19CHU111 **MATHEMATICS-I PRACTICAL** **4H – 2C**

Instruction Hours / week: L: 0 T: 0 P: 4 **Marks:** Internal: **40** External: **60** Total: **100**
End Semester Exam**:** 3 Hours

Course Objectives

This course enables the students

- To develop skills for quantitative estimation using computer language.
- To code various differentiation and integration methods in a modern computer language.
- To plot the graphs of function

Course Outcomes (COs)

On successful completion of this course, the student will be able to

- Solve complicated matrix related problems like matrix inverse and matrix multiplication.
- Acquire problem-solving skills through computer programming.
  Plot various functions and parametric curves.

List of Practical

1. Matrix addition.
2. Matrix multiplication.
3. Inverse of a matrix.
4. Transpose of a matrix
5. Plotting of graphs of function $e^{ax + b}$, log(ax + b), 1/(ax + b), sin(ax + b), cos(ax + b), |ax +b| and to illustrate the effect of a and b on the graph.
6. Plotting the graphs of polynomials of degree 4 and 5 and the derivative graph.
7. Sketching parametric curves. (Eg. Trochoid, Cycloid, Epicycloid, Hypocycloid).
8. Obtaining surface of revolution of curves.

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act, 1956)

Pollachi Main Road, Eachanari (Po), Coimbatore –641 021

DEPARTMENT OF CHEMISTRY

I B.Sc Chemistry

MATHEMATICS-I PRACTICAL

NAME : _____

REG.NO : _____

CLASS : _____

ACADEMIC YEAR : _____

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act, 1956)

Pollachi Main Road, Eachanari (Po), Coimbatore –641 021

BONAFIDE CERTIFICATE

REG.NO		**SUBJECT CODE**	

This is to certified that this is a bonafide record of work done by

_____ of _____ during

the year 2019-2020 for the Practical Examination held on _____

at Karpagam Academy of Higher Education, Coimbatore-21.

Faculty In-charge **Head of the Department**

Internal Examiner **External Examiner**

CONTENTS

EX. NO: 1
DATE: 25.07.19

MATRIX ADDITION

Question:

Add any two matrices of order three using Scilab.

Aim:

To add the two matrices of order three using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Define the matrices A and B of order three.

Step 3: Add the matrices A and B and store the value in matrix C.

Step 4: Save and execute the programme.

Step 5: Run the programme and view the output in Scilab console.

Step 6: Stop the programme.

Coding:

```
// Matrix Addition
clc;
A = [ 1 2 3; 4 5 6; 7 8 9];
B = [ 0 4 5; 7 6 8; 1 2 3];
C = A+B
```

Output:

```
-->C
C =
 1.    6.    8.
 11.   11.   14.
 8.    10.   12.
```

Result:

Thus the programme has been executed successfully and the output has been verified.

EX. NO: 2
DATE: 01.08.19

MATRIX MULTIPLICATION

Question:

Multiply any two matrices of order three using Scilab.

Aim:

To multiply the two matrices of order three using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Define the matrices A and B of order three.

Step 3: Multiply the matrices A and B and store the value in matrix C.

Step 4: Save and execute the programme.

Step 5: Run the programme and view the output in Scilab console.

Step 6: Stop the programme.

Coding:

```
// Matrix Multiplication
clc
A = [ 1 2 3; 4 5 6; 7 8 9];
B = [ 0 4 5; 7 6 8; 1 2 3];
C = A *B
```

Output:

```
-->C
C =
 17.   22.   30.
 41.   58.   78.
 65.   94.   126.
```

Result:

Thus the programme has been executed successfully and the output has been verified.

EX. NO: 3
DATE: 08.08.19

TRANSPOSE OF A MATRIX

Question:

Transpose of any matrix of order three using Scilab.

Aim:

To transpose the matrix of order three using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Define the matrix A of order three.

Step 3: Transpose the matrix A and store the value in matrix B.

Step 4: Save and execute the programme.

Step 5: Run the programme and view the output in Scilab console.

Step 6: Stop the programme.

Coding:

```
// Transpose of a matrix
clc
A = [ 1 2 3; 4 5 6; 7 8 9 ];
B = A'
```

Output:

```
-->B
B =
 1.    4.    7.
 2.    5.    8.
 3.    6.    9.
```

Result:

Thus the programme has been executed successfully and the output has been verified.

EX. NO: 4
DATE: 22.08.19

INVERSE OF A MATRIX

Question:

Find the inverse of a matrix of order three using Scilab.

Aim:

To find the inverse of a matrix of order three using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Define the matrix A of order three.

