# NEPAL ENGINEERING COLLEGE

(AFFILIATED TO POKHARA UNIVERSITY)

Changunarayan, Bhaktapur

REPORT ON:

## Root of Nonlinear Equation Using Bisection Method

SUBMITTED BY:
NAME: Subash Khanal

SUBMITTED TO:
Electrical and

CRN: 020-626 Electronics

**Experiment No: - 2**

**TITLE:-**

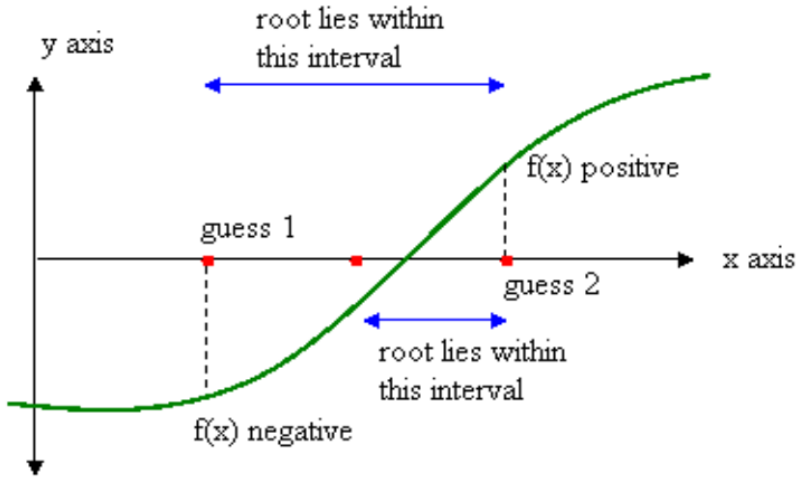
# Root of Nonlinear Equation Using Bisection Method

**OBJECTIVE:-**

To implement and calculate the root using the bisection method on Matlab and C-programming.

**THEORY:-**

The bisection method, which is alternatively called binary chopping or interval halving, is one type of incremental search method in which the interval is always divided in half. If a function changes sign over an interval, the function value at the midpoint is evaluated. The location of the root is then determined as lying at the midpoint of the subinterval within which the sign change occurs. The process is repeated to obtain refined estimates. An equation($x$) = 0, has at least one root between guess 1 (x1) and guess 2 (x2) if$f(x1) * f(x2) < 0$.



**ALGORITHM:-**

Step 1: Guesses the initial x1 and x2 which enclose the root i.e.
f(x1) * f(x2) <0.

Step 2: Calculate error =|x2-x1|>=0.001

Step 3: Calculate the x0=$\frac{x1+x2}{2}$.

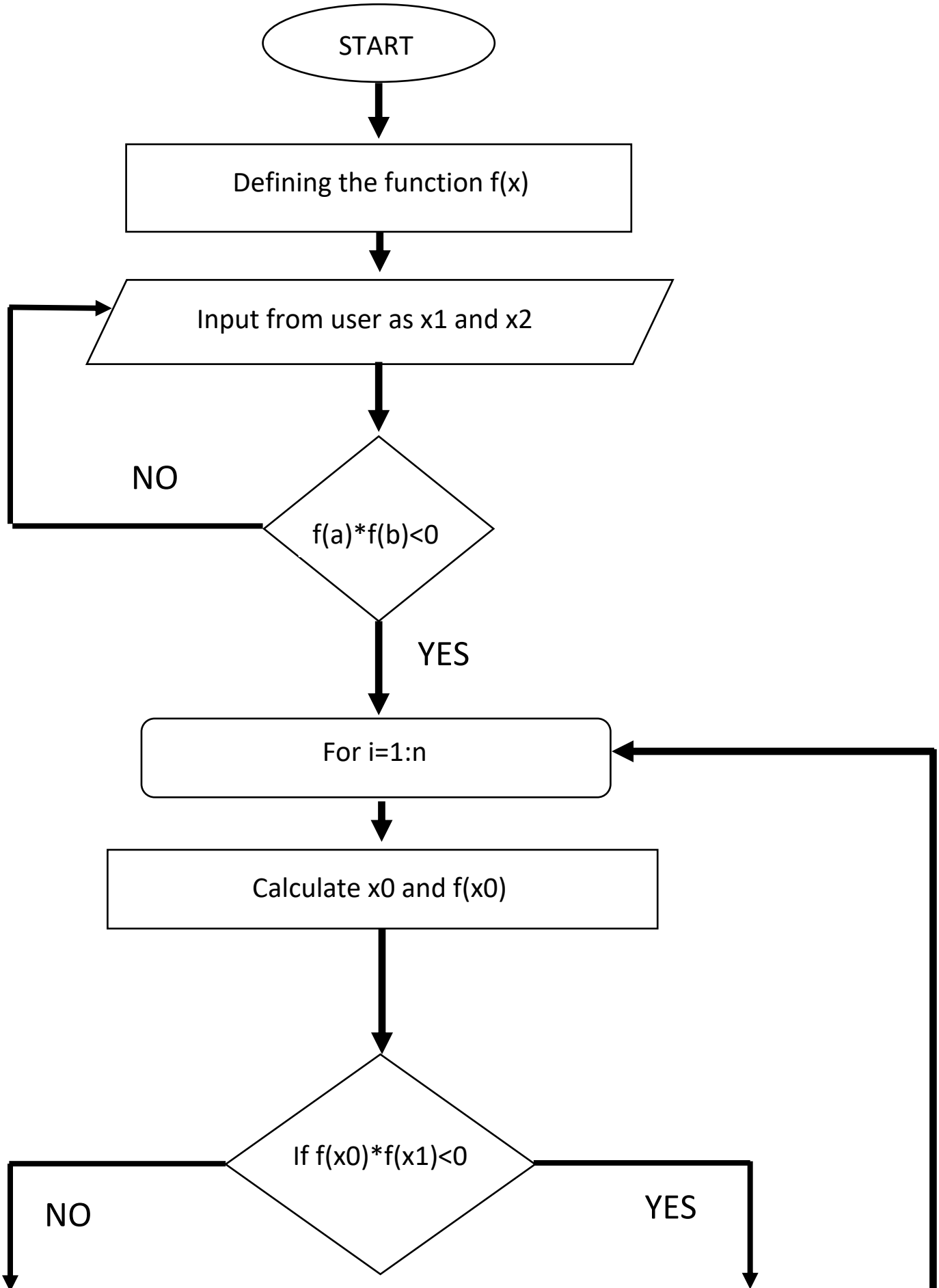
Step 4: Calculate f(x0).

