# MATHEMATICS III
# LAB MC203

**ANEESH PANCHAL**
**2K20/MC/021**

DEPARTMENT OF APPLIED MATHEMATICS

—

Submitted to

—

Prof. Jamkhongam Touthang

and Mr. Surya Giri

# VISION

TO BE A WORLD CLASS UNIVERSITY THROUGH EDUCATION, INNOVATION AND RESEARCH FOR THE SERVICE OF HUMANITY.

# MISSION

✦ TO ESTABLISH CENTRES OF EXCELLENCE IN EMERGING AREAS OF SCIENCE, ENGINEERING, TECHNOLOGY, MANAGEMENT AND ALLIED AREAS.

✦ TO FOSTER AN ECOSYSTEM FOR INCUBATION, PRODUCT DEVELOPMENT, TRANSFER OF TECHNOLOGY AND ENTREPRENEURSHIP.

✦ TO CREATE ENVIRONMENT OF COLLABORATION, EXPERIMENTATION, IMAGINATION AND CREATIVITY.

✦ TO DEVELOP HUMAN POTENTIAL WITH ANALYTICAL ABILITIES, ETHICS AND INTEGRITY.

✦ TO PROVIDE ENVIRONMENT FRIENDLY, REASONABLE AND SUSTAINABLE SOLUTIONS FOR LOCAL AND GLOBAL NEEDS

# DEPARTMENT OF APPLIED MATHEMATICS

## VISION

TO EMERGE AS A CENTRE OF EXCELLENCE AND EMINENCE BY IMPARTING FUTURISTIC TECHNICAL EDUCATION WITH SOLID MATHEMATICAL BACKGROUND IN KEEPING WITH GLOBAL STANDARDS, MAKING OUR STUDENTS TECHNOLOGICALLY AND MATHEMATICALLY COMPETENT AND ETHICALLY STRONG SO THAT THEY CAN READILY CONTRIBUTE TO THE RAPID ADVANCEMENT OF SOCIETY AND MANKIND

## MISSION

✦TO ACHIEVE ACADEMIC EXCELLENCE THROUGH INNOVATIVE TEACHING AND LEARNING PRACTICES.

✦ TO IMPROVE THE RESEARCH COMPETENCE TO ADDRESS SOCIAL NEEDS.

✦ TO INCULCATE A CULTURE THAT SUPPORTS AND REINFORCES ETHICAL,PROFESSIONAL BEHAVIOURS FOR A HARMONIOUS AND PROSPEROUS SOCIETY.

✦ STRIVE TO MAKE STUDENTS TO UNDERSTAND, APPRECIATE AND GAIN MATHEMATICAL SKILLS AND DEVELOP LOGIC, SO THAT THEY ARE ABLE TO CONTRIBUTE INTELLIGENTLY IN DECISION MAKING WHICH CHARACTERISES OUR SCIENTIFIC AND TECHNOLOGICAL AGE.

# PROGRAMME EDUCATIONAL

## OUTCOMES

✦ TO PREPARE GRADUATES WITH A SOLID FOUNDATION IN ENGINEERING, MATHEMATICAL SCIENCE AND TECHNOLOGY FOR A SUCCESSFUL CAREER IN MATHEMATICS AND COMPUTING / FINANCE / COMPUTER ENGINEERING FIELDS.

✦ TO PREPARE GRADUATES TO BECOME EFFECTIVE COLLABORATORS / INNOVATORS, WHO COULD ABLY ADDRESS TOMORROW'S SOCIAL, TECHNICAL AND ENGINEERING CHALLENGES.

✦ TO ENRICH GRADUATES WITH INTEGRITY AND ETHICAL VALUES SO THAT THEY BECOME RESPONSIBLE ENGINEERS.

# PROGRAMME OUTCOMES

The POs are defined in line with the graduate attributes set by NBA.

✦ ENGINEERING KNOWLEDGE: THE GRADUATE OF MATHEMATICS & COMPUTING MUST HAVE AN ABILITY TO APPLY KNOWLEDGE OF MATHEMATICS, BASIC SCIENCE AND COMPUTER SCIENCE TO SOLVE ENGINEERING AND RELATED PROBLEMS.

