Bisection Method of Root Finding

Algorithm

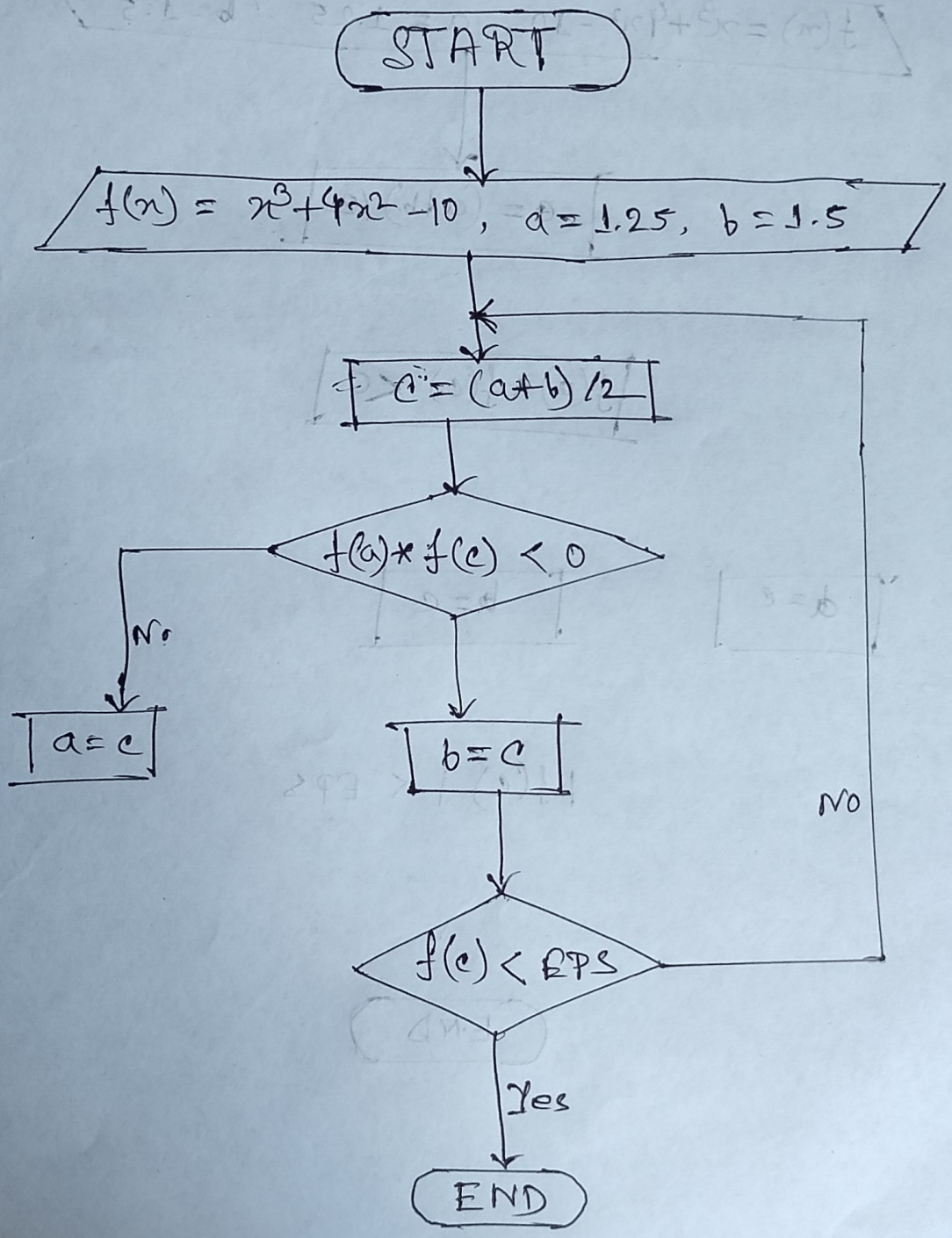
Let us consider a continuous function $f(x)$ which is defined on the closed interval $[a, b]$, is given with $f(a)$ and $f(b)$ of different signs. Then there exist a point c belong to (a, b) for which $f(c) = 0$. The iteration for approximating next root using the bisection method is, $c = \frac{a+b}{2} = b - \frac{b-a}{2}$

Follow the below procedure to get the root of the equation $f(x) = 0$;

1. Find two point say a and b such that $f(a) f(b) < 0$
2. Find the midpoint of a and b, say c.
3. c is the root of the given function if $f(c) = 0$; else follow the next steps.
4. Divide the interval $[a, b]$ — If $f(a) f(c) < 0$, there exist a root between a and c — else there exist a root between c and b.
5. Repeat ~~fill~~ from step 2 until $f(c) = 0$

Bisection Method of Root Finding

Flow Chart



Bisection Method of Root Finding

Code

```
# include <stdio.h>
# include <stdlib.h>
# include <math.h>
# define EPS 10.0e-6
# define F(x) ((x)*(x)*(x)+4*(x)*(x)-10)

int main(void) {
    int n=100, i;
    double a=1.25, b=1.5, c;
    if (F(a) * F(b) > 0) exit(0);
    for(int i=1; i<n; i++) {
        c=(a+b)/2;
        if((fabs(F(c)) < EPS)) {
            printf("Root := %lf \n", c);
            exit(0);
        }
        if (F(a) * F(c) < 0) b=c;
        else a=c;
    }
}
```