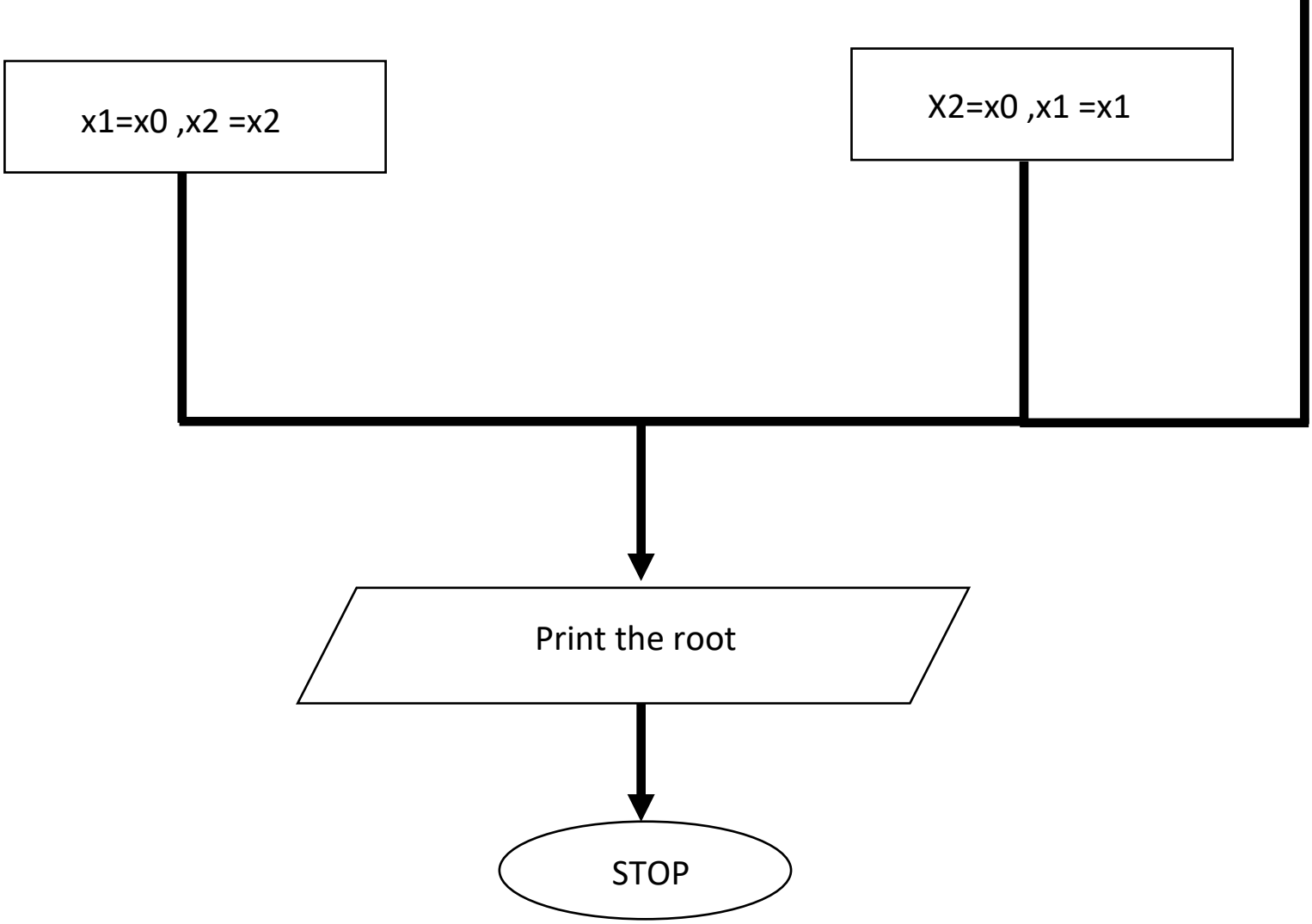
Step 5: if f(x0)*f(x1)<0 then // Condition check
Replace the value of x2 by x0.
else
Replace the value of x1 by x0.

