```
% Euler's Method
clc;
clear all;
f = input('Enter the function: ');
h = input('Enter the value of h: ');
e=input('Enter the starting value of x: ');
e_2=input('Enter the value of x you want answer: ');

x = e:h:e_2;
y = zeros(size(x));
y(1) = input('Enter the starting value of y: ');
n = numel(y);

for i=1:n-1

    y(i+1) = y(i)+(h*f(x(i),y(i)));
end
fprintf('The value of y(%0.4f)=%0.4f',e_2,y(n));


clc;
clear all;
x(1)=0;
y(1)=1;
xf=0.1;
xi=x(1);
h=0.02;
n=(xf-xi)/h;
for i=1:n
    y(i+1)=y(i)+h*((y(i)-x(i))/(y(i)+x(i)));
    x(i+1)=x(i)+h;
end
disp(y);


%R-K 4th order
clear all;
clc;
f=input('enter the function: ');
h=input('enter step length: ');
x(1)=input('enter the initial value of x: ');
y(1)=input('enter the initial value of y: ');
a=input('enter the evaluating point: ');
n=(a-x(1))/h;
for i=1:n
    k1(i)=h*f(x(i),y(i));
    k2(i)=h*f((x(i)+h/2),(y(i)+k1(i)/2));
    k3(i)=h*f((x(i)+h/2),(y(i)+k2(i)/2));
    k4(i)=h*f((x(i)+h),(y(i)+k3(i)));
    y(i+1)=y(i)+(1/6)*(k1(i)+2*(k2(i))+2*(k3(i))+k4(i));
    x(i+1)=x(i)+h;
end
fprintf('The value of y(%0.4f)=%0.4f',a,y(n+1));


% Regula Falsi
```

```
clc;
clear all;
f=input('Enter the function: ');
k=input('Enter the Highest or lowest range for solution: ');
for i=-k:1:k
    a=i;
    if f(a)*f(a+i)<0
        break;
    end
end
    b=a+1;
    if f(a)<0
        t=b;
        b=a;
        a=t;
    end
for i=1:1000
       x0=a; x1=b;
       x2(i) =x0-(x1-x0) / (f(x1) -f(x0))*f(x0);
if f(x2(i))>0
       b=x2(i);
else
       a=x2(i);
end
p=x2(i);
end
fprintf('The Root is: %0.4f',p);
```

```
% Secant Method
clc;
clear all;
f=input('Enter the function: ');
x0=input('Enter initial guess: ');
x1=input('Enter final guess: ');
tol=0.0001;
for i=1:1000
       x2 =(x0*f(x1)-x1*f(x0)) / (f(x1) -f(x0));
if abs(x2-x1)<tol
    break;
end
x0=x1;
x1=x2;
end
fprintf('The Root is: %0.4f',x2);
```

```
%Gauss Jacobi
clc;
clear all;
A = input('Enter coefficient matrix A: ');
b = input('Enter Source vector B: ');
P = input('Enter inital guess vector: ');
```

```
n = input('Enter no. of itteration: ');
e = input('Enter tollerance:');
N = length(b);
X = zeros (N,1);
for j=1:n
    for i=1:N
        X(i)= (b(i)/A(i,i)) -((A(i,[1:i-1,i+1:N])*P([1:i-1,i+1:N])))/A(i,i);
    end
    if abs ( (X-P))<e
        break
    end
    P=X;
end
fprintf('The soln. is: ');
fprintf('\n\t%.4f',P);
```

```
%Gauss Seidal
clc;
clear all;
A = input('Enter coefficient matrix A(after rearranging the equations): ');
B = input('Enter Source vector B: ');
P = input('Enter inital guess vector: ');
n = input('Enter no. of itteration: ');
e = input('Enter tollerance:');
N = length(B);
X = zeros (N,1);
Y = zeros (N,1);
for j=1:n
    for i=1:N
        X(i)= (B(i)/A(i,i))-(A(i,[1:i-1,i+1:N])*P([1:i-1,i+1:N]))/A(i,i);
        P(i)= X(i);
    end
    if abs (Y-X)<e
        break
    end
    Y=X;
end
fprintf('The soln. is: ');
fprintf('\n\t%.4f',Y);
```

```
%Taylor Series
clc;
clear all;
close all;
f = input('Enter the function y'': '); %@(x,y) (x*y);
fprime=input('Enter the function y": '); %@(x,y) ((x^2)*y+y);
f2prime=input('Enter the function y"'': '); %@(x,y) ((x^3*y)+(3*x*y)+(y^2));
a = input('Enter the starting point:  ');
b = input('Enter the finding point:  ');
n = input('Enter no. of subintervals: ');
alpha = input('Enter the initial condition of y:  ');
```

```
h = (b-a)/n;
x=[a zeros(1,n)];
w=[alpha zeros(1,n)];

for i = 1:n+1
    x(i+1)=x(i)+h;
    wprime=f(x(i),w(i))+(h/2)*fprime(x(i),w(i))+(h/6)*f2prime(x(i),w(i));
    w(i+1)=w(i)+h*wprime;
end
 fprintf('y(%.2f)=%.5f\n', b, w(i));


%Modified Euler's Method
clc;
clear all;
f = input('Enter the function: ');
x1 = input('Enter initial value of x: ');
y1 = input('Enter initial value of y: ');
h = input('Enter the value of h: ');
a=input('Enter the finding point: ');
A=f(x1,y1);
y(1)=y1+h*A;
for n=1:100
y(n+1)=y1+(h/2)*(A+f((x1+h),y(n)));
    if abs(y(n+1)-y(n))<0.00001
        break;
    end
end
disp(y)
```