1. MATLAB - BASIC OPERATIONS

```
clc
clear all
 Close all
rv=[1 2 3 4 5]       % row vector
cv=rv'                    % ' - Transpose
cv2=[1;2;3;4;5]  % column vector
lrv=length(rv);
srv=size(rv);
A=[1 2;3 4]

dA=det(A)
rA=rank(A)
lA=tril(A)
uA=triu(A)
dA=diag(A)

B=[4 5;8 12]
C=A+B
D1=A*B    % * - matrix multiplication
D2=A.*B   % .* - component-wise multiplication
E=inv(A)  % inv(A) - inverse of matrix A
F1=B'
H=A.^3
A=[1 2 3 4;5 6 7 8;9 10 11 12;13 14 15 16]
size(A)
```

Slicing a Matrix

```
A(2,:)=[]  % removes the elements of row 2
size(A)
A(:,3)=[]  % removes the elements of column 3
size(A)
sm1=A(2:3,2:3)
sm2=A([1 3],[2 1])
```

Transpose operations

```
x=[1 2 3 4 5]
y=[2 2 2 3 3]
z1=x*y'
z2=x'*y
z3=x.*y
z4=x.^y
z5=x./y
```

2. SOLVING SYSTEM OF EQUATIONS

```
A=[4 5;7 8]
b=[7 21]'
```

```
x=A\b

set(h,'color','r')
```

5. PLOTTING OF CURVES AND SURFACES

Making 3D surface plots, contour plots, and gradient plots in Matlab is slightly more complicated than making simple line graphs, but we will present some examples that, with simple modifications, should enable you to create most of the pictures that you will need.

Matlab provides a variety of functions for displaying data as 2-D or 3-D graphics for 2-D graphics; the command plots vector x1 versus vector y1, vector x2 versus vector y2, etc. on the same graph. Other commands for 2-D graphics are: polar, bar, stairs, loglog, semilogx and semilogy.

For 3-D graphics, the most commonly used commands are:

plot3(x1,y1,z1,'linestyle', x1,y1,z1,'linestyle',…..)

contour(x,y,z),mesh(x,y,z),surf(x,y,z)

The first statement is a 3-D analogue of plot() and plots lines and points in 3-D. the second statement produces contour plots of the matrix z using the vectors x and y to control the scaling on the x- and y- axes. For surface or meshplots, you use the third statement where x, y are vectors or matrices and z is a matix. Other commands available for 3-D graphics are: pcolor, image, contour3, fill3, cylinder, and sphere.

A. Mathematical form:

- Importance visualizing curves and surfaces
- Draw the curve for the given function f(x)
- Draw the surface for the given function.

B. MATLAB Syntax used:

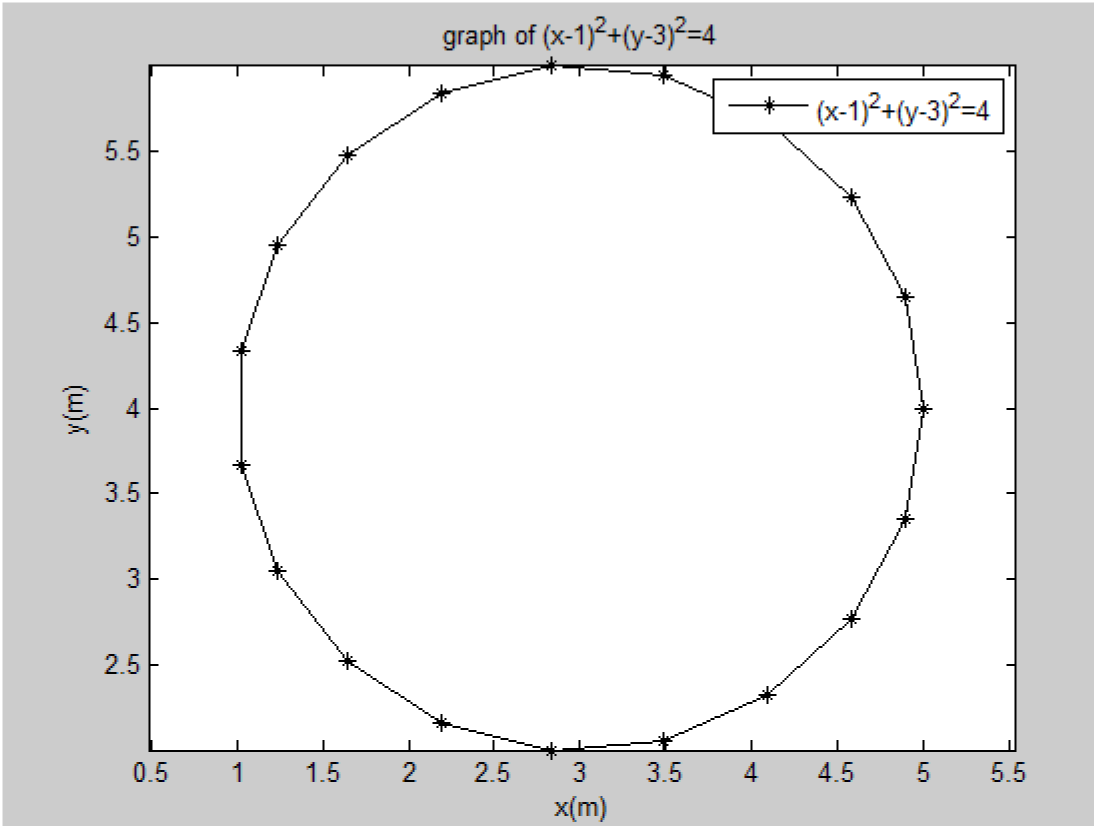
Command	Description
plot(y)	Plots the columns of Y versus the index of each value when Y is a real number. For complex Y, plot(Y) is equivalent to plot(real(Y),imag(Y))
plot3(X1,Y1,Z1)	Displays a three-dimensional plot of a set of data points
surf(Z)	Creates a three-dimensional shaded surface from the z components in matrix Z, using x = 1:n and y = 1:m, where [m,n] = size(Z)
ezsurf(fun)	Creates a graph of fun(x,y) using the surf function. fun is plotted over the default domain: $-2\pi < x < 2\pi$, $-2\pi < y < 2\pi$.
y = linspace(a,b,n)	Generates a row vector y of n points linearly spaced between and including a and b.