✦ PROBLEM ANALYSIS: AN ABILITY TO IDENTIFY, ANALYZE AND FORMULATE COMPLEX ENGINEERING PROBLEMS TO REACH LOGICAL CONCLUSION.

✦ DESIGN/DEVELOPMENT OF SOLUTION: AN ABILITY TO DESIGN AND CONDUCT EXPERIMENTS, ANALYZE AND INTERPRET THE DATA.

✦ CONDUCT INVESTIGATIONS OF COMPLEX PROBLEMS: AN ABILITY TO USE RESEARCH BASED KNOWLEDGE AND APPLY RESEARCH METHODS TO PROVIDE VALID CONCLUSION.

✦ MODERN TOOL USAGES: AN ABILITY TO CREATE, SELECT AND IMPLEMENT APPROPRIATE TECHNIQUES, SUCH AS ARTIFICIAL INTELLIGENCE, NEURAL NETWORK TO MODEL COMPLEX COMPUTER ENGINEERING ACTIVITY.

✦ THE ENGINEER AND SOCIETY: AN ABILITY TO EXPLORE THE IMPACT OF ENGINEERING SOLUTIONS ON THE SOCIETY AND ALSO ON CONTEMPORARY ISSUES ON SOCIETAL AND ENVIRONMENTAL CONTEXT.

✦ ENVIRONMENT AND SUSTAINABILITY: AN ABILITY TO DESIGN A FEASIBLE SYSTEM, COMPONENT OR PROCESS WITHOUT VIOLATING NORMS FOR PUBLIC HEALTH AND SAFETY, CULTURAL, SOCIAL AND ENVIRONMENTAL ISSUES.

✦ ETHICS: AN ABILITY TO UNDERSTAND AND PRACTICE PROFESSIONAL AND ETHICAL RESPONSIBILITIES. 9. INDIVIDUAL AND TEAM WORKS : AN ABILITY TO FUNCTION EFFECTIVELY AS AN INTEGRAL MEMBER OR A LEADER IN A MULTIDISCIPLINARY TEAM.

✦ COMMUNICATION: AN ABILITY TO COMMUNICATE EFFECTIVELY IN BOTH ORAL AND WRITTEN FORM FOR EFFECTIVE TECHNICAL DECISION MAKING, REPORT MAKING AND PRESENTATION.

✦ PROJECT MANAGEMENT AND FINANCE: AN ABILITY TO DEMONSTRATE PRINCIPLE OF MANAGEMENT AND APPLY THEM TO SUITABLE PROJECTS.

✦ LIFELONG LEARNING: AN ABILITY TO RECOGNIZE THE NEED FOR AND TO READY FOR LIFE LONG LEARNING TO KEEP UPDATED ON TECHNOLOGICAL CHANGES.

# ACKNOWLEDGEMENT

We would sincerely like to thank Prof. Jamkhongam Touthang and Mr. Surya Giri who encouraged me in carrying out the experiments and has a guiding spirit behind completion of this course.

I am very thankful to him for putting tremendous efforts from his side to assist me as much as possible.

## CONTENTS

| | |
|---|---|
| 07. | Sketch the inside of the circle r = a and outside of the cardioid x = r(1 - cos(t)) and find the area enclosed. |
| 08. | Plot the surface defined by the function F(x,y) = xy exp[-2 ( x^2 + y^2 )] on the domain -2 ≤ x ≤ 2 and -2 ≤ y ≤ 2. Find the values and locations of the maxima and minima of the function. |
| 09. | Solve by variation of parameters<br>y'' + 4 y = sec(x). |
| 10. | Determine the characteristic polynomial of a matrix by evaluating the polynomial P(λ) at n+1 points. Then fit the polynomial to these points. |

# EXPERIMENT - 1

Write a program to show the consistency and inconsistency of the system of linear equations. If the system is consistent then solve the given system of equations for unique/infinite solutions (with degree of freedom).

