C Language: Algorithms

Text: Chapter 4 of *The C Programming Language* (B. W. Kernighan & D. Ritchie)

Assignment I: Root Finding

In this assignment you will implement a program that finds the root of a polynomial using the bisection method.

Suppose we have a polynomial $f(x) = 4 - x^3$. We want to find the value of x that makes $f(x) = 0$. (This value, call it t, is the cube root of 4. In fact, we can use the bisection method to find the n^{th} root of a number.) If we know that the function $f(x)$ is definitely 0 somewhere inside an interval $[l, r]$, and that $f(x)$ is strictly increasing (or decreasing) over that interval, then the bisection method is a fast way of finding the *root* of $f(x)$ to within a good level of accuracy.

Our job then is to describe the bisection method succinctly: We assume that $f(x)$ is strictly increasing over the interval chosen (call it $[l, r]$). Suppose also that somewhere inside the interval, $f(x)$ is zero. Then clearly $f(l) < 0$ and $f(r) > 0$. We can find the point t where $f(t) = 0$ by progressively splitting the interval, choosing smaller and smaller intervals until we find t, or the interval is small enough that we consider our answer found. In particular, take a point z that is the mid-point of the interval $[l, r]$. If $f(z) > 0$ then clearly the root of $f(x)$ is to be found in the interval $[l, z]$. Similarly, if $f(z) < 0$ then …
What this means is that we can take the new interval $[l, z]$ – replacing r with z – and ask the same question of its mid-point! We can keep doing this until we find z such that $f(z) = 0$ or until the interval $[l, z]$ (or $[z, r]$ for that matter) is small.

In this assignment you will implement this bisection method for root finding. Write a function `bisect()`, which calls your polynomial function `f()` repeatedly until it finds the root to within 3 decimal places. Check your `bisect()` function using $f(x) = x^2 - 2$, whose root I am sure we all know! Use your program to compute the 7^{th} root of 126, the cube root of 43 and the 5^{th} root of 729. Be careful to choose the right initial interval!
Submit the printed source code for your program.

Note: You should write separate functions for `bisect()` and the polynomial it iterates on.
Challenge for the bored: Implement the Newton method for root finding and use it to find the above roots. Do this on addition to the problem above.

This assignment is due 1 week from the 20^{th} of June, 2022.