ISSN 2349-0292
Impact Factor 2.365

Global Journal of Advanced Engineering Technologies and Sciences

# COMPUTER PROGRAMMING FOR ITERATIVE IMPLEMENTATION OF THE INTERMEDIATE VALUE THEOREM TO FIND THE REAL ROOTS OF A NONLINEAR FUNCTION BY BISECTION METHOD

**P.R. Kolhe*, P P Kolhe**
*Assistant Professor, Dr BSKKV Dapoli.

## Abstract

In mathematical theorem to solve iterative implementation of the Intermediate Value Theorem( IVT) to find the real roots of a nonlinear function. Manually it is very difficult task to get the required output number of iteration need to perform. In the era of Information Communication Technology (ICT) .The ICT programming technique, it is easier task. This paper discusses ICT programming techanique for an iterative implementation of the Intermediate Value Theorem to find the real roots of a nonlinear function by Bisection method. The programming effort for **Bisection Method in C** language is simple and easy.

The convergence is linear, slow but steady. Computer programming for Iterative Implementation of the Intermediate Value Theorem to Find the Real Roots of a Nonlinear Function By Bisection Method using C language Version 3.0V Turbo C++ copyright (c) 1990, 1992 by Borland International was developed.

**Keywords**: ICT ,IVT, nonlinear ,theorem , Turbo C++ .iteration.

## Introduction

In mathematical theorem to solve iterative implementation of the Intermediate Value Theorem to find the real roots of a nonlinear function. Manually it is very difficult task to get the required output number of iteration need to perform. In the era of Information Communication Technology (ICT). The computer programming technique it is easier task. Computer programming for an iterative implementation of the Intermediate Value Theorem to find the real roots of a nonlinear function by Bisection method.

According to the theorem, If a function f(x)=0 is continuous in an interval (a,b), such that f(a) and f(b) are of opposite nature or opposite signs, then there exists at least one or an odd number of roots between a and b."

## Configuration of the System

Desktop System with any configuration where at least 'C' compiler install.
(Intel® Pentium® 4, 2.0 GHz, 1 GB DDR 2-RAM, Intel 845 Series Motherboard , Nvidia Geforce 4® –440-8x AGP card , Microsoft® Windows™ XP Professional Version 2002 Service Pack 2 )

## About the platform

The programming effort for **Bisection Method in C** language using C language Version 3.0 Turbo c++ copyright (c)1990, 1992 by Borland International, Inc.

### Computer programming approach

Using C program for bisection method is one of the simplest computer programming approaches to find the solution of nonlinear equations. It requires two initial guesses and is a closed bracket method. Bisection method never fails.

The programming effort for **Bisection Method in C** language is simple and easy. The convergence is linear, slow but steady. The overall accuracy obtained is very good, so bisection method is more reliable in comparison to the Newton Raphson method or the Regula-Falsi method.

**Features of Bisection Method:**

- Type – closed bracket

- No. of initial guesses – 2
- Convergence – linear
- Rate of convergence – slow but steady
- Accuracy – good
- Programming effort – easy
- Approach – middle point
-

**C program for bisection method**

To find a root of the nonlinear function for example the equation $X^3$-4x-9 .
Computer program statement as ***x^3 – 4*x – 9***.
The initial guesses taken are a and b. The calculation is done until the following condition is satisfied:
|a-b| < 0.0005 OR If (a+b)/2 < 0.0005 (or both equal to zero) where, (a+b)/2 is the middle point value.

**Variables define in C program:**

- itr – a counter variable which keeps track of the no. of iterations performed
- maxmitr – maximum number of iterations to be performed
- x – the value of root at the nth iteration
- a, b – the limits within which the root lies
- allerr – allowed error
- x1 – the value of root at (n+1)th iteration

Example of equation taken for output:- f(x) = x^3 – 4*x – 9

**Source code in C programming language for Bisection Method**

```
/* C Program for Bisection Method Source Code */

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

 main ()
{
   int itr = 0, maxmitr;
   float x, a, b, allerr, x1;
    clrscr();
   printf("\nEnter the values of a, b, allowed error and maximum iterations:\n");
   scanf("%f %f %f %d", &a, &b, &allerr, &maxmitr);
   bisection (&x, a, b, &itr); /* sub function call 8/

   do
   {
      if (fun(a)*fun(x) < 0)
         b=x;
      else
         a=x;
      bisection (&x1, a, b, &itr);
      if (fabs(x1-x) < allerr)
      {
         printf("After %d iterations, root = %6.4f\n", itr, x1);
         return 0;
      }
      x=x1;
   }
   while (itr < maxmitr);
   printf("The solution does not converge or iterations are not sufficient");
```

```
   return 1;
}

/* function  sub program -1 */
float fun (float x)
{
   return (x*x*x - 4*x - 9);
}

/* function sub program-2*/
void bisection (float *x, float a, float b, int *itr)
/* this function performs and prints the result of one iteration */
{
   *x=(a+b)/2;
   ++(*itr);
   printf("Iteration no. %3d X = %7.5f\n", *itr, *x);
}
```

## Conclusion

Output for Bisection method for erxample equation $X^3$-4x-9 at the 11 iteration root is 2.7065 for value of a=3 and b= 2 at the error value 0.0005 at maximum iteration is 20.

**Output-**

```
Enter the values of a, b, allowed error and maximum iterations:
3 2 0.0005 20
Iteration no.   1 X = 2.50000
Iteration no.   2 X = 2.75000
Iteration no.   3 X = 2.62500
Iteration no.   4 X = 2.68750
Iteration no.   5 X = 2.71875
Iteration no.   6 X = 2.70313
Iteration no.   7 X = 2.71094
Iteration no.   8 X = 2.70703
Iteration no.   9 X = 2.70508
Iteration no.  10 X = 2.70605
Iteration no.  11 X = 2.70654
After 11 iterations, root = 2.7065
```

## Reference

1. S. Shastri " Numerical Mathematics "
2. E Balgruhaswami "Programming in C"
3. Y .Kanetkar " Let's C"