## Code:

```
A = [7 5 -3; 3 -5 2; 5 3 -7];
B = [16; -8; 0];
s = size(A);
ra = rank(A);
rab = rank([A B]);
if ra==s & rab==s
   disp('Unique Solution')
   x = inv(A)*B
   %Degree of freedom, dof
   dof = s-ra
elseif ra~=rab
   disp('No Solution')
else
   disp('Infinite Solution')
end
```

## Results:

```
>> Experiment_1

Unique Solution

x =

     1
     3
     2


dof =

     0     0

>>
```

# EXPERIMENT - 2

Graphically compare sin(x) and its Taylor's series expansion (up to degree 10) in the neighborhood of 1.

## Code:

```
%Variables
syms x;

%Plotting Taylor series expansion upto degree 10
taylor_exp = taylor(sin(x),x,1,'order',11)
taylor_plot = ezplot(x,t);
set(taylor_plot,'color','r');
grid;
hold on;

%Plotting sine function
y = sin(x);
sine_plot = ezplot(x,y)
set(sine_plot,'color','b')
hold off;

%Labeling and giving titles
title('Taylor series');
legend('taylor series','sin(x)');
xlabel('x-axis');
ylabel('y-axis');
```

## Results:

```
>> Experiment_2

taylor_exp =
sin(1) - (sin(1)*(x - 1)^2)/2 + (sin(1)*(x - 1)^4)/24 - (sin(1)*(x - 1)^6)/720 + (sin
(1)*(x - 1)^8)/40320 - (sin(1)*(x - 1)^10)/3628800 + cos(1)*(x - 1) - (cos(1)*(x - 1)
^3)/6 + (cos(1)*(x - 1)^5)/120 - (cos(1)*(x - 1)^7)/5040 + (cos(1)*(x - 1)^9)/362880

taylor_plot =

  Line with properties:

        Color: [0 0.4470 0.7410]
        LineStyle: '-' LineWidth: 0.5000
        Marker: 'none' MarkerSize: 6
```

MarkerFaceColor: 'none'

XData: [1×300 double]
YData: [1×300 double]
ZData: [1×0 double]

Show all properties

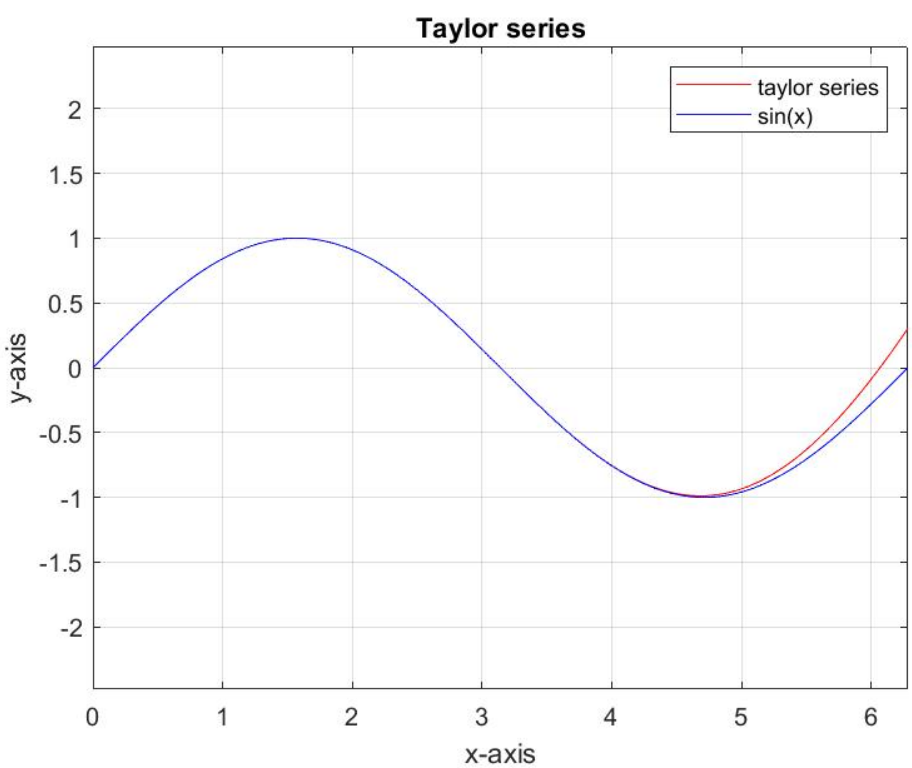
sine_plot =

Line with properties:

Color: [0.8500 0.3250 0.0980]
LineStyle: '-' LineWidth: 0.5000
Marker: 'none' MarkerSize: 6
MarkerFaceColor: 'none'
XData: [1×300 double]
YData: [1×300 double]
ZData: [1×0 double]

Show all properties

>>



## EXPERIMENT - 3

Given two polynomials f= 15 x^3 - 7 x^2 + 2 x + 4 and g= 9 x^3 - 17 x + 3. Do the following

- Find the product of f and g
- Find the quotient and remainder of f divided g
- Find the roots of f and g
- Find the value of f at x=3 and g at x=2i

## Code:

```
%Coefficient matrix of polynomial function f and g
f = [15 -7 2 4];
g = [9 -17 3];

%Product of f and g
prod_const = conv(f,g)

%Roots of f and g
root_f = roots(f)
root_g = roots(g)

%Quotient and Remainder when f is divided by g
[Q,R] = deconv(f,g);
Quotient = Q
Remainder = R

%Value of f at 3 and g at 2i
value_f = polyval(f,3)
value_g = polyval(g,2i)
```

## Results:

```
>> Experiment_3

prod_const =

  135  -318  182  -19  -62  12


 root_f =

  0.4672 + 0.5933i
  0.4672 - 0.5933i
```

```
  -0.4676 + 0.0000i


root_g =

  1.6919
  0.1970


Quotient =

  1.6667  2.3704


Remainder =

  0  0  37.2963  -3.1111


value_f =

  352


value_g =

  -33.0000  -34.0000i


>>
```

# EXPERIMENT - 4

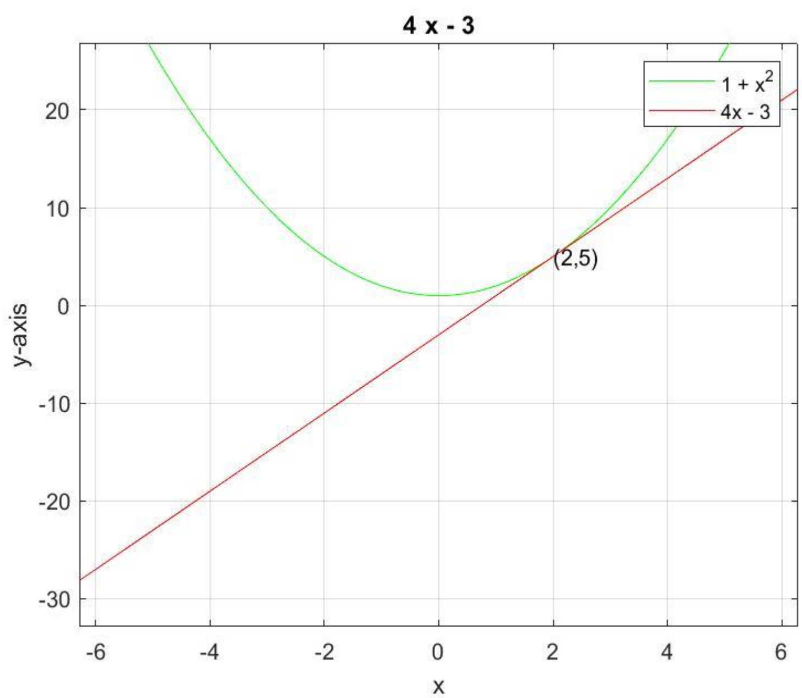
To draw a tangent line at point on a given curve y = 1+ x^2 at the point (2, 5) and also find the radius of curvature at that point.