mesh(X,Y,Z)	Draws a wireframe mesh with color determined by Z so color is proportional to surface height
[X,Y] = meshgrid(x,y)	Transforms the domain specified by vectors x and y into arrays X and Y, which can be used to evaluate functions of two variables and three-dimensional mesh/surface plots.
subplot(m,n,p)	Breaks the figure window into an m-by-n matrix of small axes, selects the pth axes object for the current plot, and returns the axes handle.
colormap(map)	Sets the colormap to the matrix map. If any values in map are outside the interval [0 1], you receive the error Colormap must have values in [0, 1].
Prism	Repeats the six colors red, orange, yellow, green, blue, and violet
Flag	Consists of the colors red, white, blue, and black. This colormap completely changes color with each index increment
hsv	Varies the hue component of the hue-saturation-value color model. The colors begin with red, pass through yellow, green, cyan, blue, magenta, and return to red.
shading flat	Controls the color shading of surface and patch graphics objects.

1. PARAMETRIC PLOTTING

```
clc
clearall
close all
t = linspace(0, 2*pi,20);
x = 3+2*cos(t);
y = 4+2*sin(t);
plot(x,y,'k-*')
axis equal
xlabel('x(m)')
ylabel('y(m)')
title('graph of (x-1)^2+(y-3)^2=4')
legend('(x-1)^2+(y-3)^2=4')
```

