Project 3 due by February 25. You must make an appointment to visit my office on or berfore February 25, at a mutually agreeable time, in order to demonstrate the programming part of this project for me.

1. Let $f(x) = (x-3)^4(x-2)^3(x-1)x^2(x+1)(x+2)^4(x+3)^3$. To which zero of f does the bisection method converge when applied on the interval $[-2.3, 3.1]$?

2. Let

$$g(x) = \frac{4x}{5} + \frac{2}{5x^4}. \tag{1}$$

a. find the positive fixed point of g.

b. find an interval $[a, b]$, containing the fixed point found in part (a), in which g maps this interval onto itself and g is a contraction mapping on this interval. The endpoints of the interval are not allowed to coincide with the fixed point.

c. What is the maximum number of fixed point iterations needed in order to find the fixed point of g with 10^{-10} accuracy? The starting point should be one of the endpoints of the interval found in part (b).

3. (root finding and the gamma function) The Taylor series approximation for e^x expanded about $x_0 = 0$ is

$$P_n(x) = 1 + x + x^2/2 + x^3/(3!) + \ldots + x^n/(n!) \tag{2}$$

and the remainder term is:

$$R_n(x) = \frac{e^{\xi(x)}}{(n+1)!}x^{n+1}. \tag{3}$$

If $0 \le x \le x_{max}$, then a bound on the error is:

$$\frac{e^{x_{max}}}{(n+1)!}x_{max}^{n+1}. \tag{4}$$

The number of terms n in the Taylor series needed in order to meet a prescribed tolerance ϵ satisfies the following equation:

$$\frac{e^{x_{max}}}{(n+1)!}x_{max}^{n+1} \le \epsilon. \tag{5}$$

Since $\Gamma(n) = (n-1)!$ where n is an integer, we can estimate the number of terms by finding the roots of the following function:

$$f(t) = \frac{e^{x_{max}}}{\Gamma(t+2)}x_{max}^{t+1} - \epsilon. \tag{6}$$

(a) Write a bisection method algorithm to solve $f(t) = 0$ for t. The user of your computer program should be able to prescribe x_{max} and ϵ. You are allowed to assume that x_{max} and ϵ are chosen such that $f(0) > 0$. Your program first has to search for a value of t that satisfied $f(t) < 0$. So your starting interval will be $[a, b]$ where $a = 0$ and $f(b) < 0$.

(b) repeat part (a) except using the "weighted bisection method." Instead of $c = (a+b)/2$ at each iteration, the following *weighted* algorithm has:

$$c = \frac{|f(b)|a + |f(a)|b}{|f(a)| + |f(b)|} \tag{7}$$

Compare and contrast (in terms of complexity and number of iterations required for convergence) this alternate "weighted bisection method" to the standard bisection method from part (a).

Notes:

- The Gamma function $\Gamma(t)$ is available in most c++ compilers (after including the appropriate header files) and is defined as

$$\Gamma(t) = \int_0^\infty x^{t-1} e^{-x} dx. \tag{8}$$

- A sample code for finding the roots of a given function $f(t)$, but using Newton's method, is posted online under course library.