**Bisection Method of finding root of Equation**

The bisection method is used to find the roots of polynomialequation. It separates the interval and subdivides the interval in which the root of the equation lies. The principle behind this method is the intermediate theorem for continuous functions. It works by narrowing the gap between the positive and negative intervals until it closes in on the correct answer. This method narrows the gap by taking the average of the positive and negative intervals.

**Algorithm:**

1. Start
2. Read initial guesses a, b and tolerance level E
3. Compute f1 = f(a) and f2 = f(b)
4. If (f1\*f2)>0, then end. Else follow the next steps.
5. Compute mid = (a+b)/2
6. If f(mid)<=E, then display mid as Root and exit. Else follow next steps.
7. If (f(mid)\*f1)<0, then set b = mid
8. Else set a = mid
9. Go to Step 5 and continue until root is found.
10. Stop.

**Flowchart:**

**Code:**

#include <stdio.h>

#include <math.h>

#define tol 10.e-6

double val(double x)

{

return pow(x, 3) + 4 \* pow(x, 2) - 10;

}

int main()

{

double a = 1.25, b = 1.5, mid, f\_mid;

int iterations = 0;

printf("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -\n");

printf("Iter\t\ta\t\tb\t\tc\t\tf(a)\t\tf(b)\t\tf(c)\n");

printf("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -\n");

while (1)

{

mid = (a + b) / 2;

f\_mid = val(mid);

iterations++;

printf("%d\t%.6lf\t%.6lf\t%.6lf\t%.6lf\t%.6lf\t%.6lf\n",iterations, a,b,mid,val(a),val(b),f\_mid);

if (fabs(f\_mid)<=tol)

{

printf("Root: %lf\n", mid);

break;

}

else if (f\_mid \* val(a) < 0)

b = mid;

else

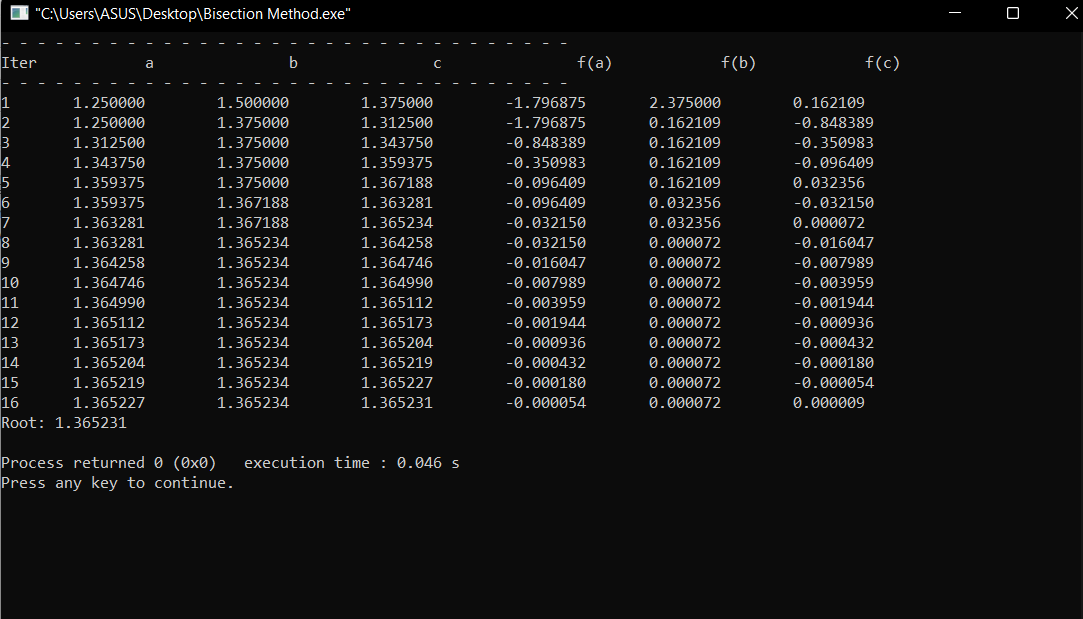
a = mid;

}

return 0;

}

**Output:**

****