OUTPUT

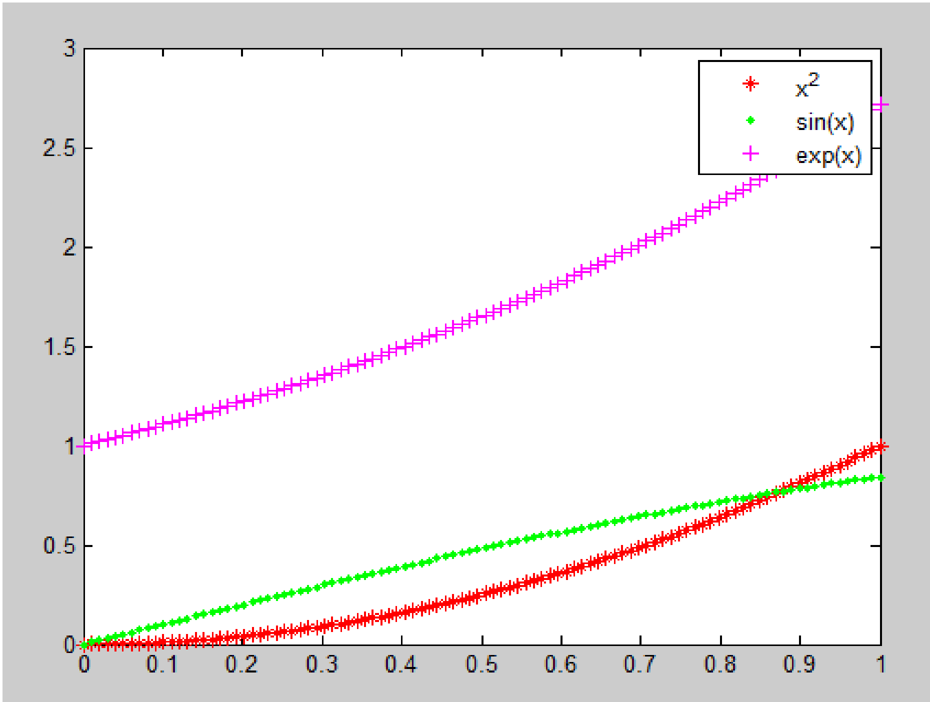


2. MULTIPLE PLOTS IN A FIGURE WINDOW (USING COMMAND HOLD ON)

```
clc
clear all
close all
x = linspace(0,1)
plot(x,x.^2,'r*')
hold on
plot(x,sin(x),'g.')
plot(x,exp(x),'m+')
legend('x^2','sin(x)','exp(x)')
```

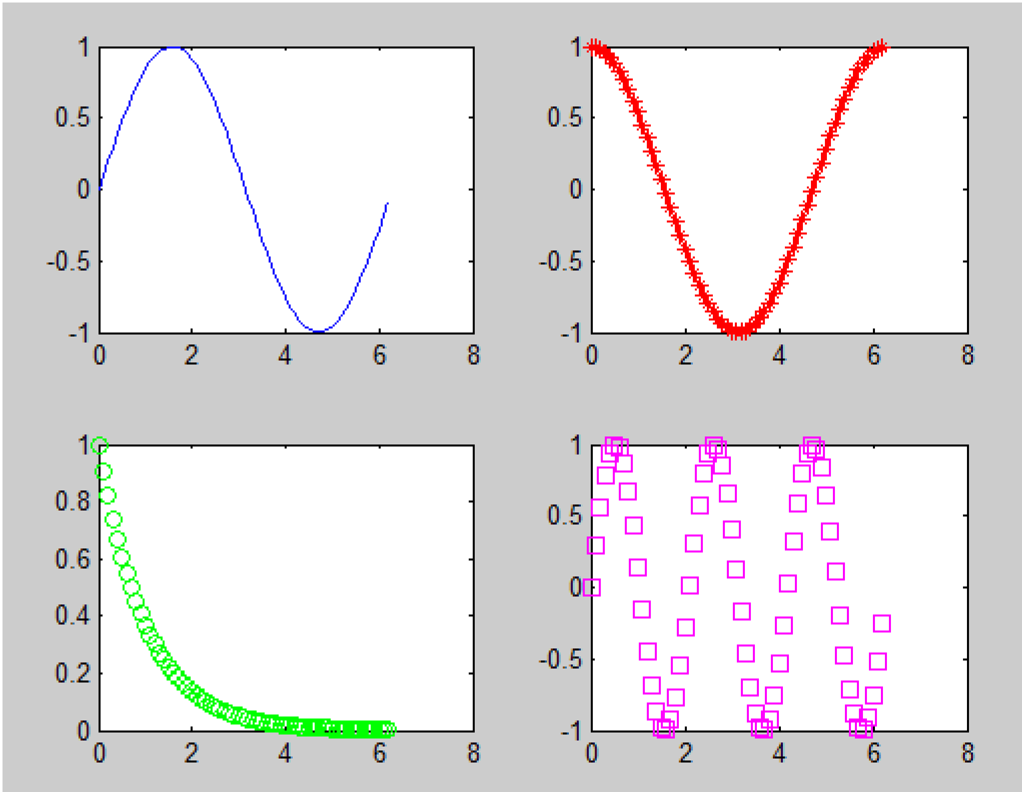
```
OUTPUT
x =

  Columns 1 through 9
```

```
         0    0.0101    0.0202    0.0303    0.0404    0.0505
0.0606    0.0707    0.0808

  Columns 10 through 18

    0.0909    0.1010    0.1111    0.1212    0.1313    0.1414
0.1515    0.1616    0.1717

  Columns 19 through 27

    0.1818    0.1919    0.2020    0.2121    0.2222    0.2323
0.2424    0.2525    0.2626

  Columns 28 through 36

    0.2727    0.2828    0.2929    0.3030    0.3131    0.3232
0.3333    0.3434    0.3535

  Columns 37 through 45

    0.3636    0.3737    0.3838    0.3939    0.4040    0.4141
0.4242    0.4343    0.4444

  Columns 46 through 54

    0.4545    0.4646    0.4747    0.4848    0.4949    0.5051
0.5152    0.5253    0.5354

  Columns 55 through 63

    0.5455    0.5556    0.5657    0.5758    0.5859    0.5960
0.6061    0.6162    0.6263

  Columns 64 through 72

    0.6364    0.6465    0.6566    0.6667    0.6768    0.6869
0.6970    0.7071    0.7172

  Columns 73 through 81

    0.7273    0.7374    0.7475    0.7576    0.7677    0.7778
0.7879    0.7980    0.8081

  Columns 82 through 90

    0.8182    0.8283    0.8384    0.8485    0.8586    0.8687
0.8788    0.8889    0.8990

  Columns 91 through 99

    0.9091    0.9192    0.9293    0.9394    0.9495    0.9596
0.9697    0.9798    0.9899

  Column 100

    1.0000
```