Step 6: If absolute value of ($x$0) is less than or equal to give limit, then $root$ = $x$0 (Go to step 7)
else
Go to step 3.

Step 7: Stop the program

# FLOW-CHART


START
Defining the function f(x)
Input from user as x1 and x2
NO
f(a)*f(b)<0
YES
For i=1:n
Calculate x0 and f(x0)
If f(x0)*f(x1)<0
NO
YES

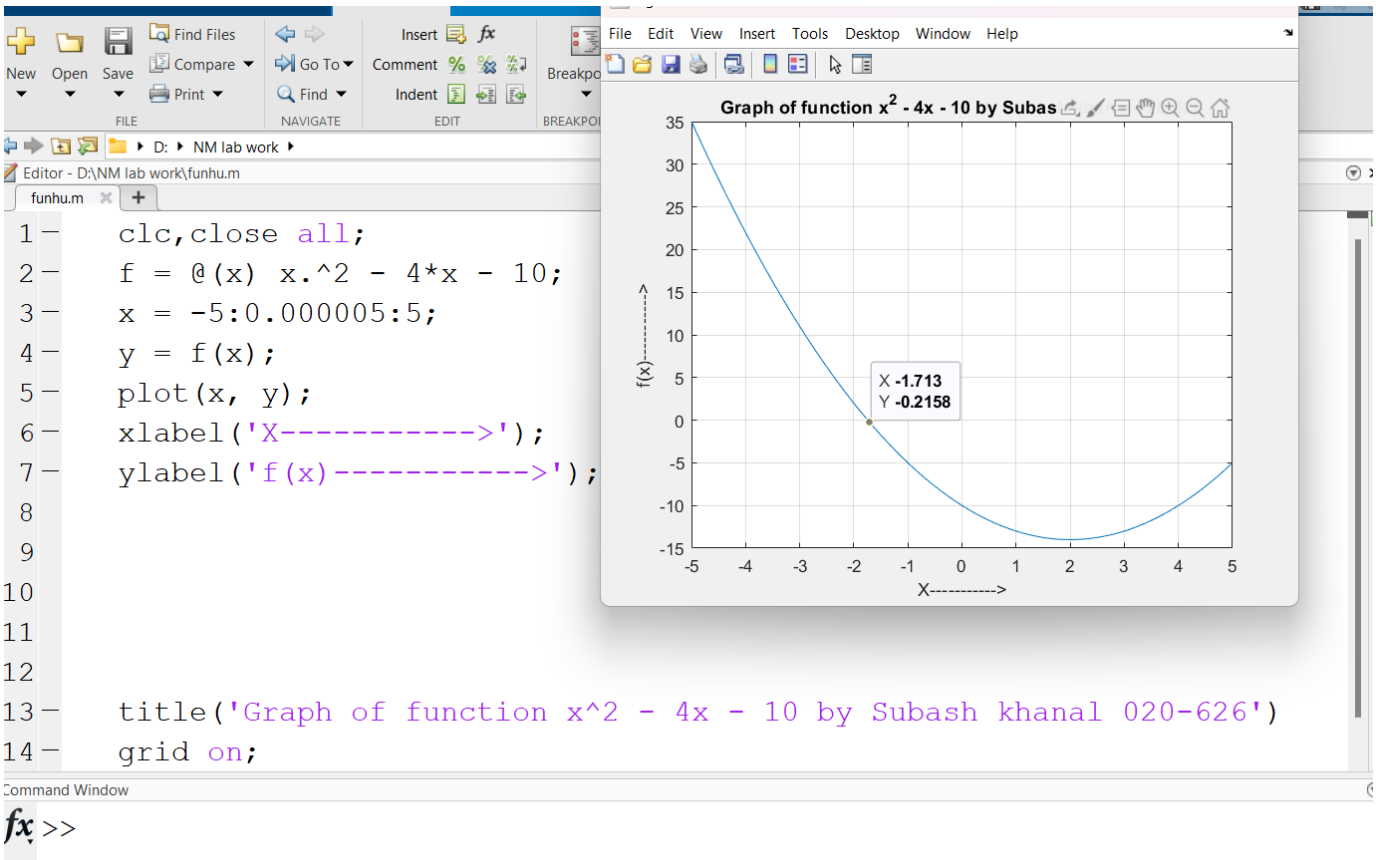

x1=x0 ,x2 =x2
X2=x0 ,x1 =x1
Print the root
STOP

**Question**: Implement above algorithm in MATLAB to calculate a root of the following equations

a) $x^2 - 4x\text{-}10=0$

Solution:-

Graph of the given Function:-




So from the above the root is nearly equal to -1.713.

