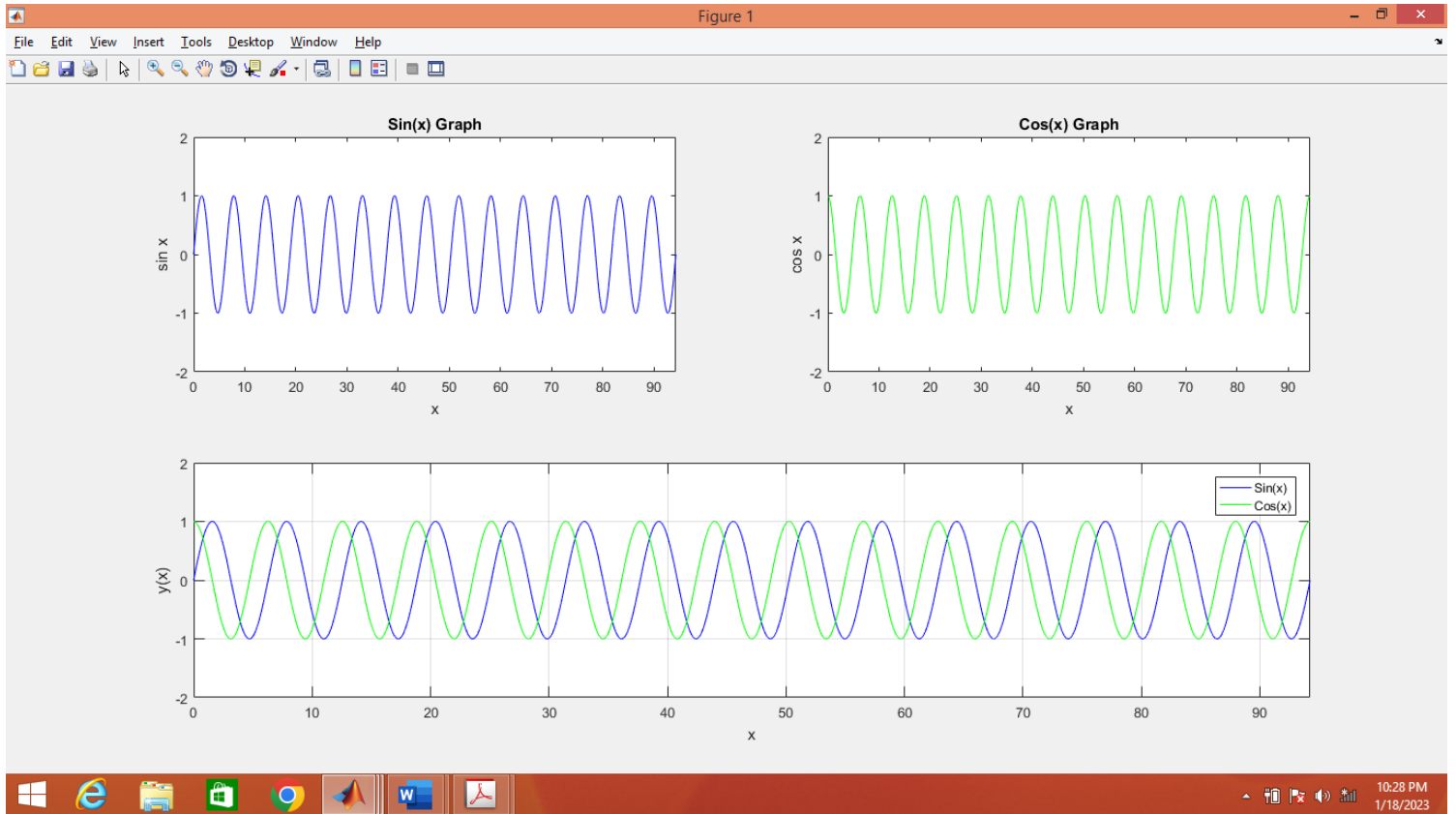
Program:

In Script Window:

```
%Ploting 2D graph
x = 0:pi/500:30*pi ;
y1= sin(x);
y2= cos(x);
figure;
subplot(2,2,1);
plot(x,y1,'b'), xlabel('x'), ylabel('sin x'),title('Sin(x) Graph'), axis([0 30*pi -2 2])
subplot(2,2,2);
plot(x,y2,'g'), xlabel('x'), ylabel('cos x'),title('Cos(x) Graph'), axis([0 30*pi -2 2])
subplot(2,2,[3 4]);
plot(x,y1,'b',x, y2, 'g'),xlabel('x'), ylabel('y(x)'), legend('Sin(x)','Cos(x)')
axis([0 30*pi -2 2])
grid on;

%Plotting 3D graph
t = x;
x = (3+cos(sqrt(32)*t)).*cos(t);
y = sin(sqrt(32)*t);
z = (3+cos(sqrt(32)*t)).*sin(t);
figure;
plot3(x,y,z)
grid on;
xlabel('x(t)');
ylabel('y(t)');
zlabel('z(t)');
```

In Command Window:

```
>>plot2d3d
```

Conclusion:

Thus, the MATLAB program for creating and plotting 2-D and 3-D graph was implemented without any error and the output is displayed in the figure window successfully.

10. Solving Linear Programming Problems

Question:

$Maximize\ z =\ 4x_1 + 3x_2,$

Subject to the constraints,

$$2x_1 + x_2 \le 1000$$
$$x_1 + x_2 \le 800$$
$$0 \le x_1 \le 400$$
$$0 \le x_2 \le 700.$$

Aim:

To write the MATLAB program for solving given linear programming problem.

Procedure:

- Click the MATLAB icon in the desktop.
- From the MATLAB desktop click on **New Script** and then define given input details on it and then use **linprog** function for solving given linear programming problem after that save the file as **lpp** and with **m** extension by using **Save** button.
- In command window, call the file by using the file name **lpp.** Then we get the output in the command window.

Program:

In Script Window:

```
% Solving Linear Programming Problems
f = [4 3]
A = [ 2 1; 1 1]
b = [1000 800]
lb = [0 0];
ub = [400 700];
Aeq = [];
beq = [];
[x fval] = linprog(-f,A,b,Aeq, beq, lb, ub);
fval = -fval;
x
fval
```

In command window:

>> lpp

f =

   4   3

A =

   2   1

   1   1

b =

    1000   800

Optimization terminated.

x =

  200.0000

  600.0000

fval =

   2.6000e+03

>>

Conclusion:

      Thus, the MATLAB program solving linear programming problem was implemented without any error and the output is displayed in the command window successfully.