3. MULTIPLE GRAPHS IN A FIGURE WINDOW BY USING SUBPLOT

```
clc
clear all
close all
x=0:0.1:2*pi;
subplot(2,2,1)
plot(x,sin(x));
subplot(2,2,2)
plot(x,cos(x),'r-*');
subplot(2,2,3)
plot(x,exp(-x),'go')
subplot(2,2,4);
plot(x,sin(3*x),'ms')
```

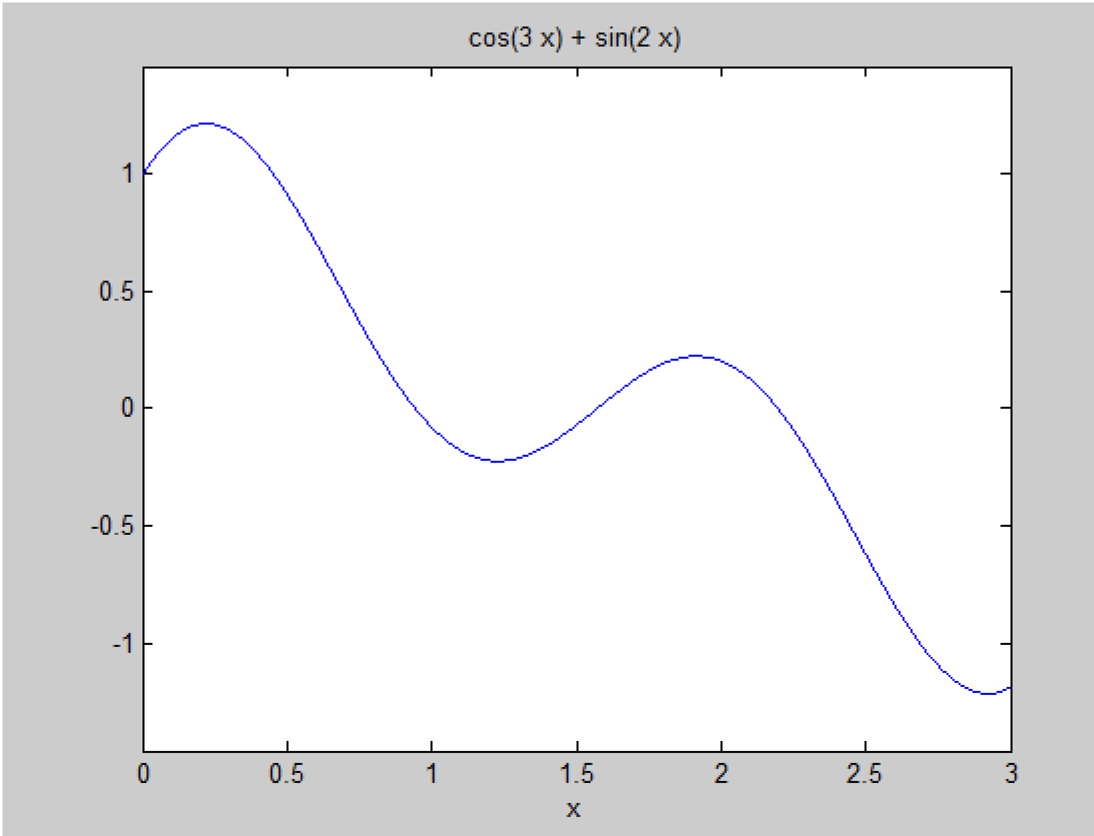


3. GRAPH OF A CURVE THROUGH EZPLOT COMMAND

```
clc
clear all
symsx         % Declaring the parameters as a symbolic object
f=sin(2*x)+cos(3*x)
figure(1)
ezplot(f)
figure(2)
ezplot(f,[0,3])
```