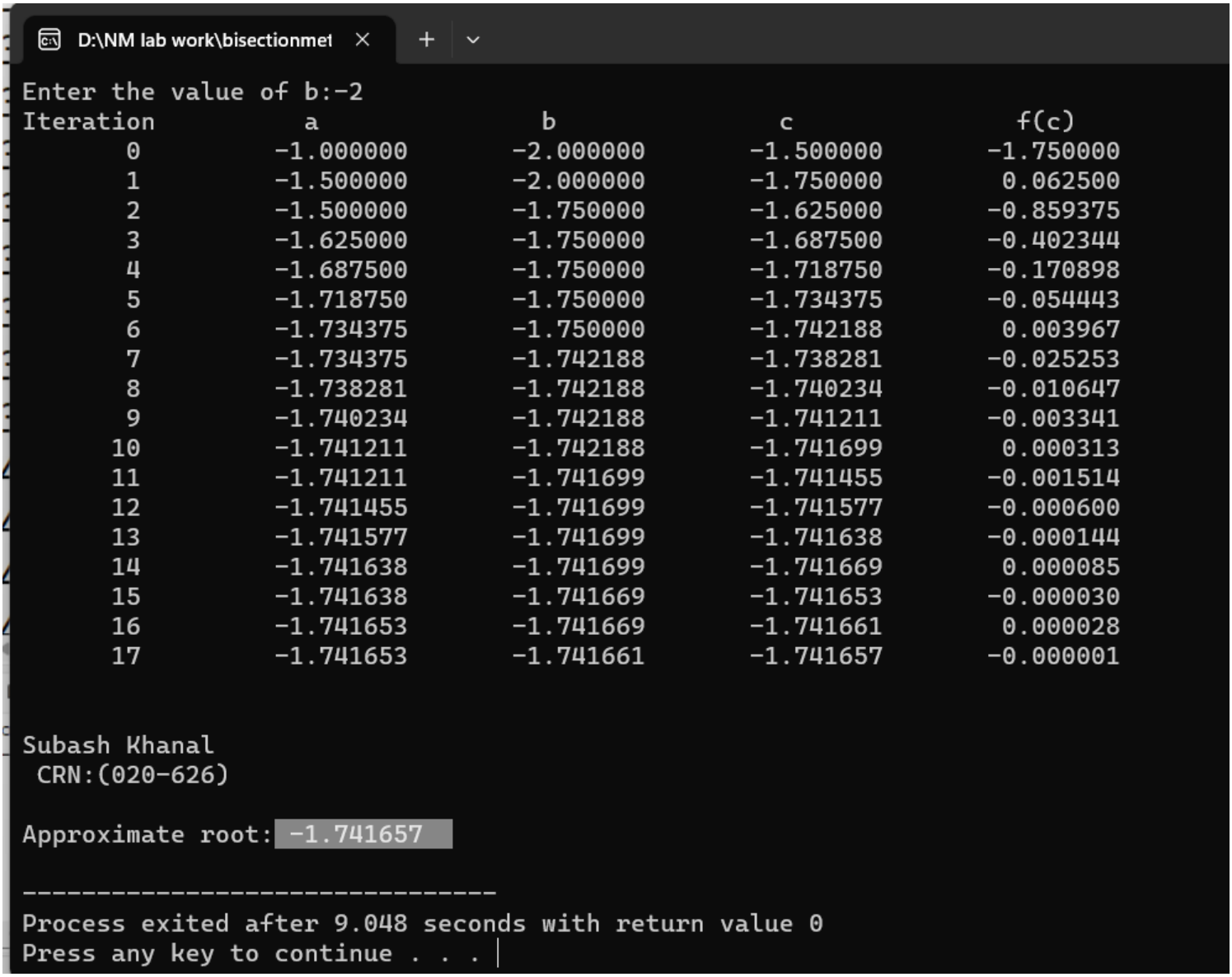
Using C-programming

Syntax:-

```
#include <stdio.h>
#include<conio.h>
#include <math.h>

double func(double x) {
    // Defining the function
    return x*x-4*x-10;
}

double bisection(double a, double b, double epsilon) {
    double c;
    int iterations = 0;

    printf("Iteration\t   a\t\t   b\t\t   c\t\t   f(c)\n");

    while (1) {
        c = (a + b) / 2.0;

        printf("%8d\t%10.6f\t%10.6f\t%10.6f\t%10.6f\n",
iterations, a, b, c, func(c));

        if (func(c) == 0.0 || fabs(b - a) < epsilon) {
            break;
        }
         if (func(c) * func(a) < 0) {
            b = c;
        } else {
            a = c;
```

```
        }

        iterations++;

        if (iterations > 100) {