Step 3: Inverse the matrix A and store the value in matrix B.

Step 4: Save and execute the programme.

Step 5: Run the programme and view the output in Scilab console.

Step 6: Stop the programme.

Coding:

```
// Inverse of a matrix
clc
A = [ 1 3 1; 1 1 2; 2 3 4];
B = inv(A)
```

Output:

```
-->B
B =
    2.    9.   -5.
    0.   -2.    1.
   -1.   -3.    2.
```

Result:

Thus the programme has been executed successfully and the output has been verified.

EX. NO: 5
DATE: 29.08.19

Plotting of graphs of functions e^{ax+b}, $\log(ax+b)$, $\frac{1}{ax+b}$, $\sin(ax+b)$ $\cos(ax+b)$, $|ax+b|$ and to illustrate the effect of a and b on the graph

Question:

Plot exponential, logarithmic, trigonometric functions using Scilab.

Aim:

To plot exponential, logarithmic, trigonometric functions using Scilab**.**

Algorithm:

Step 1: Start the programme.

Step 2: Give the values of a and b by real numbers.

Step 3: Define the domain of the function and store it in x.

Step 4: Store the plotting function in y.

Step 5: Plot the function y and give the title of the graph.

Step 6: Save and execute the programme.

Step 7: Run the programme and view the output in graphic window.

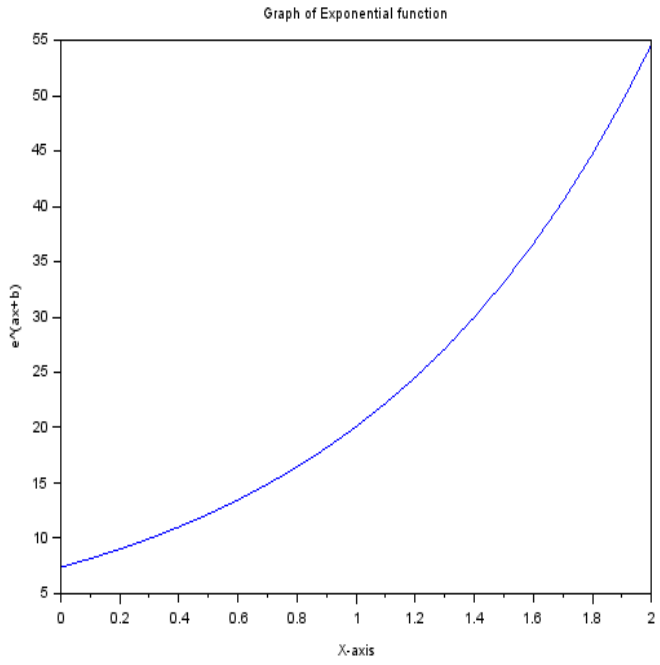
Step 8: Stop the programme.

Coding:

(a) Exponential function

```
clf
a = 1;
b = 2;
x = [ 0 : 0.1 : 2];
y =  exp(a*x +b);
plot2d(x, y,3);
title('Graph of Exponential Function')
xlabel ("X-axis")
ylabel (" Y-axis")
```

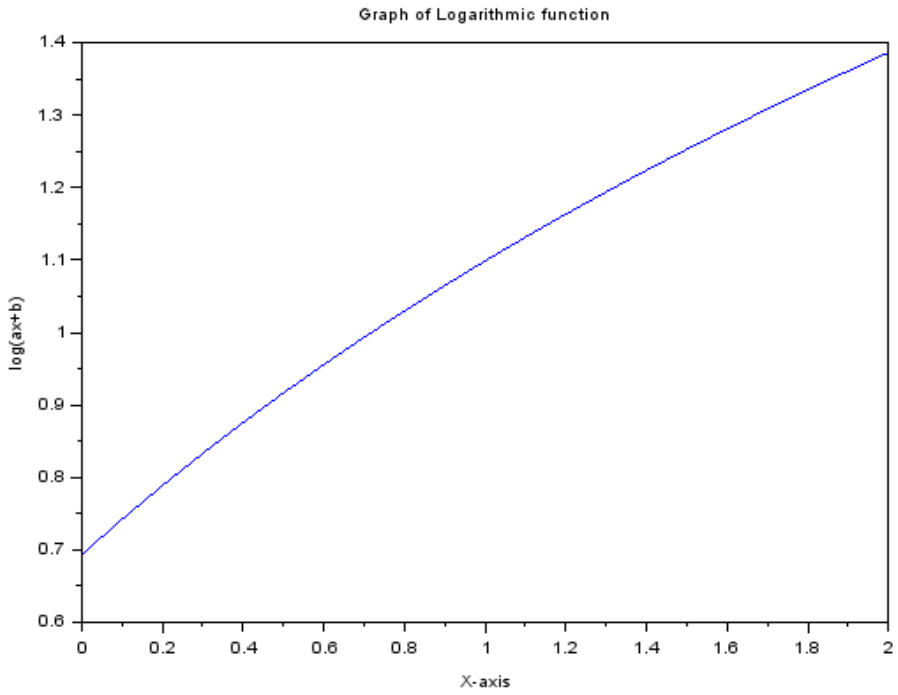
Output:



(b) Logarithmic function

```
clf
a = 1;
b = 2;
x = [ 0 : 0.1  : 2];
y =  log(a*x +b);
plot2d(x, y,3);
title('Graph of Logarithmic Function')
xlabel ("X-axis")
ylabel ("log(ax+b)" )
```

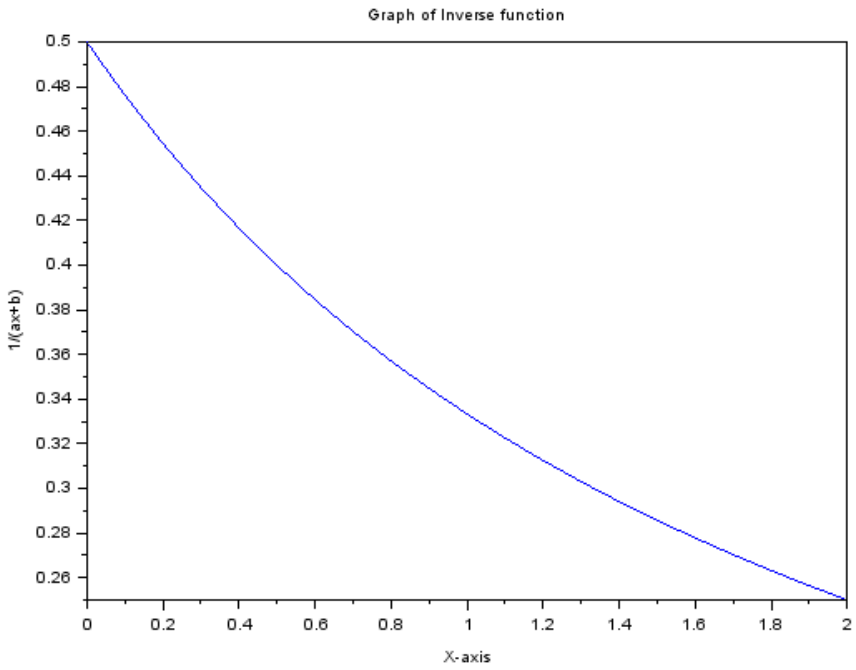
Output:



(c) Inverse function

```
clf
a = 1;
b = 2;
x = [ 0 : 0.1 : 2];
y =  (a*x+b)^-1;
plot2d(x, y,3);
title('Graph of Inverse Function')
xlabel ("X-axis")
ylabel (" 1/ ax+b " )
```

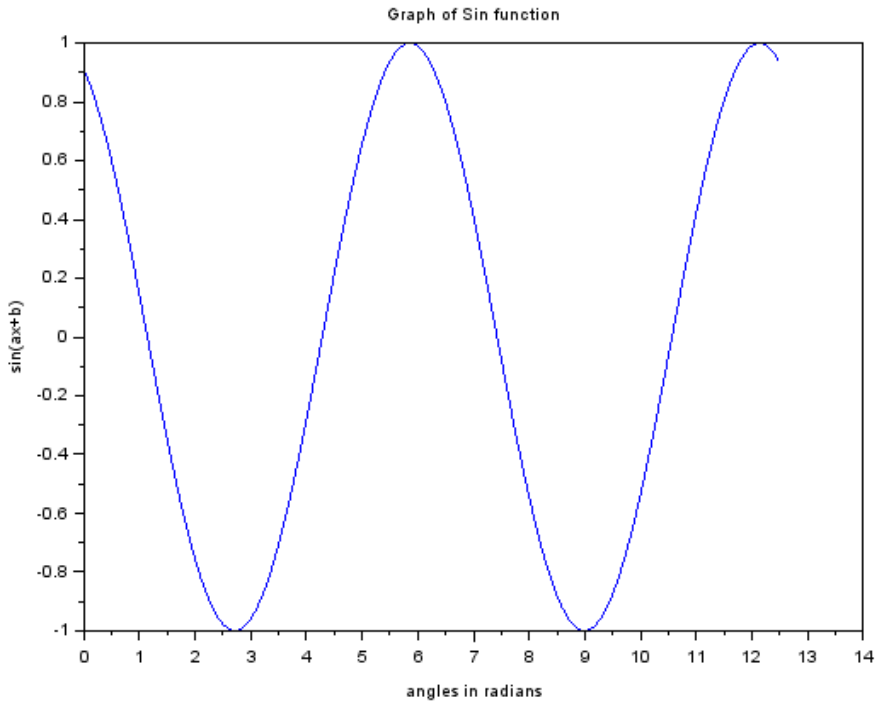
Output:



(d) Sine function

```
clf

a = 1;

b = 2;

x = [ 0 : 0.1  : 2*%pi];

y =  sin(a*x+b);

plot(x, y);

title(" Graph of Sin Function")
xlabel ("angles in radians")
ylabel (" sin(ax+b)" )
```

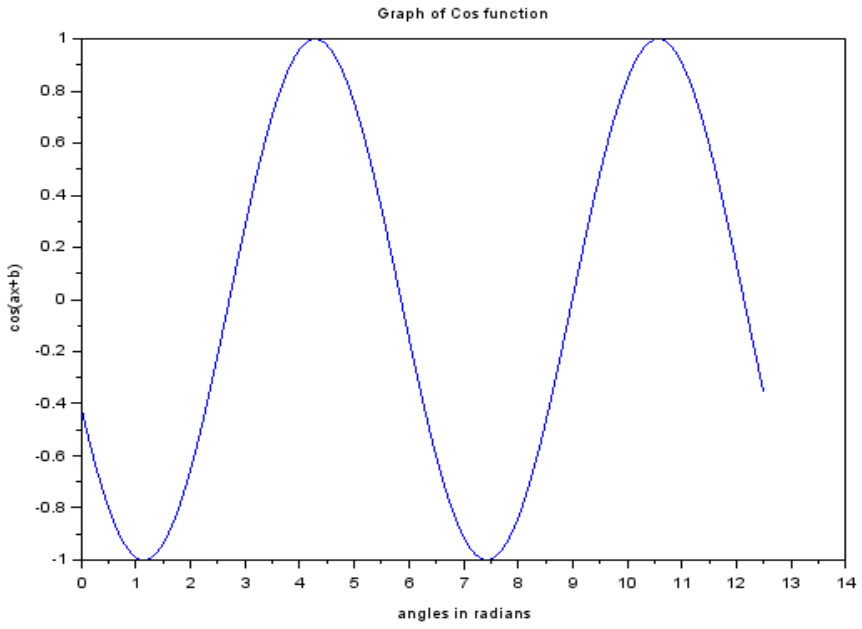
Output:



(e) Cosine function

```
clf
a = 1;
b = 2;
x = [ 0 : 0.1  : 2*%pi];
y = cos(a*x+b);
plot2d(x, y,3);
title('Graph of Cos Function')
xlabel ("angles in radians")
ylabel (" cos(ax+b)" )
```

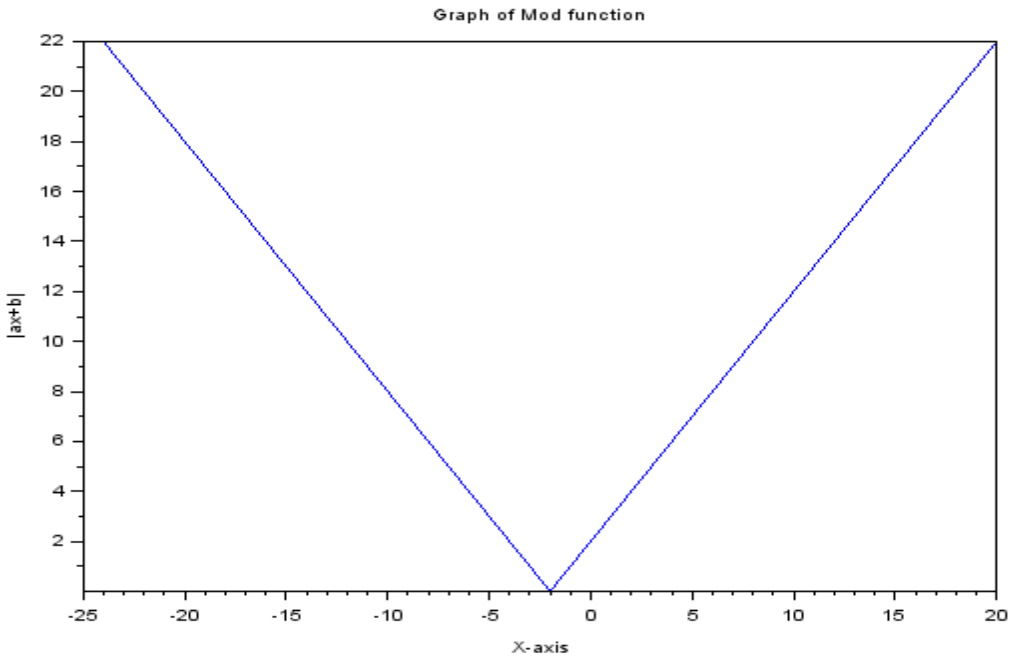
Output:



(f) Mod function

```
clf

a = 1;

b = 2;

x = [ -20 : 0.1  : 20];

y =  abs(a*x+b);

plot2d(x, y,3);

title('Graph of Mod Function')
xlabel ("X-axis")
ylabel (" |ax+b|" )
```

Output:



Result:

Thus the programme has been executed successfully and the outputs have been taken.

EX. NO: 6
DATE: 05.09.19

Plotting of graphs of polynomials of degree 4 and 5 and the derivative graph

Question:

Plot the graphs of polynomials of degree 4 and 5 and also its derivative using Scilab.

Aim:

To plot the graphs of polynomials of degree 4 and 5 and also its derivative using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Use linspace command to fix the values of x.

Step 3: Define the polynomials p1(x) and p2(x).

Step 4: Display the values of p1(x) and p2(x)

Step 4: Find derivatives of p1(x) and p2(x) and store it in d1 and d2.

Step 5: Use plot2d command to plot the graphs of polynomials and its derivatives

Step 6: Save and execute the programme.

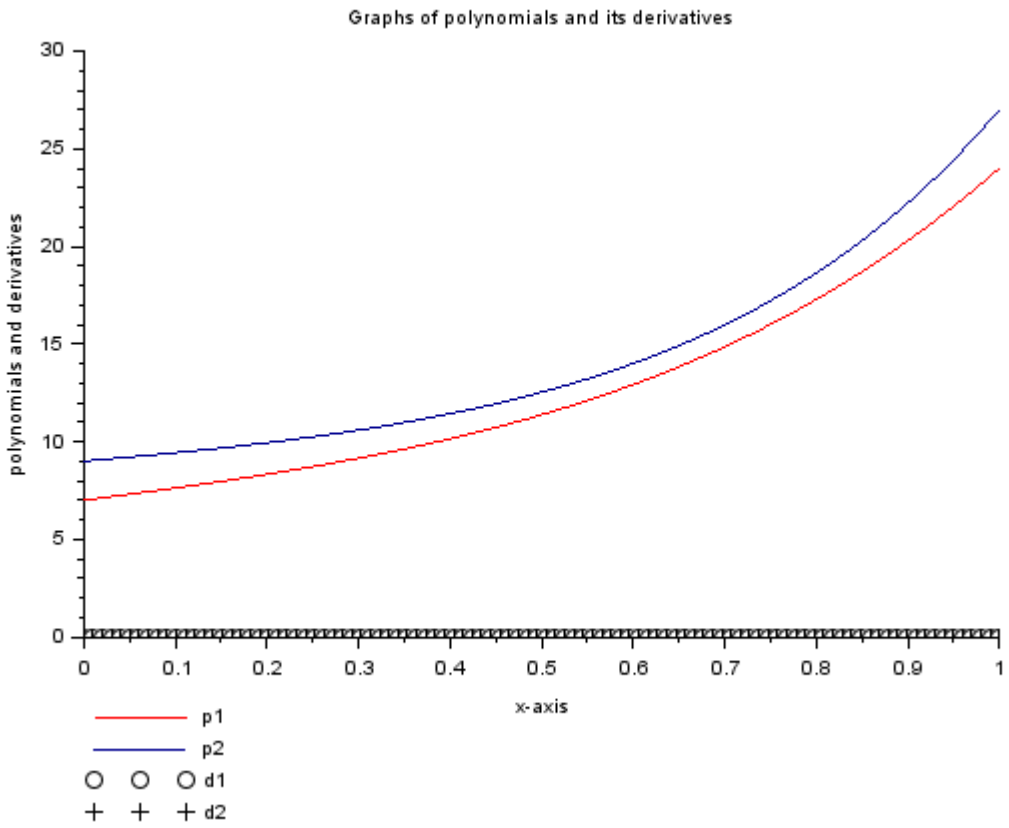
Step 7: Run the programme and view the output in graphic window.