OUTPUT

```
f =

cos(3*x) + sin(2*x)
```



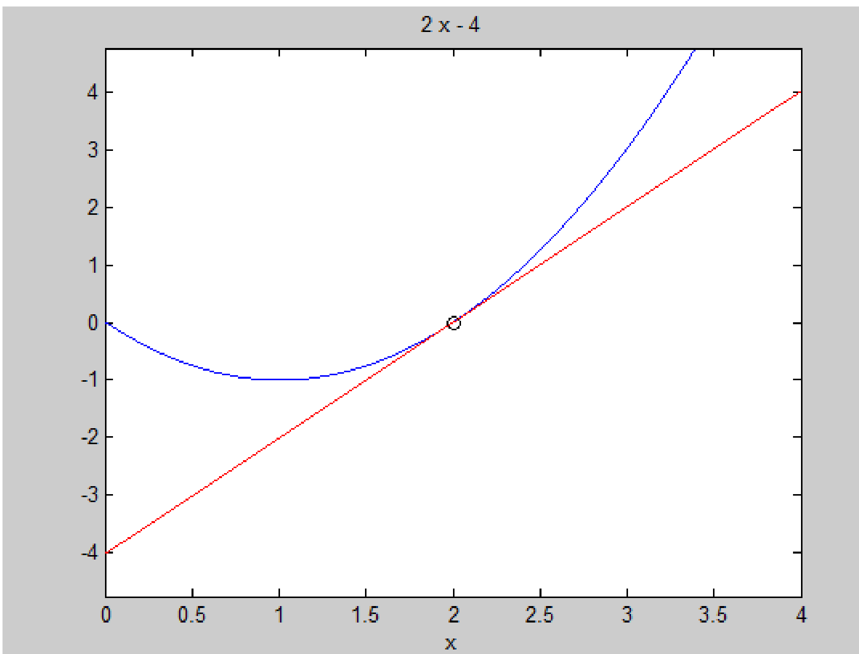
4. GRAPH OF A CURVE AND ITS TANGENT LINE IN THE NEIGHBOURHOOD D OF A POINT.

```
symsx
%y=input('enter the function f in terms of x:')% Example, Try the
function y=x^2-2*x;
y=x^2-2*x;
%x1 = input('Enter x value at which tangent  : '); % Example, Try
the point x1 =  2
x1=2;
D=[x1-2 x1+2]% Region about x1 (or Neighbourhood of x1)
ezplot(y,D)   % graph of the curve in D
hold on
```

%Equation of the tangent line passing through x1.

```
yd = diff(y,x); % Differentiation in MATLAB
slope = subs(yd,x,x1); % Finding the slope at x1
y1 = subs(y,x,x1);      % Finding the value of the function at the
given point
plot(x1,y1,'ko')        % plot the point

Tgt_line = slope*(x-x1)+y1 % Tangent Line Equation at the given
point
h = ezplot(Tgt_line,D);     % Plotting the Tangent Line
set(h,'color','r')
```