            printf("Maximum iterations exceeded.\n");
            break;
        }
    }

    printf("\n\nSubash Khanal\n CRN:(020-626)\n\n");

    return c;
}

int main() {
    double a, b, epsilon;
    double root;
     printf("Enter the value of a:");
    scanf("%lf",&a);
    printf("Enter the value of b:");
    scanf("%lf",&b);
    epsilon = 0.00001;

    // Call the bisection method
    root = bisection(a, b, epsilon);
    printf("Approximate root: %lf\n", root);

    return 0;
    getch();
}
```

Output:-




The root of the given function is highlighted in the output of c-programming output which was -1.7416257 which is nearly equal to -1.713 which was interception/root from Matlab graph.

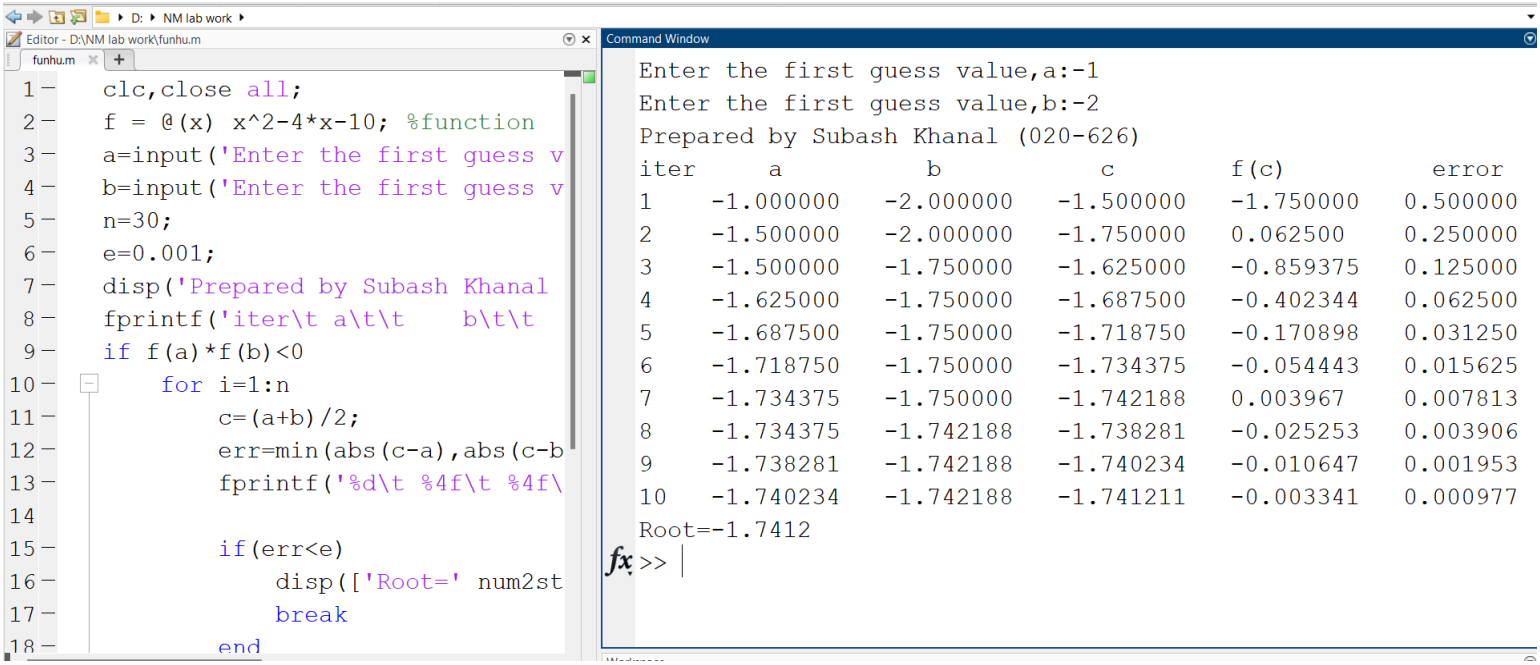
Using Matlab.