## Code:

```
syms x
y1 = 1 + x.^2;
p = ezplot(y1);
set(p,'color','g');
grid;
yd1 = diff(y1,x);
s1 = subs(yd1,x,2);
y2 = s1*(x-2)+5;
hold on;
title('Tangent to curve and Radius of curvature')
xlabel('x-axis');
ylabel('y-axis');
q = ezplot(y2);
set(q,'color','r');
text(2,5,'(2,5)');
legend('1 + x^2','4x - 3');
hold off;
yd2 = diff(yd1,x);
s2 = subs(yd2,x,2);
roc = ((1+(s1^2))^1.5)/s2
```

## Results:

```
>> Experiment_4

roc =

(17*17^(1/2))/2

>>
```



# EXPERIMENT - 5

Write a program to determine the largest two eigenvalues of a matrix.

## Code:

```
A = [1 2 3;4 5 6;7 8 9];

%This gives 2 max eigen values of A
[V,D] = eigs(A,2,'LM');

%D will contain 2 maximum eigen values
D
```

## Results:

```
>> Experiment_5

D =

   16.1168         0
         0   -1.1168

>>
```

# EXPERIMENT - 6

Using the inbuilt ode solver ode23 and ode45 find y(3), where y is the solution of the following initial value problem and hence compare the value with the actual answer.

y' = y/x, y(0) = 1.

## Code I:

**diffy.m**

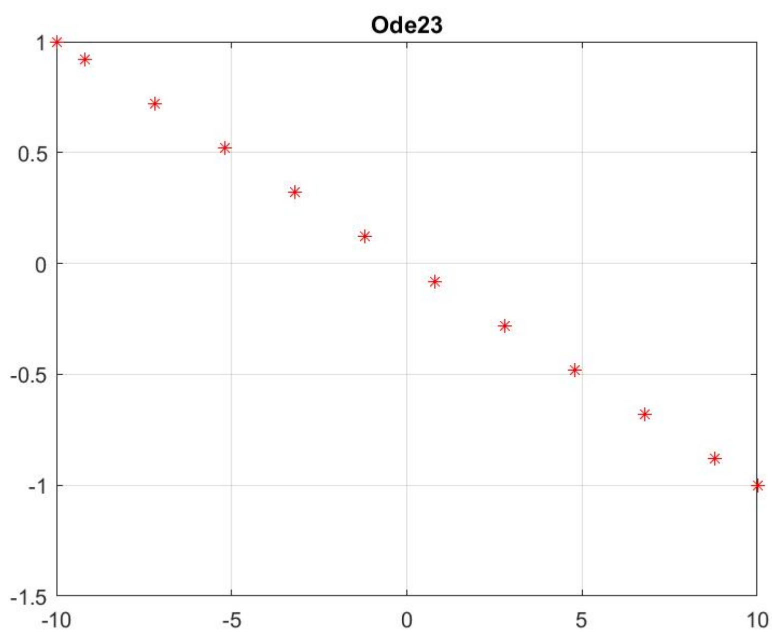
```
function [ydash] = diffy(x,y)
   ydash = y/x;
end
```

**Experiment_6**

```
yspan = [-10 10];
y0 = 1;
[x,y] = ode23('diffy',yspan,y0);
plot(x,y,'r*')
ysol = y(3)
grid;
title('Ode23');
```

