MTH 351, HW Assignment 2

The goal of this assignment is to explore a few methods of *root-finding*, that is, methods for solving

$$f(x) = 0. \qquad (1)$$

In fact, we seek a solution of the problem $x = exp(-x)$ which is set up in the form (1) using either $f_1(x) = x - exp(-x)$, or $f_2(x) = 1000xexp(x) - 1000$, or $f_3(x) = 1/x - exp(x)$.

Notes and additional info: Let x^* be the true solution to (1). Recall that a method has convergence rate C /order α if the error $e_{n+1} := |x^* - x_{n+1}| \leq Ce_n^\alpha$ for iterates $n = 1, 2, 3, \ldots$. A good estimate for α can be provided by calculating

$$\alpha \approx \frac{\log(e_{n+1}) - \log(e_n)}{\log(e_n) - \log(e_{n-1})}$$

for a few subsequent iterations. Note that in order to do that, one must know the *exact* root x^*. If it is not known, then e_n cannot be calculated ! IHowever, in Newton (and secant) method a good *estimate* for e_n is $e_n \approx \upsilon_n = x_{n+1} - x_n$. (For fixed-point iteration one can use Aitken's formula, discussed in Section 3.4.1 of the textbook).

1. Use bisection method to solve the problem (1) with each f_1, f_2, f_3 (if possible) on interval [0,1] with tolerance=1.e-4 and max # iterations 20. Note: you can use Atkinson/Han's MATLAB code. I DO NOT NEED TO SEE YOUR CODE. If there is a problem for any f_i, modify the setup to eliminate this problem. How many iterations did the algorithm take for each f_1, f_2, f_3 ? Discuss.

Pick only one of f_1, f_2, f_3 and predict how many iterations would be needed to solve the problem i) with tolerance 1e-6, and ii) to machine precision (call the solution `xbis`). Verify in MATLAB. Is $f_i(xbis) = 0$?. Comment.

2. Use Newton method to solve the problem as in #1. Note: you can use Atkinson/Han's MATLAB code appropriately modified. DO NOT SHOW THE CODE.

Explore sensitivity to the starting point and choice of the function: try using $x_0 = 1, 0, -1, 10$ for each f_i, if possible. Use tolerance eps=1.e-10. Problems ? For which f_i does the method appear to converge the fastest ? Discuss. Call one good converged solution `xnewt`.

3. Pick the same f_i as in #2 and investigate convergence rate of the Newton method: assume x^* is known and equal to `xnewt` obtained in #2. Is α *really* equal 2 as predicted by theory ? What if we do not know x^* and only use estimates υ_n of e_n ?

Note: here you have to write your own code or modify Atkinson/Han's M-file `newton.m` and make it deliver, for each iteration n, the error ε_n and its estimate υ_n. SHOW THE CODE.

4. Write your own code to find the solution of a problem $x = exp(-\beta x)$ using the *fixed-point* method with an initial guess $x_0 = 0$. SHOW THE CODE. Try $\beta = 10, 2, 1, 0.1$. For what β are we guaranteed that the fixed-point method

converges ? Does it actually converge ? Estimate the convergence rate (if applicable) for $\beta = 0.1$ (use Aitken's formula)

Extra: do # 3 with secant method.