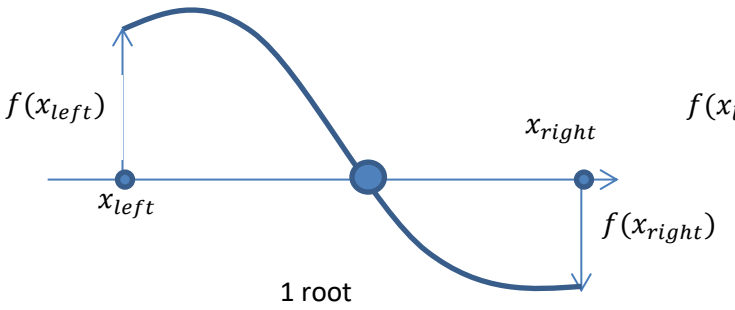
QUESTION 1 – Numerical Analysis in C

Iterative approximations for root finding is an important concept in numerical analysis. There are various methods used for this purpose, e.g. Bisection.

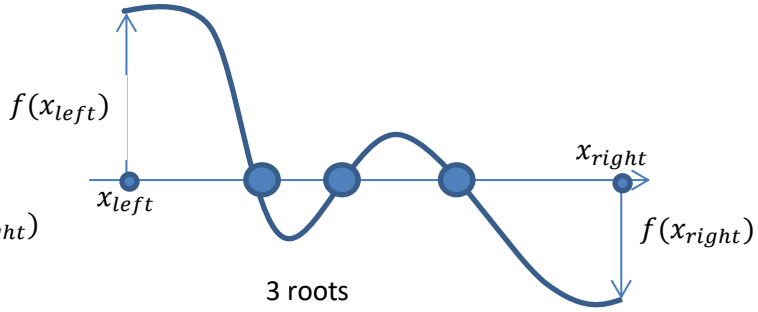
First lets see the basics of the problem:

k is a root of an equation $f(x) = 0$ if $f(k) = 0$

Bisection methods continues iteratively until either the true root is found or approximates by less than epsilon. The method first checks whether there is a sign change in the given interval, e.g. :

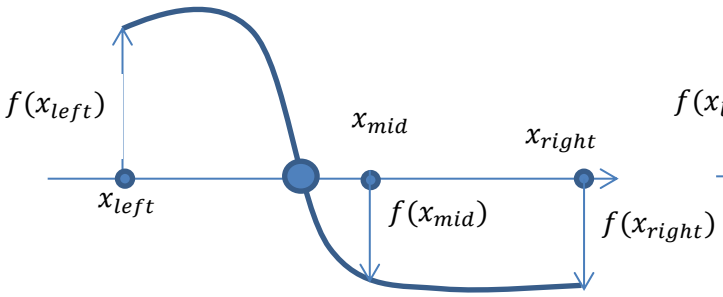


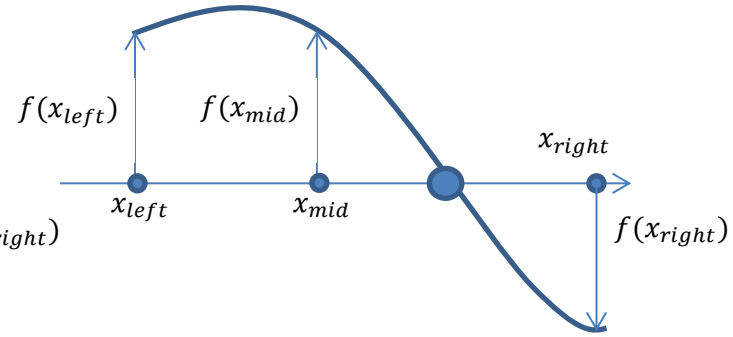
1 root

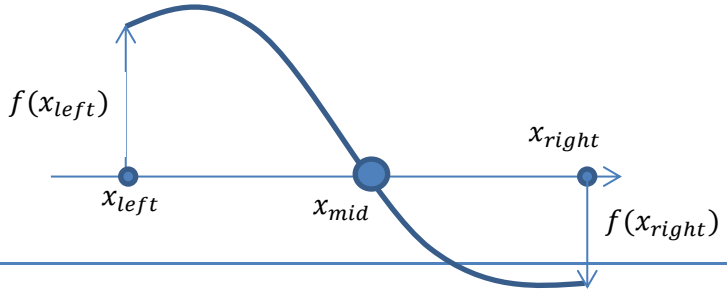


3 roots

There are 3 possibilities when the interval $[x_{left}, x_{right}]$ is bisected, either the root is in the half interval $[x_{left}, x_{mid}]$, root is in the half interval $[x_{mid}, x_{right}]$, or $f(x_{mid}) = 0$







For an arbitrary given function g→$5x^3 - 2x^2 + 3$, and h→$x^4 - 3x^2 - 8$, how can you apply Bisection method in C?

Hint: Your program should contain 4 functions:

- int main(void)
- double bisect(double xleft, double xright, double epsilon, double f(double fargs))
- double g(double x)
- double h(double x)

```
Enter interval endpoints> -1.0 0.0

Enter tolerance> 0.001


Function g
New interval is [-1.0000000, -0.5000000]
New interval is [-0.7500000, -0.5000000]
New interval is [-0.7500000, -0.6250000]
New interval is [-0.7500000, -0.6875000]
New interval is [-0.7500000, -0.7187500]
New interval is [-0.7343750, -0.7187500]
New interval is [-0.7343750, -0.7265625]
New interval is [-0.7304688, -0.7265625]
New interval is [-0.7304688, -0.7285156]
New interval is [-0.7294922, -0.7285156]
   g(-0.7290039) = -2.697494e-005


Function h
May be no root in [-1.0000000, 0.0000000]
```

QUESTION 2 - Recursion

Fill in the following recursive function that calculates the value of a number (base) raised to a power. Assume that power is a nonnegative integer.

```
int power_raiser(int base, int power)

{

        int ans;

        if(power == .....)

                ans = .....;

        else

                ans = .....*.....;

        return ans;

}
```

QUESTION 3 – Recursion

Write a recursive function which checks whether a given number is prime or not.