## Results:

```
>> Experiment_6

ysol =

   0.7200

>>
```



# Code II:

**diffy.m**

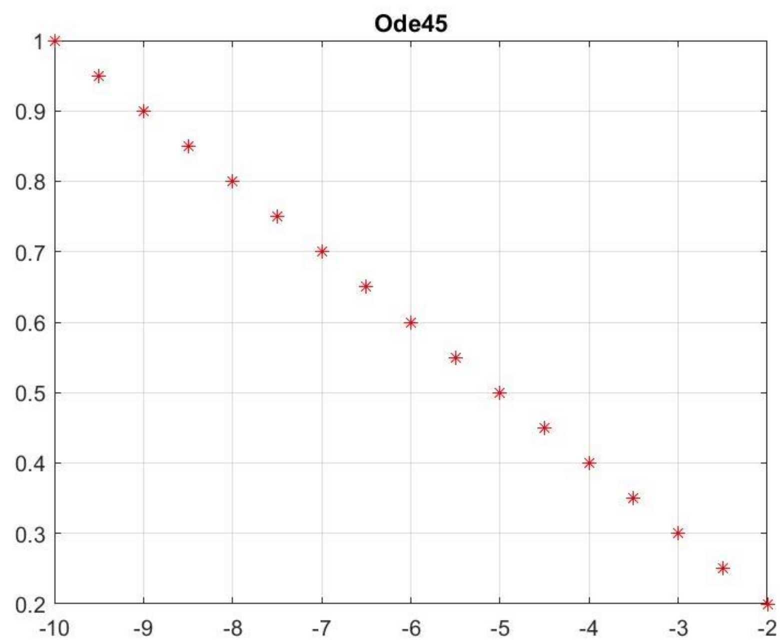
```
function [ydash] = diffy(x,y)
   ydash = y/x;
end
```

**Experiment_6_2**

```
yspan = [-10 10];
y0 = 1;
[x,y] = ode45('diffy',yspan,y0);
plot(x,y,'r*')
ysol = y(3)
grid;
title('Ode45');
```

# Results:

```
>> Experiment_6_2

ysol =

   0.9000

>>
```



## EXPERIMENT - 7

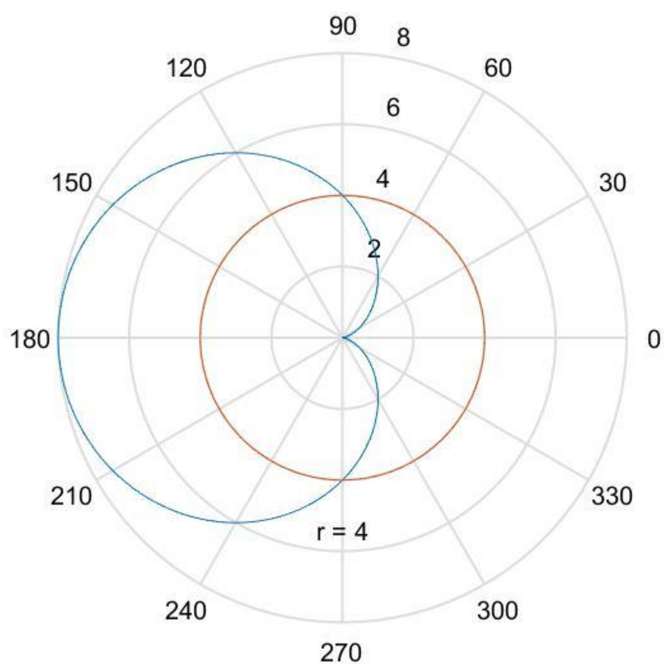
Sketch the inside of the circle r = a and outside of the cardioid x = r(1 - cos(t)) and find the area enclosed.

## Code:

```
%Circle: r = a and Cardiod r = a(1-cos(t))
%Assume a = 4
syms t;
f = 4*(1-cos(t));
ezpolar('4*(1-cos(t))');
hold on;
g = 4;
ezpolar('4');
hold off;

syms ar;
area = abs(int(f,t,-pi/2,pi/2)-int(g,t,-pi/2,pi/2))
```

## Results:

```
>> Experiment_7

area =

    8

>>
```




r = 4 (1-cos(t))

# EXPERIMENT - 8

## Plot the surface defined by the function F(x,y) = xy exp[-2 ( x^2 + y^2 )] on the domain -2 ≤ x ≤ 2 and -2 ≤ y ≤ 2. Find the values and locations of the maxima and minima of the function.

