Name : Deepankar Sharma

Course: BCA

Student Id : 20041299

University Roll No: 2092014

**Regula Falsi Method**

**Algorithm:**

1. Start

2. Define function f(x)

3. Input

a. Lower and Upper guesses x0 and x1

b. tolerable error e

4. If f(x0)\*f(x1) > 0

print "Incorrect initial guesses"

goto 3

End If

5. Do

x2 = x0 - ((x0-x1) \* f(x0))/(f(x0) - f(x1))

If f(x0)\*f(x2) < 0

x1 = x2

Else

x0 = x2

End If

While abs(f(x2) > e

6. Print root as x2

1. Stop

**Code :**

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

#define **phi**(x) (x\*x)- **log**(x)- 12

double **differential**(double x0)

{

    const double delta = 1.0e-10;

    double x1 = x0 - delta;

    double x2 = x0 + delta;

    double y1 = **phi**(x1);

    double y2 = **phi**(x2);

*// printf("gradient= %f\n", grad);*

    return (y2 - y1) / (x2 - x1);

*// return (pow(-2.718282, -1\*x)-cos(x));*

}

int **main**()

{

    int k = 0;

    double x0, x1, x2, f0, f1, f2, g0;

    int step = 1, N;

    double allErr;

**printf**("Enter the allowed Error: ");

**scanf**(" %lf", &allErr);

**printf**("Enter the interval lower limit: ");

**scanf**(" %lf", &x0);

**printf**("Enter the interval upper limit: ");

**scanf**(" %lf", &x1);

    f0 = **phi**(x0);

    f1 = **phi**(x1);

    if (f0 \* f1 > 0.0)

    {

**printf**("\n\nIncorrect Initial Guesses !!!!\n");

**exit**(0);

    }

**printf**("Enter maximum iteration:  ");

**scanf**("%d", &N);

    {

        {

**printf**("\nStep\t\tx0\t\tx1\t\tf(x0)\t\tf(x1)\t\tx2=(x0f1-x1f0)/(f1-f0)\t\tf(x2)\n");

            do

            {

                x2 = x0 - (x0 - x1) \* f0 / (f0 - f1);

                f2 = **phi**(x2);

**printf**("%d\t\t%f\t%f\t%f\t%f\t%f\t\t\t\t%f\n", step, x0, x1, f0, f1, x2, f2);

                x0=x1;

                x1=x2;

                f0=f1;

                f1=f2;

*// if (f0 \* f2 < 0)*

*// {*

*//     x1 = x2;*

*//     f1 = f2;*

*// }*

*// else*

*// {*

*//     x0 = x2;*

*//     f0 = f2;*

*// }*

                step = step + 1;

            } while (**fabs**(f2) > allErr);

**printf**("\nRoot is: %f\n", x2);

        }

    }

}