Syntax:-

```
    clc,close all;
f = @(x) x^2-4*x-10; %function
%Taking intial value from user .
a=input('Enter the first guess value,a:');
b=input('Enter the first guess value,b:');
n=30;
e=0.001;
disp('Prepared by Subash Khanal (020-
626)')
fprintf('iter\t a\t\t    b\t\t    c\t\t
f(c)\t\t   error\n')
%for printing in the table from
if f(a)*f(b)<0
    for i=1:n
        c=(a+b)/2;
        err=min(abs(c-a),abs(c-b));
        fprintf('%d\t %4f\t %4f\t %4f\t
%4f\t %4f\n',i,a,b,c,f(c),err)

        if(err<e)
            disp(['Root=' num2str(c)])
            break
        end
        if f(c)*f(a)<0 %first condition
            b=c;
        elseif f(c)*f(b)<0 %Second
condition
            a=c;
```

```
        end
    end
else
    % if the root is not define i betwwen
the guess
    disp('No root between given interval')
end
```

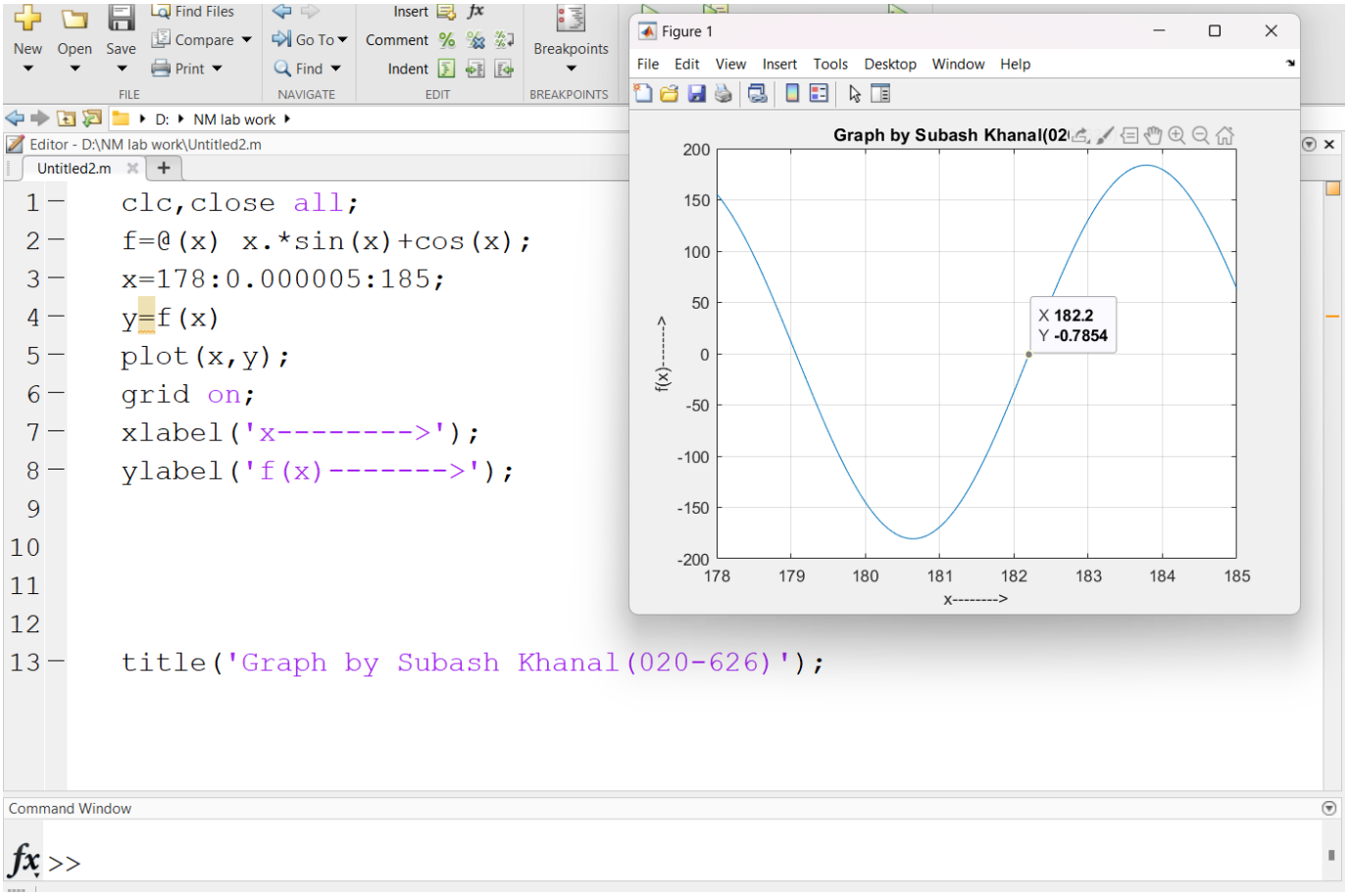
## Output:-



b) $x\sin(x) + \cos(x) = 0$

Solution:-

Graph of the given Function:-




So from the above the root is nearly equal to 182.2.