## Code:

```
syms x y;
[x,y] = meshgrid(-2:0.1:2,-2:0.1:2);
f = x.*y.*exp(-2*(x.^2 + y.^2));
xlabel('X-Axis');
ylabel('Y-Axis');
grid;
figure(1);
mesh(x,y,f)
figure(2);
xlabel('X-Axis');
ylabel('Y-Axis');
grid;
contour(x,y,f)
hold on

maxima = max(max(f))
lmaxima = find(f == max(max(f)));
pos1 = [x(lmaxima) y(lmaxima)];
plot(x(lmaxima),y(lmaxima),'*')
text(x(lmaxima),y(lmaxima),'Maxima')

minima = min(min(f))
lminima = find(f == min(min(f)));
pos2 = [x(lminima) y(lminima)];
plot(x(lminima),y(lminima),'*')
text(x(lminima),y(lminima),'Minima')
hold off
```

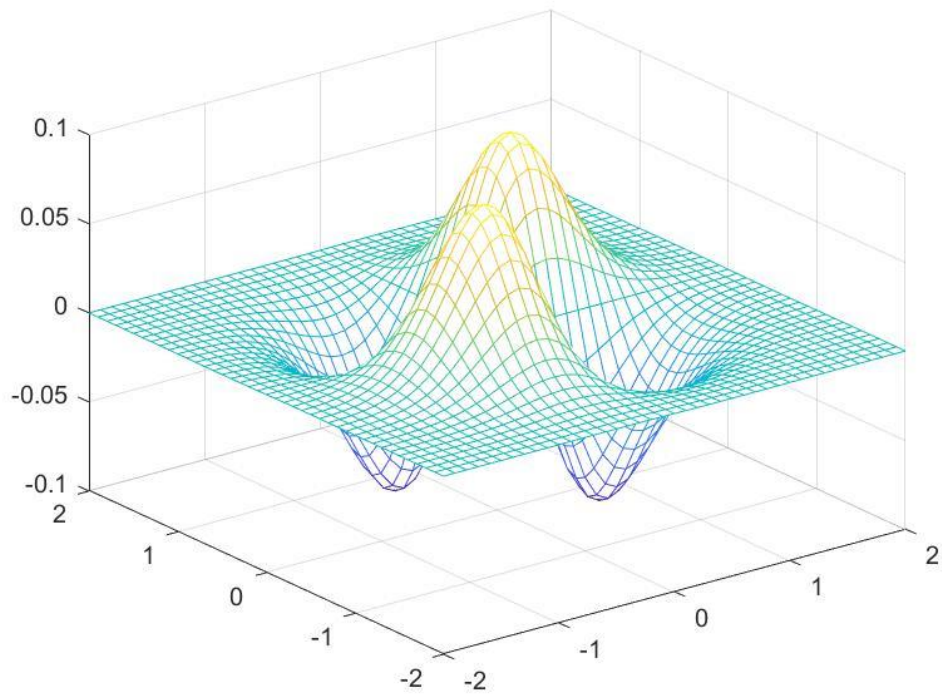
## Results:

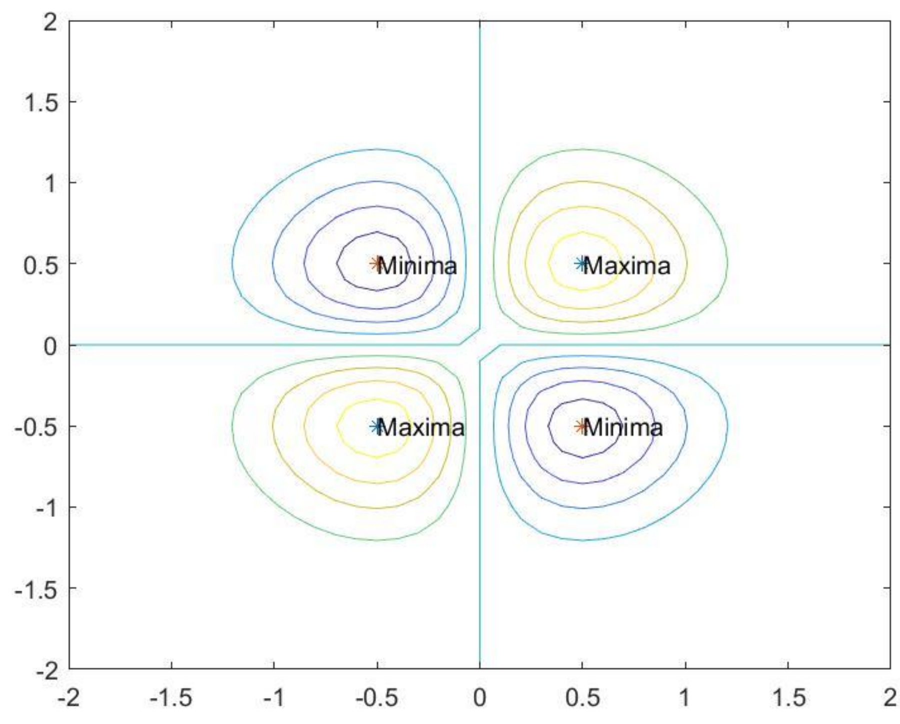
```
>> Experiment_8

maxima =

   0.0920
```