OUTPUT

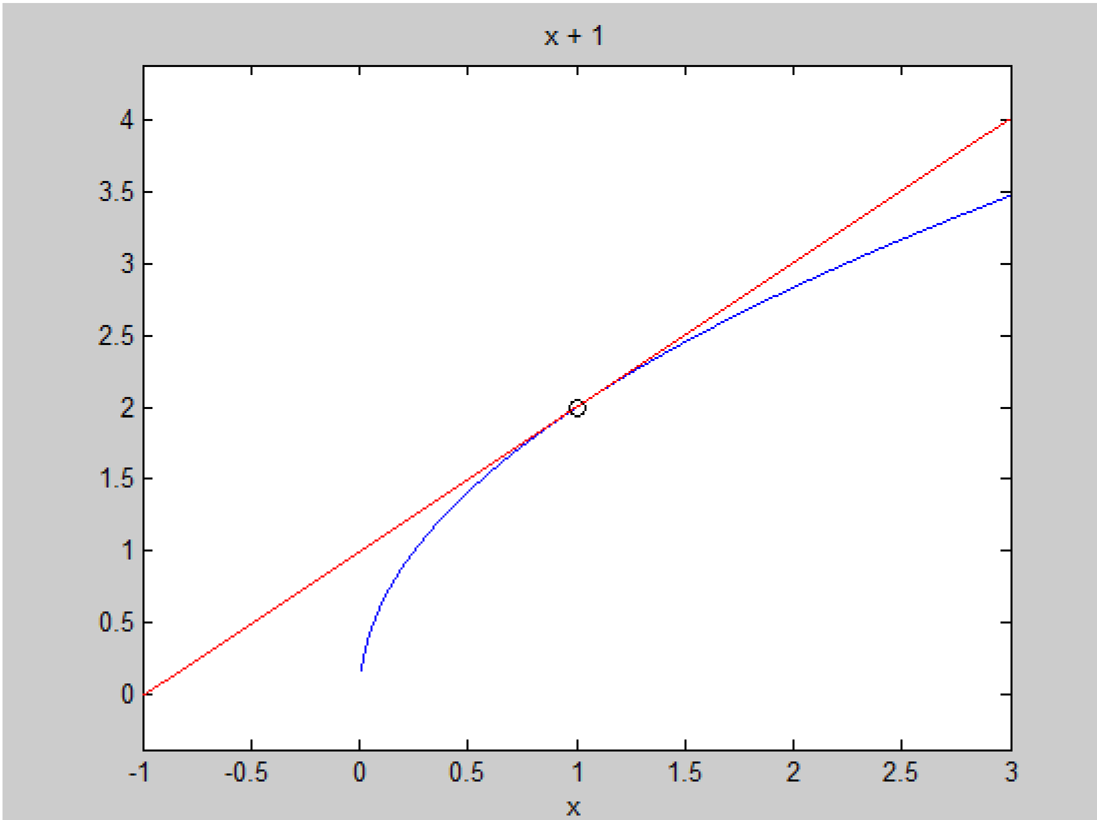


1) Find the equation of tangent to the curve $y = 2\sqrt{x}$ at (1,2).

```
syms x
y= 2*sqrt(x);
x1=1;
D=[x1-2 x1+2]

D =

   -1    3

ezplot(y,D)
hold on
yd = diff(y,x);
slope = subs(yd,x,x1);
y1 = subs(y,x,x1);
```

```
plot(x1,y1,'ko')
Tgt_line = slope*(x-x1)+y1
Tgt_line = x + 1
 h = ezplot(Tgt_line,D);
 set(h,'color','r')
```

OUTPUT



2) Find the equation of tangent to the curve $y = x^3$ at (-2,-8).

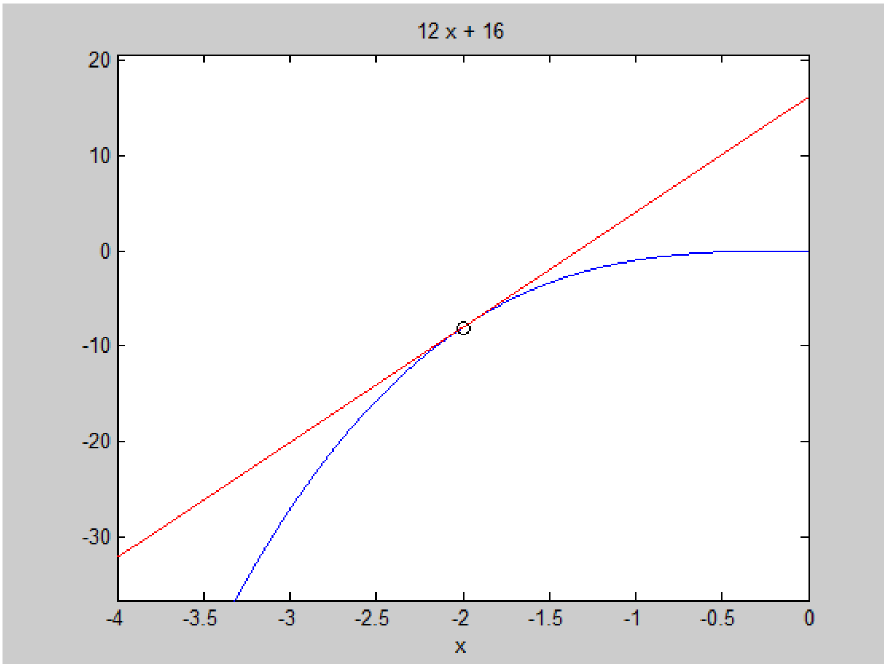
```
syms x
y=x^3;
x1=-2;
D=[x1-2 x1+2]

D =

   -4     0

ezplot(y,D)
hold on
yd = diff(y,x);
slope = subs(yd,x,x1);
y1 = subs(y,x,x1);
plot(x1,y1,'ko')
Tgt_line = slope*(x-x1)+y1
```

Tgt_line =

12*x + 16

h = ezplot(Tgt_line,D);
set(h,'color','r')

OUTPUT



5. EXTREMA OF A SINGLE VARIABLE FUNCTION.

```
syms x real
f= input('Enter the function f(x):');
fx= diff(f,x)
c = solve(fx)
cmin = min(double(c));
cmax = max(double(c));
ezplot(f,[cmin-2,cmax+2])
hold on
fxx= diff(fx,x)
for i = 1:1:size(c)
    T1 = subs(fxx, x ,c(i) );
    T3= subs(f, x, c(i));
 if (double(T1)==0)
    sprintf('The point x is %d inflexion point',double (c(i)))
    else
         if (double(T1) < 0)
    sprintf('The maximum point x is %d', double(c(i)))
    sprintf('The value of the function is %d', double (T3))
         else
```

```
    sprintf('The minimum point x is %d', double(c(i)))
                sprintf('The value of the function is %d', double
(T3))
         end
 end
plot(double(c(i)), double(T3), 'r*', 'markersize', 15);
end
pause
h=ezplot(fx,[cmin-2,cmax+2])
set(h,'color','r')
hold on
pause
e=ezplot(fxx,[cmin-4,cmax+4])
set(e,'color','g')
hold off
```