Step 8: Stop the programme.

Coding:

```
clf
x=linspace(0, 1, 100)';
p1= 4*x^4 + 5*x^3 +2*x^2 + 6* x +7;
p2= 7*x^5 -3*x^4 +8*x^3 +2 *x^2 +4*x +9;
disp(p1);
disp(p2);
d1=derivat(p1);
d2=derivat(p2);
plot2d(x, [ p1 p2 d1 d2], [ 5  9 -9  -1], leg="p1@p2@d1@d2");
title('Graphs of polynomials and its derivatives')
xlabel(" X-axis")
ylabel(" Polynomials and derivatives")
```

Output:



Result:

Thus the programme has been executed successfully and the output has been verified.

EX. NO: 7
DATE: 12.09.19

Sketching Parametric Curves (Trochoid, Cycloid, Epicycloid, Hypocycloid)

Question:

Sketch the parametric curves (Trochoid, Cycloid, Epicycloid, Hypocycloid)

Aim:

To sketch the parametric curves (Trochoid, Cycloid, Epicycloid, Hypocycloid)

Algorithm:

Step 1: Start the programme.

Step 2: Give the values of a and b and define the

Step 3: Define the domain of the function and store it in t.

Step 4: Define the parametric curves.

Step 5: Plot the function x,y and give the title of the graph.

Step 6: Save and execute the programme.

Step 7: Run the programme and view the output in graphic window.

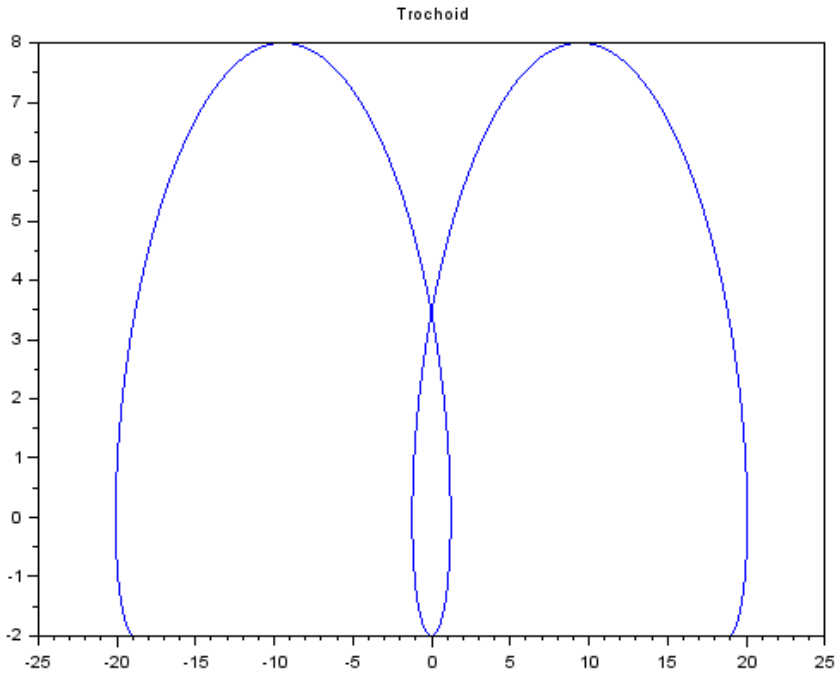
Step 8: Stop the programme.

Coding:

a) **Trochoid**

```
clf
a=3;
b=5;
t = [-2*%pi:0.01:2*%pi];
x = a*t-b*sin(t) ;
y = a-b*cos(t);
plot2d(x,y,3);
title('Trochoid')
```

