MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 1:

Aim:- Program to solve algebraic and transcendental equation by bisection method.

```
clc;

clear;

close;

deff('y=f(x)','y=x^3-x-1');

x1=1, x2=2; // f(1) is negative & f(2) is positive

d=0.0001; //for accuracy of root

c=1;

printf('Succcessive approximatiions  x1 \t x2  \t  f(m) ');

while abs(x1-x2)>d

   m=(x1+x2)/2;

printf('\n\t %f \t %f \t %f \t %f \n',x1,x2,m,f(m));

 if f(m)*f(x1)>0

    x1=m;

 else

    x2=m;

end

c=c+1; // to cound the number of iterations

end

printf('\n the solution of equation after %i iteration is %g',c,m);
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 2:

Aim:- Program to solve differential equation using Euler's method.

```
clc;
clear;
close;
deff('z=f(y)','z=-y')
y(1)=1;//value at 0
h=0.01;c=0.01;
for i=1:4
   y(i+1)=y(i)+h*f(y(i))
   printf('\n y(%g)=%g\n',c,y(i+1));
   c=c+0.01;
end
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 3:

Aim:- Program to solve algebraic and transcendental equation by false position method.

```
clc;
clear;
close
deff('y=f(x)','y=x^3-2*x-5');
a=2, b=3;//f(2) is negative & f(3) is positive)
d=0.00001;
printf('successive iteration\n \t  a  \t  b  \t  f(a)  \t  f(b)  \t   x1  \n');
for i=1:25
   x1=b*f(a)/(f(a)-f(b))+a*f(b)/(f(b)-f(a));
   if(f(a)*f(x1))>0
      b=x1;
   else
      a=x1;
   end
   if abs(f(x1))<d
      break
   end
  printf('\n%f\t%f\t%f\t%f\t%f\n',a,b,f(a), f(b),x1);
end
printf('the root of the equation is %f', x1);
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 4:

Aim:- Program to solve linear system of equation using Gauss Jordan method.

```
clc;
clear;
close;
A=[2,1,1,10;3,2,3,18;1,4,9,16]; //augmented matrixf
for i=1:3
   j=i
   while(A(i,i)==0&j<=3)
  for k=1:4
          B(1,k)=A(j+1,k)
      A(j+1,k)=A(i,k)
         A(i,k)=B(1,k)
end
disp(A);
j=j+1;
end
disp(A);
for k=4:-1:i
   A(i,k)=A(i,k)/A(i,i)
end
disp(A)
for k=1:3
  if(k~=i) then
     l=A(k,i)/A(i,i)
     for m=i:4
        A(k,m)=A(k,m)-l*A(i,m)
```

```
 end
end
end
disp(A)
end
for i=1:3
   printf('\n x(%i)=%g\n', i,A(i,4))
end
```

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 5:

Aim:- Program for Lagrange interpolation.

```
clc;
clear;
close;
x=[300 304 305 307];
y=[2.4771 2.4829 2.4843 2.4871];
x0=301;
log_301=0;
poly(0,'x');
for i=1:4
   p=y(i);
  for j=1:4
     if i~=j
        then
        p=p*((x0-x(j))/(x(i)-x(j)))
     end
  end
  log_301=log_301+p;
end
disp(log_301,'log_301=');
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 6:

Aim:- Program to solve algebraic and transcendental equation by Newton Raphson's method.

```
clc;
clear;
close
deff('y=f(x)','y=x^3-2*x-5');
deff('y1=f1(x)','y1=3*x^2-2');//first derivative
x0=2;
d=0.0001;
c=0;n=1;
printf('successive iterations \n\t x0 \t\t f(x0) \t\t f1(x0)\n');
while n==1
   x2=x0;
   x1=x0-(f(x0)/f1(x0));
   x0=x1;
   printf('\t %f \t %f \t %f \n',x2,f(x1),f1(x1))
   c=c+1;
   if abs(f(x0))<d then
      break;
   end
end
printf('the root of %i iteration is %f', c,x0);
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 7:

Aim:- Program to solve differential equation using Runge Kutta 2nd order and 4th order method.

```
clc;

clear;

close;

deff('y=f(x,y)','y=y-x')

y=2;

x=0;

h=0.1;

k1=h*f(x,y);

k2=h*f(x+h,y+k1);

y1=y+(k1+k2)/2

printf('\n y(0.1) by second order runge kutta method: %0.4f',y1);

y=y1;

x=0.1;

h=0.1;

k1=h*f(x,y);

k2=h*f(x+h,y+k1);

y1=y+(k1+k2)/2

printf('\n y(0.2) by second order runge kutta method %0.4f',y1);

y=2,x=0,h=0.1;

k1=h*f(x,y)

k2=h*f(x+h/2, y+k1/2);

k3=h*f(x+h/2,y+k2/2);

k4=h*f(x+h,y+k3);

y1=y+(k1+2*k2+2*k3+k4)/6;

printf('\n y(0.1) by fourth order runge kutta method: %0.4f',y1);
```

Prof Pooja Jha

```
y=y1, x=0.1, h=0.1;

k1=h*f(x,y);

k2=h*f(x+h/2, y+k1/2);

k3=h*f(x+h/2,y+k2/2);

k4=h*f(x+h,y+k3);

y1=y+(k1+2*k2+2*k3+k4)/6;

printf('\n y(0.1) by fourth order kutta method: %0.4f',y1);

y=2,x=0,h=0.1;
```

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 8:

Aim:- Program for numerical integration using Simpson's $1/3^{rd}$ rule

```
clc;
clear;
close;
x=[0.00 0.25 0.50 0.75 1.00];
y=[1.000 0.9896 0.9589 0.9089 0.8415];
y=y^2;
h=x(2)-x(1);
l=length(x);
area =0;
for i=1:1
   if i==1|i==1 then
      area = area + y(i)
   elseif(modulo(i,2))==0 then
         area = area + 4 * y(i)
      elseif(module(i,2))~=0 then
            area = area + 2*y(i)
            end
end
area = area * (h*%pi)/3;
printf('area bounded by the curve is ....%f',area);
```

Prof Pooja Jha

MVM DEGREE COLLEGE OF COMMERCE AND SCIENCE

PRACTICAL NO 9:

Aim:- Program for numerical integration using trapezoidal rule.

```
clc;
clear;
close;
x=[7.47 7.48 7.49 7.0 7.51 7.52];
f_x=[1.93 1.95 1.98 2.01 2.03 2.06];
h=x(2)-x(1);
l=length(x);
area=0;
for i=1:1
   if i==1|i==1 then
      area = area + f_x(i)
   else
      area = area + 2*f_x(i)
      end
end
area = area * (h/2);
printf('area bounded by the curve is %f', area);
```

Prof Pooja Jha