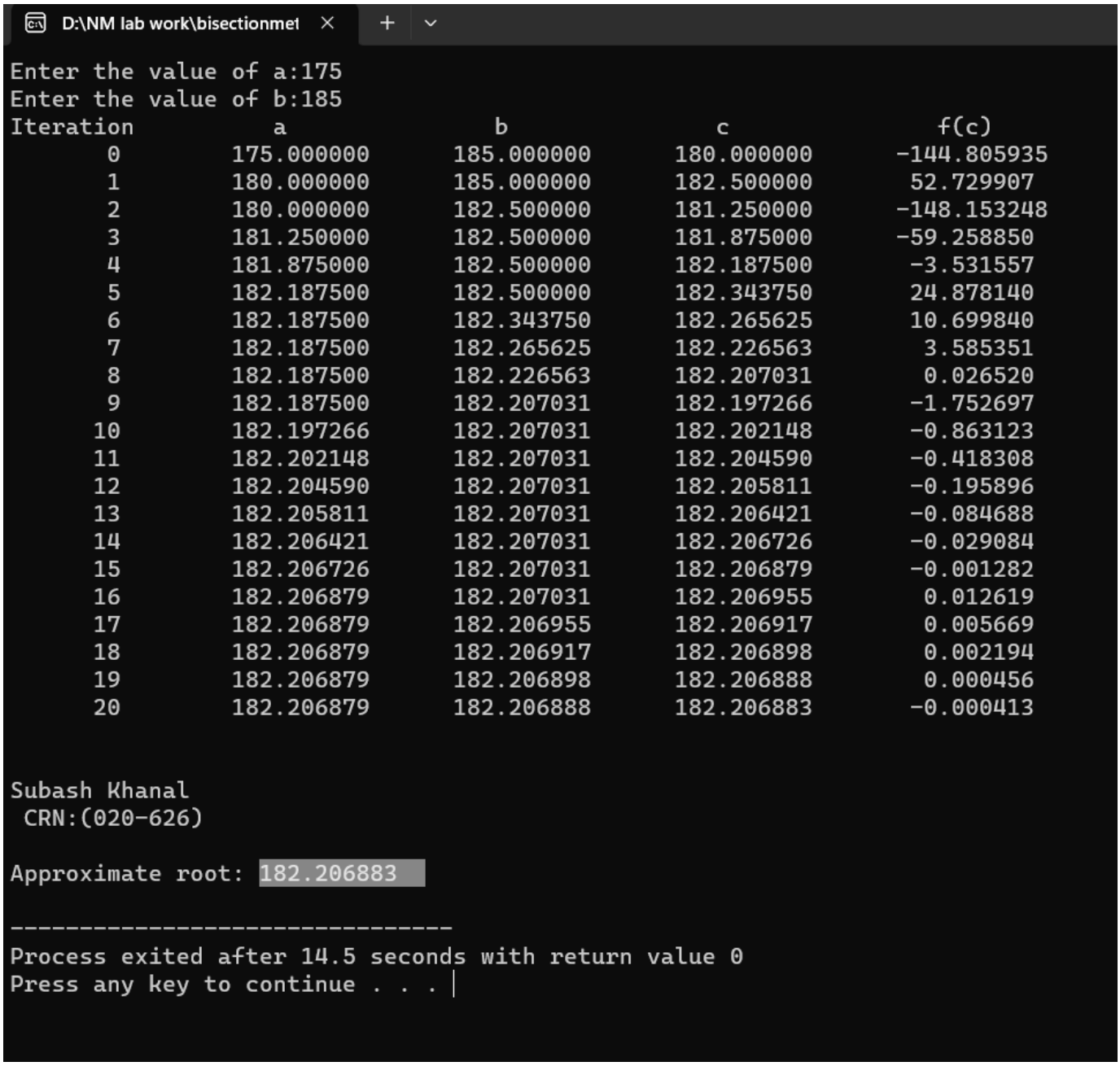
Using C-programming

Syntax:-

```c
#include <stdio.h>
#include<conio.h>
#include <math.h>

double func(double x) {
    // Defining the function
    return x*sin(x)+cos(x);
}

double bisection(double a, double b, double epsilon) {
    double c;
    int iterations = 0;

    printf("Iteration\t   a\t\t   b\t\t   c\t\t   f(c)\n");

    while (1) {
        c = (a + b) / 2.0;

        printf("%8d\t%10.6f\t%10.6f\t%10.6f\t%10.6f\n",
iterations, a, b, c, func(c));

        if (func(c) == 0.0 || fabs(b - a) < epsilon) {
            break;
        }
         if (func(c) * func(a) < 0) {
            b = c;
        } else {
            a = c;
        }
```

```
        iterations++;

        if (iterations > 100) {
            printf("Maximum iterations exceeded.\n");
            break;
        }
    }

    printf("\n\nSubash Khanal\n CRN:(020-626)\n\n");

    return c;
}

int main() {
    double a, b, epsilon;
    double root;
     printf("Enter the value of a:");
    scanf("%lf",&a);
    printf("Enter the value of b:");
    scanf("%lf",&b);
    epsilon = 0.00001;

    // Call the bisection method
    root = bisection(a, b, epsilon);
    printf("Approximate root: %lf\n", root);

    return 0;
    getch();
}
```

Output:-




The root of the given function is highlighted in the output of c-programming output which was 182.206883 which is equal to 182.20which was interception/root from Matlab graph.