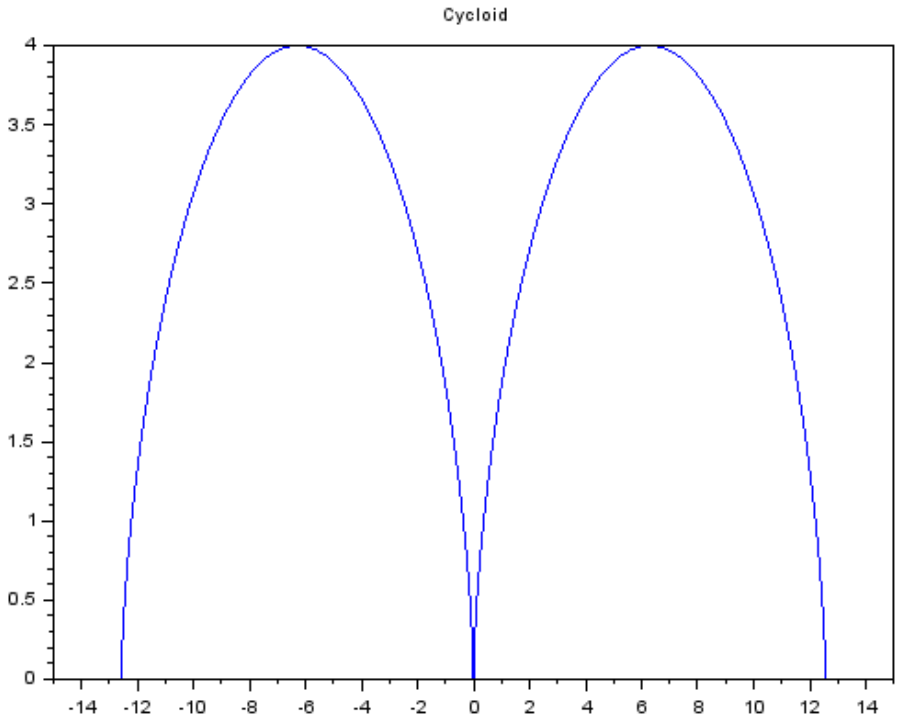
Output:



b) Cycloid

```
clf
a = 2;
t =[-2*%pi:0.01:2*%pi];
x = a*(t-sin(t));
y = a*(1-cos(t));
plot2d(x,y,3);
title('Cycloid')
```

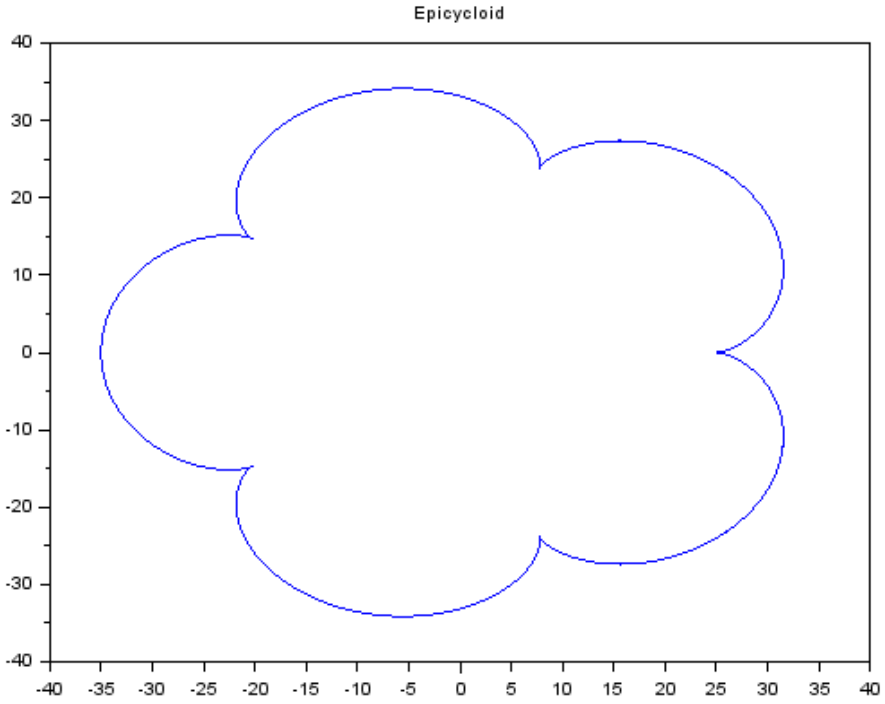
Output:



c) Epicycloid

```
clf
k=5;
r=5;
t = [-2*%pi:0.01:2*%pi];
x = r*(k+1)*cos(t)-r*cos((k+1)*t) ;
y = r*(k+1)*sin(t)-r*sin((k+1)*t) ;
plot(x,y);
title('Epicycloid')
```

