**CSE – 608 LABs**

**1. Introduction to basic MATLAB operations and functions.**

(Simple kisu code LAB-1 pdf theke niye korte hobe)

**2. Study of Bisection method.**

**3. Study of Regular Falsi method.**

(2,3 Previously kora ase)

**4. Study of Newton-Raphson Method**

syms x;

fx = input('Enter Function: ','s');

f= eval(['@(x)',fx]);

a = input('Enter value: ');

s=1;

f1 = diff(sym(f),x,1);

for i = 1:100

v = subs(f1,x,a);

z = a - (f(a)/v);

fprintf("%g %f %f %f %f\n",i,a,z,f(z),s);

a = z;

v = subs(f1,x,a);

xk = a - (f(a)/v);

s = abs(xk-z)/abs(xk) \*100;

if s < 0.0001

break

end

end

fprintf("Root is %f\n",a);

**5. Study of Secant Method**

clc;

clear all;

fx=input('Enter the function ,F(x) = ','s');

f=eval(['@(x)',fx]) ;

a=input('Enter a=');

b=input('Enter b=');

x(1)=a;

x(2)=b;

s=1;

fprintf('N\t\t x(i-1)\t\t x(i)\t\t x(i+1)\t\t f(x)\t\t Error\n');

for k=3:103;

it(k)=abs(k-2);

x(k)=x(k-1)-(f(x(k-1))\*(x(k-1)-x(k-2)))/(f(x(k-1))-f(x(k-2)));

c=f(x(k));

fprintf('%g %f %f %f %f %f\n\n',it(k),x(k-2),x(k-1),x(k),c,s);

s=((abs(x(k)-x(k-1)))/abs(x(k)))\*100;

if s<=.001

break;

end

end

fprintf('\n\nThe root is =%f',x(k))

**6.Study of Gauss elimination Method**

clc;

clear all;

a=input('Enter matrix A = ');

b=input('Enter matrix B = ');

[m,n]=size(a);

for k=1:m-1

for i=k+1:m

fact=a(i,k)/a(k,k);

for j=1:n

a(i,j)=a(i,j)-a(k,j)\*fact;

end

b(i,1)=b(i,1)-b(k,1)\*fact;

end

end

x(m)=b(m,1)/a(m,n);

for i=m-1:-1:1

sum=0;

for j=i+1:n

sum=sum+a(i,j)\*x(j);

end

x(i)=(b(i,1)-sum)/a(i,i);

end

disp('After forward elimination the matrix [A B] :');

disp([a b]);

fprintf('\nThe Required solution : ');

for i=1:n

fprintf('\nx(%d) = %f',i,x(i));

end

**7. Study of Trapezoidal Method**

fx = input('Enter Function: ','s');

f= eval(['@(x)',fx]);

n = input('Enter n: ');

a = input('Enter a: ');

b = input('Enter b: ');

h = (b-a)/n;

s = 0;

for i=a:h:b

fprintf("Value using x = %d is %f\n",i,f(i));

if(i==a || i==b)

s = s + f(i)/2;

else s = s + f(i);

end

end

fprintf("Integral value= %f\n",s\*h);

**8. Study of Simpson’s Rule**

clc

clear all

fx = input('Enter Function: ','s');

f= eval(['@(x)',fx]);

n = input('Enter n: ');

a = input('Enter a: ');

b = input('Enter b: ');

h = (b-a)/n;

s = 0;

j=0;

for i =a:h:b

if(i==a || i ==b)s =s+f(i);

elseif (rem(j,2)==1) s = s+ 4\*f(i);

elseif (rem(j,2)==0) s = s+ 2\*f(i);

else s= s+ f(i);

end

j = j+1;

end

s = s\*h/3

fprintf("The integral value = %f,",s);

**9. Study of Runge-Kutta Method.**

fx = input('Enter Function dy/dx= ','s');

f= eval(['@(x,y)',fx]);

x0 = input('Enter x0= ');

y0 = input('Enter y0= ');

xn = input('Enter xn= ');

n = input('Enter number of iteration= ');

h = (xn-x0)/n;

for i = x0:h:xn-h

k1 = h\*f(i, y0);

k2 = h\*f(i + (1/2)\*h, y0 + (1/2)\*k1);

k3 = h\*f(i + (1/2)\*h, y0 + (1/2)\*k2);

k4 = h\*f(i + h, y0 + k3) ;

y0 = y0 + (1/6) \*(k1 + 2\*k2 + 2\*k3 + k4);

fprintf("y(%f) = %f\n",i+h,y0);

end

Alternate code,

clc

clear all

fx=input('Enter the function dy/dx = ','s');

f=eval(['@(x,y)',fx]);

x0= input('Enter the initial value of x = x0= ' );

y0=input('Enter value of y at initial value of x = y0= ');

xn = input('Enter the value of x = xn = ' );

h = input('Enter the value of step size= h = ');

fprintf('\nThe value of y(%f) = %f\n',x0,y0);

for i = x0:h:xn-h

k1=h\*f(i,y0);

k2=h\*f(i+(h/2),y0+k1/2);

k3=h\*f(i+(h/2),y0+k2/2);

k4=h\*f(i+h,y0+k3);

k =(1/6)\*(k1 +2\*k2 + 2\*k3+ k4);

y = k+ y0;

fprintf('The value of y(%f) = %f\n',i+h,y);

end