minima =

-0.0920

>>



# EXPERIMENT - 9

Solve by variation of parameters y'' + 4y = sec(x).

## Code:

```
syms x y(x)
yc=dsolve(diff(y,x,2)+4*y==0)
y1=cos(2*x);
y2=sin(2*x);
dy1=diff(y1,x);
dy2=diff(y2,x);
%Solving for Wronskian
w=[cos(2*x) sin(2*x); dy1 dy2]
r=det(w)
w1=[0 sin(2*x);sec(x) dy2]
r1=det(w1)
w2=[cos(2*x) 0; dy1 sec(x)]
r2=det(w2)
%Using Wronskian for the solution
u=int(r1/r)
v=int(r2/r)
yp=u*cos(2*x)+v*sin(2*x)
y=yc+yp
```

## Results:

```
>> Experiment_9

yc =

C1*cos(2*x) - C2*sin(2*x)


w =

[    cos(2*x),   sin(2*x)]
[ -2*sin(2*x), 2*cos(2*x)]


r =

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

```
w1 =

[        0,   sin(2*x)]
[ 1/cos(x), 2*cos(2*x)]


r1 =

-sin(2*x)/cos(x)


w2 =

[    cos(2*x),        0]
[ -2*sin(2*x), 1/cos(x)]


r2 =

cos(2*x)/cos(x)


u =

cos(x)


v =

sin(x) - atanh(sin(x))/2


yp =

cos(2*x)*cos(x) - sin(2*x)*(atanh(sin(x))/2 - sin(x))


y =

cos(2*x)*cos(x) + C1*cos(2*x) - C2*sin(2*x) - sin(2*x)*(atanh(sin(x))/2 - sin(x))

>>
```

# EXPERIMENT - 10

Determine the characteristic polynomial of a matrix by evaluating the polynomial P(λ) at n+1 points. Then fit the polynomial to these points.

## Code:

**diffy.m**

```
function [co]=charpoly(A)
   [m n]= size(A)
   if(m~=n)
      disp('matrix is not square');
      co=[ ];
      Return;
   end
   for i=1:(n+1)
      x(i)=(i-1)*pi/n;
      y(i)=det(A-x(i)*eye(n));
   end
   co=polyfit(x,y,n)
end
```

**Experiment_10**

```
A=[7 8 6 7; 1 4 6 8 ; 1 2 3 8; 3 6 8 9]
charpoly(A)
z=length(ans);
syms x
f=0;
i=4;
for y=1:1:z
f=f+ans(y).*x.^(i-1) ;
i=i+1;
end
```

## Results:

```
>> Experiment_10

A =

    7    8    6    7
    1    4    6    8
    1    2    3    8
    3    6    8    9
```

```
m =

     4


n =

     4


co =

    1.0000  -23.0000   28.0000  154.0000  -82.0000


ans =

    1.0000  -23.0000   28.0000  154.0000  -82.0000

>>
```