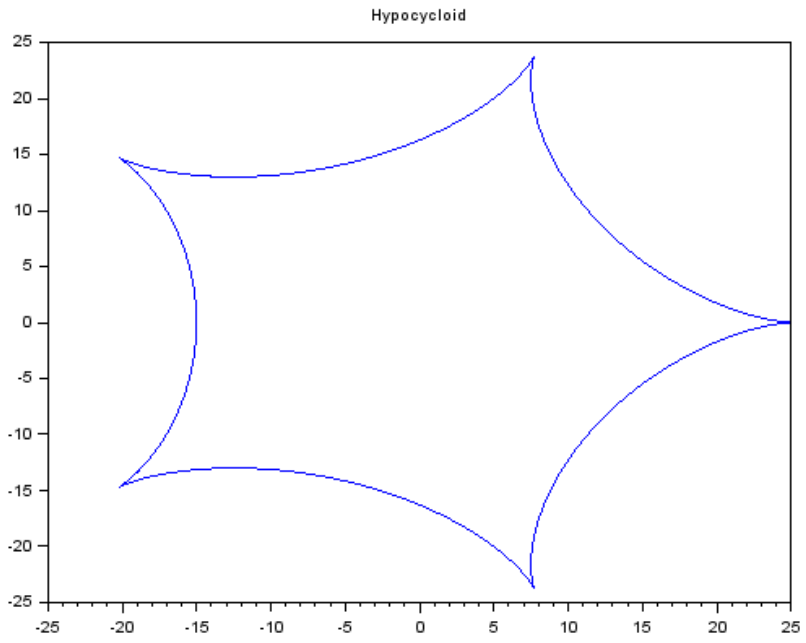
Output:



d) Hypocycloid

```
clf
k=5;
r=5;
t = [-2*%pi:0.01:2*%pi];
x = r*(k-1)*cos(t)+r*cos((k-1)*t) ;
y = r*(k-1)*sin(t)-r*sin((k-1)*t) ;
plot(x,y);
title('Hypocycloid')
```

Output:



Result:

Thus the programme has been executed successfully and the outputs have been taken.

EX. NO: 8
DATE: 26.09.19

Obtaining surface of revolution of curves

Question:

Obtain the surface of revolution of the circle using Scilab.

Aim:

To obtain the surface of revolution of the circle using Scilab.

Algorithm:

Step 1: Start the programme.

Step 2: Define the range of u and range of v.

Step 3: Define the functions x,y and z.

Step 4: Use the 3dplot command to plot the revolution of circle.

Step 5: Save and execute the programme.

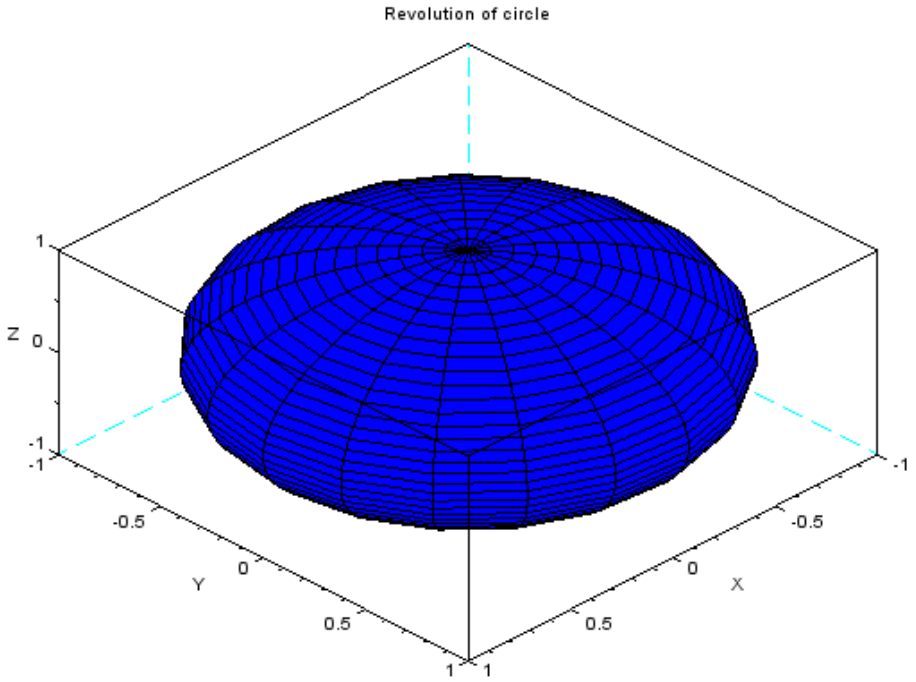
Step 6: Run the programme and view the output in graphic window.

Step 7: Stop the programme.

Coding:

```
u = linspace(-%pi/2,%pi/2,40);
v = linspace(0,2*%pi,20);
x = cos(u)'*cos(v);
y = cos(u)'*sin(v);
z = sin(u)'*ones(v);
plot3d2(x,y,z);
title('Revolution of Circle')
```

Output:



Result:

Thus the programme has been executed successfully and the output has been taken.