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

#define f(X) ((X*X*X)+4*X*X-10.0)
#define EPS 1.0e-6

double root(int n,double a,double b);

int main()
{

        double a,b;
        int i,n;
        a=1.25,b=1.5;
        n=100;

        if(!((f(a)*f(b))<0)) {
                printf("Roots are invalid\n");
                exit(0);
        }

        printf("Root(BS):%lf\n",root(n,a,b));

        return 0;
}

double root(int n,double a,double b){
        int i;
        double c;
        for(i=1;i<n;i++){
                c=(a+b)/2;
                printf("%3d %.6lf %.6lf %.6lf %.6lf %.6lf %.6lf\n",i,a,b,c,f(a),f(b),f(c));

                if(fabs(f(c))<=EPS){
                        return c;
                }

                if(f(a)*f(c)<0) b=c;
                else a=c;
        }
}
```

# Bisection Method

## Algorithm:

1. Set boundary a and b
2. Check if root exist:

   if ($f(a) * f(b) < 0$) not true

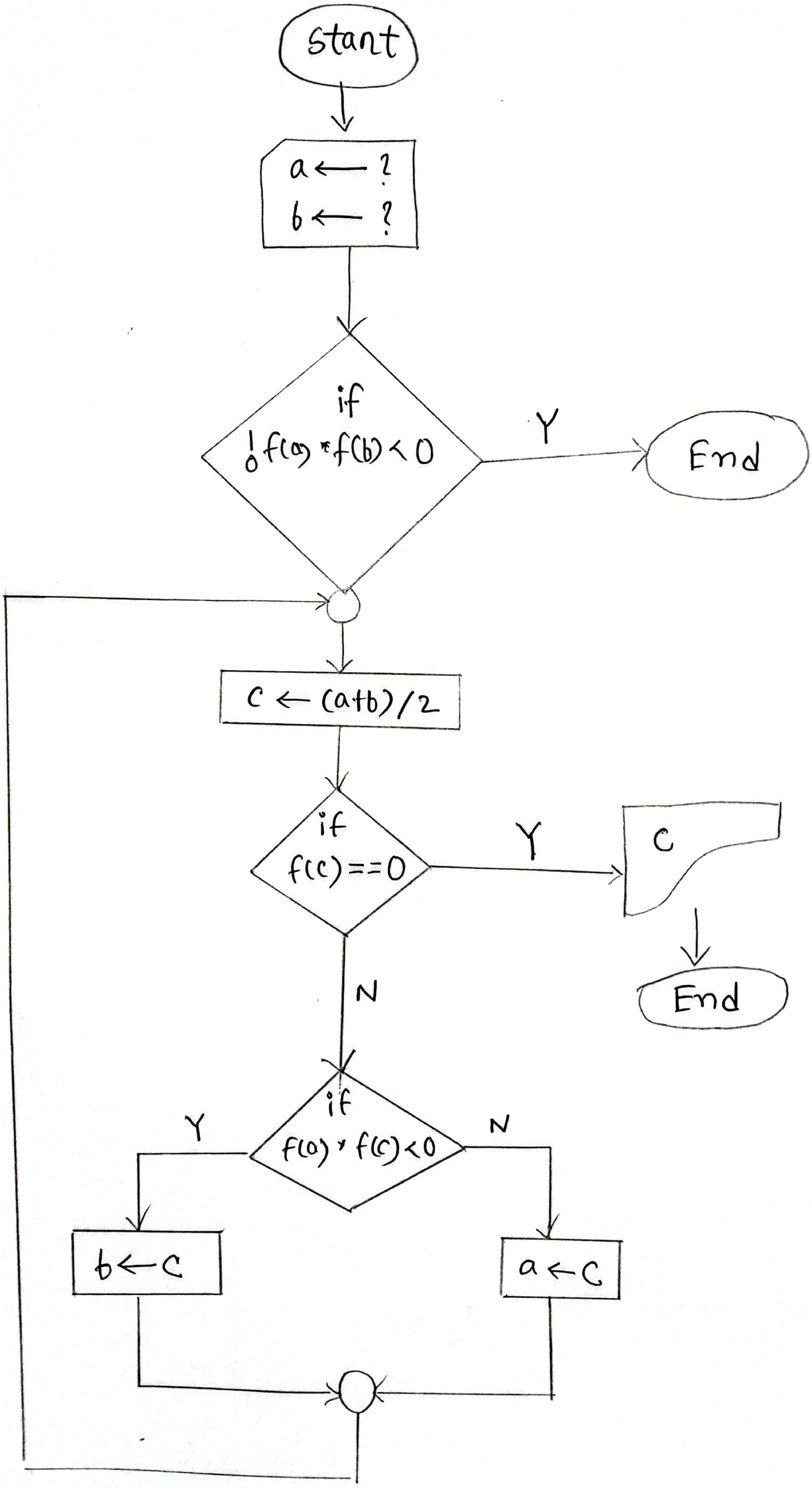
   then exit(0).
3. Els set root $c = (a+b)/2$.
4. If $f(c) = 0$, then c is the root.

   print c.
5. Else update boundary:.

   If ($f(a) * f(c) < 0$) then $b = c$

   Else $a = c$
6. Repeat from step 3 until $f(c) = 0$

# Flow Chant



Stant
a ← ?
b ← ?
if
! f(a) * f(b) < 0
Y
End
c ← (a+b)/2
if
f(c)==0
Y
c
End
N
if
f(a) * f(c) < 0
Y
N
b ← c
a ← c