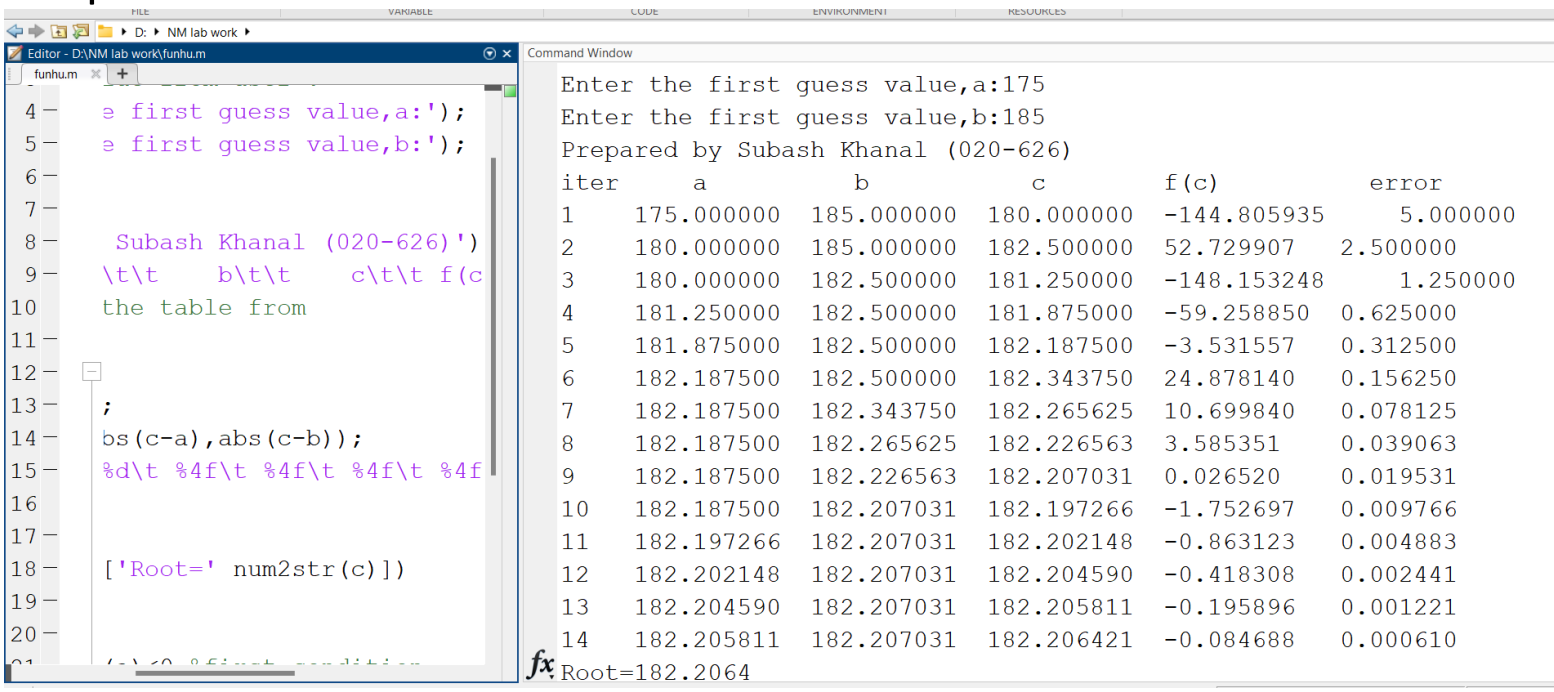
Using Matlab

Syntax:-

```
clc,close all;
f = @(x) x.*sin(x)+cos(x); %function
%Taking intial value from user .
a=input('Enter the first guess value,a:');
b=input('Enter the first guess value,b:');
n=30;
e=0.001;
disp('Prepared by Subash Khanal (020-
626)')
fprintf('iter\t a\t\t    b\t\t    c\t\t
f(c)\t\t   error\n')
%for printing in the table from
if f(a)*f(b)<0
    for i=1:n
        c=(a+b)/2;
        err=min(abs(c-a),abs(c-b));
        fprintf('%d\t %4f\t %4f\t %4f\t
%4f\t %4f\n',i,a,b,c,f(c),err)

        if(err<e)
            disp(['Root=' num2str(c)])
            break
        end
        if f(c)*f(a)<0 %first condition
```

```
            b=c;
        elseif f(c)*f(b)<0 %Second
condition
            a=c;
        end
    end
else
    % if the root is not define i betwwen
the guess
    disp('No root between given interval')
end
```

## Output: -



**Description:-**

From above program on both c and matlab it was clear that root are same using any of c-programming or matalb and also from the graph. So, using above program we can find the root.

**Conclusion:**-

Hence, from above we can implement and calculate the root using the bisection method on Matlab and C-programming.