**Experiment No: 02**

**Name of the Experiment:** Study of Bisection Method to Obtain the Roots of a Nonlinear Equation.

**Objectives:**

The objective of this experiment is to find the value of root of an equation by bisection method, using Matlab for a very precise value.

**Theory:**

For Mathematics & Numerical Methods in order to find the roots, the Bisection method is a renowned one. For a Polynomial Equation f(x) =0, its roots are founded by applying this method, provided that the roots lie within the interval [a, b] and f(x) is continuous in the interval.

The input for the method is a continuous function f, an interval [a, b], and the function values f (a) and f (b). The function values are of opposite sign (there is at least one zero crossing within the interval). Each iteration performs these steps:

Calculate c, the midpoint of the interval, c = (a + b)/ 2

1. Calculate the function value at the midpoint, f(c).

2. If convergence is satisfactory (i.e., c - is sufficiently small, or |f(c)| is sufficiently small), return c and stop iterating.

3. Examine the sign of f(c) and replace either (a, f (a)) or (b, f (b)) with (c, f(c)) so that there is a zero crossing within the new interval.

**Tools:**

**Methodology:**

**Algorithm of Bisection Method:**

Step 1-- Start

Step 2-- Read x1, x2, e here x1 and x2 are initial guesses e is the absolute error i.e. the desired degree of accuracy

Step 3-- Compute: f1 = f(x1) and f2 = f(x2)

Step 4-- If (f1\*f2) > 0, then display initial guesses are wrong and go to (11). Otherwise continue.

Step 5-- x = (x1 + x2)/2.

Step 6-- If ([(x1 – x2)/x] < e), then display x and go to (11). Here [ ] refers to the modulus sign.

Step 7-- Else, f = f(x).

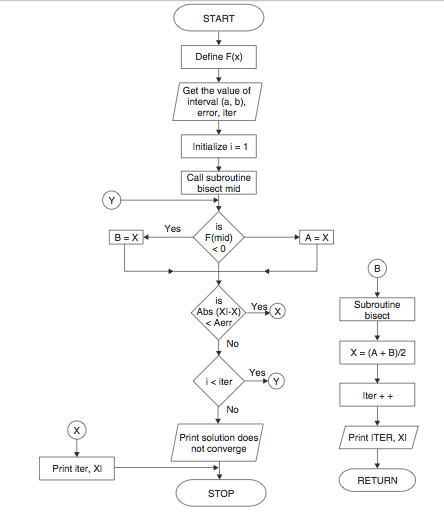
Step 8-- If ((f\*f1) > 0), then x1 = x and f1 = f.

Step 9-- Else, x2 = x and f2 = f.

Step 10-- Go to (5). Now the loop continues with new values.

Step 11—Stop.

**Flowchart:**



**Bisection Method in MATLAB Code:**

The given function is f(x) =2x^2-15x+3

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

y=@(x) 2\*x^2-15\*x+3;

x1 = input('Enter the value of x1: ');

x2 = input('Enter the value of x2: ');

error=input('Enter error allowed in the answer: ');

if y(x1)\*y(x2)>0

fprintf('No roots exists of the funtion');

return

end

if y(x1)==0

fprintf('xi is one of the roots'); % checking

return

elseif y(x2)==0

fprintf('x2 is one of the roots'); % checking

return

end

fprintf('Iteration\t x\_1\t x\_2\t xp\t y(x)\n');

for i=1:100

xp=(x1+x2)/2; % bisection

if y(x1)\*y(xp)<0

fprintf('\n%d\t %f\t %f\t %f\t %f\t %f\n ',i,x1,x2,xp,y(xp) );

x2=xp;

else

fprintf('\n%d\t %f\t %f\t %f\t %f\t %f\n ',i,x1,x2,xp,y(xp) );

x1=xp;

end

if abs(y(x1))< error

break

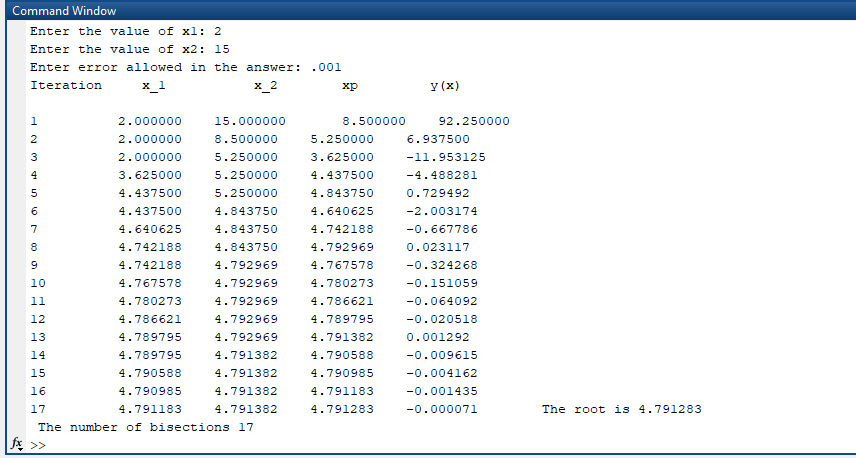
end

end

fprintf(' The root is %f \n The number of bisections %d \n',x1,i)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Output:**



**Results & Discussion:**

The resultant root of the given function is 4.781283.

**Precaution:**

1. MATLAB Code was written properly.

2. Variables are declared properly.

3. The desired degree of accuracy were kept as small as possible.

4. Appropriate syntax was maintained.

**Conclusion:**

**Discussion:**

In this experiment, we can get one root of the given nonlinear function after 17 iterations. And the value of resultant root is very close to the original root.

**References:**

<https://www.youtube.com/watch?v=fCKUOWiM-6s>

<https://www.codewithc.com/bisection-method-in-matlab/>