OUTPUT

Enter the function f(x):x^4

fx =

4*x^3

c =

0
0
0

fxx =

12*x^2

ans =

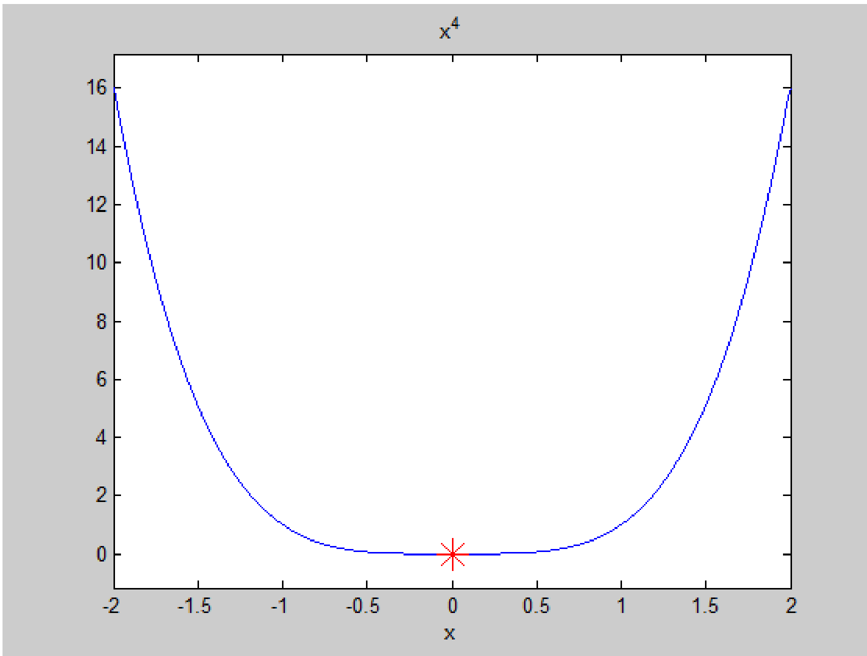
The point x is 0 inflexion point

ans =

The point x is 0 inflexion point

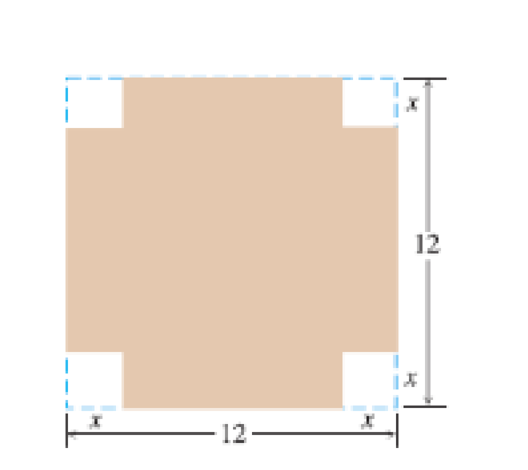
ans =

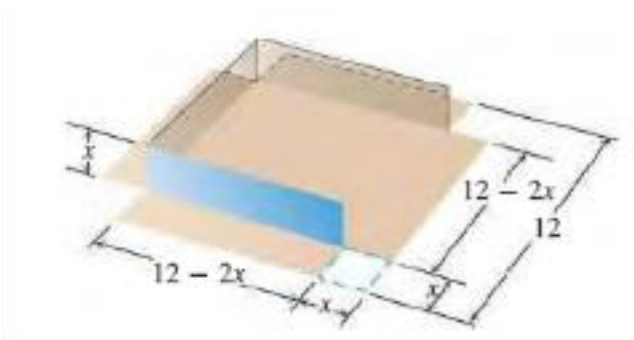
The point x is 0 inflexion point



Practice Problems:

1. An open-top box is to be made by cutting small congruent squares from the comers of a 12-in.-by-12-in. sheet of tin and bending up the sides. How large should the squares cut from the corners be to make the box hold as much as possible?





$$V(x) = x(12-2x)^2 = 4x^3 - 48x^2 + 144x$$

Enter the function f(x):4*x^3-48*x^2+144*x

fx =12*x^2 - 96*x + 144

c =

2

6

fxx =24*x - 96

ans =The maximum point x is 2

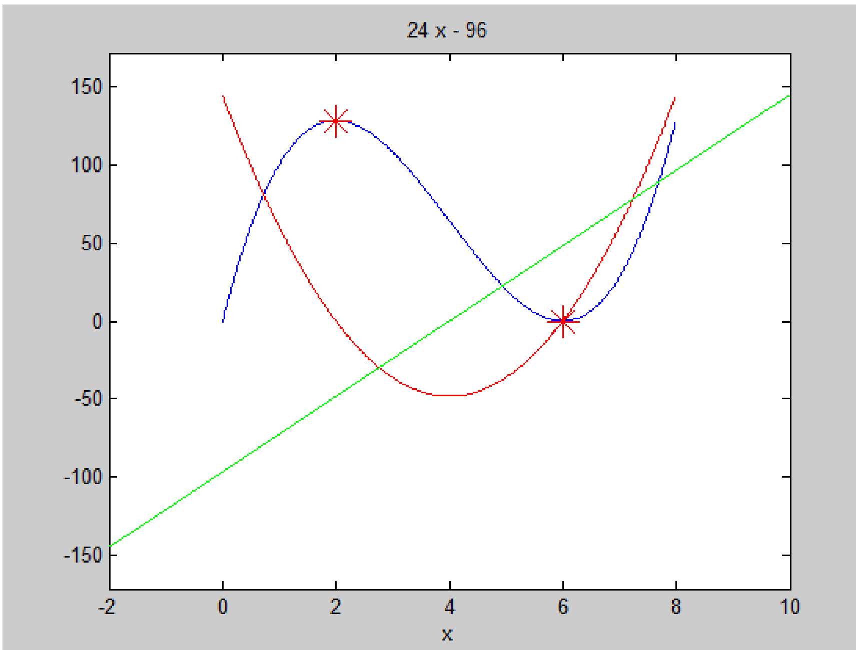
ans =The value of the function is 128

ans =The minimum point x is 6

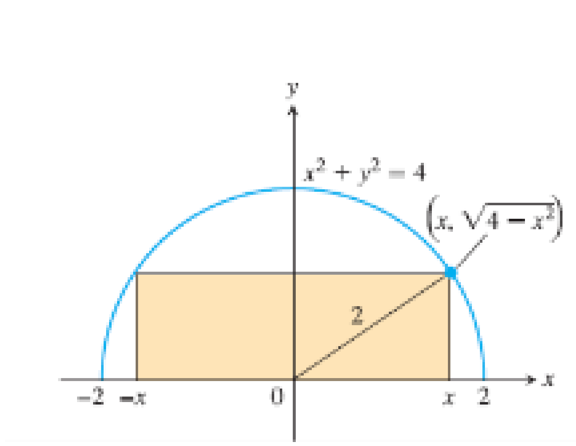
ans =The value of the function is 0

h = 179.0059

e =180.0059



2. A rectangle is to be inscribed in a semicircle of radius 2. What is the largest area the rectangle can have, and whatare its dimensions?



Length :2x, Height: $\sqrt{4-x^2}$ Area: 2x $\sqrt{4-x^2}$. A(x)= 2x $\sqrt{4-x^2}$.

OUTPUT

Enter the function f(x): 2*x*sqrt(4-x^2)

fx =2*(4 - x^2)^(1/2) - (2*x^2)/(4 - x^2)^(1/2)

c = 2^(1/2)

-2^(1/2)

fxx =- (2*x^3)/(4 - x^2)^(3/2) - (6*x)/(4 - x^2)^(1/2)

ans =The maximum point x is 1.414214e+00

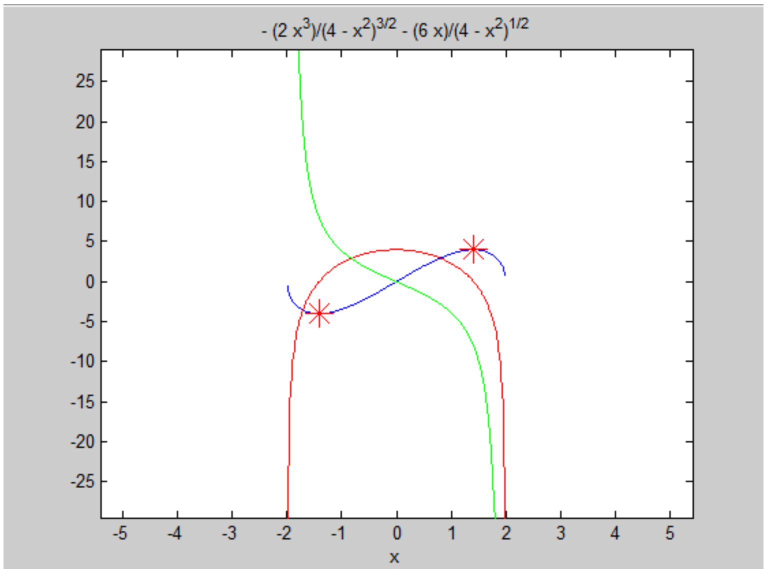
ans =The value of the function is 4

ans =The minimum point x is -1.414214e+00

ans =The value of the function is -4

h = 179.0071

e = 180.0071


- (2 x3)/(4 - x2)3/2 - (6 x)/(4 - x2)1/2
25
20
15
10
5
0
-5
-10
-15
-20
-25
-5 -4 -3 -2 -1 0 1 2 3 4 5
x