MA/CS 321: Intro to Numerical Methods

HW 3

Due: Saturday September 20, 2014

Note: Turn in all code that you have written and turn in all output generated by your functions/scripts. Your functions/scripts must be commented and output must be formatted nicely.

1. (2.4.3) Consider using the partial sum

\[ S_n(x) = \sum_{j=0}^{n} (-1)^j \frac{x^{2j+1}}{(2j+1)!} \]

to approximate \( \sin x \). For \( x = 0.1, 1, \) and \( 10 \), calculate \( S_n(x) \) by both methods LS and SL, and compare the results against \( \sin x \). Use \( n = 10, 100, 1000 \).

2. (3.1.1 b and f) Use the bisection method and Newton’s method with a hand calculator or computer to find the indicated roots of the following equations. Use an error tolerance of \( \varepsilon = 0.0001 \). Note the root and the number of iterations required for bisection and Newton’s method to meet the desired tolerance.

(a) The smallest positive root of \( x = 1 + 0.3 \cos(x) \).
(b) The real root of \( x^3 - 2x - 2 = 0 \).

3. (3.1.1.e & 3.1.10) Consider the equation \( e^{-x} = \sin x \). Find an interval \([a, b]\) that contains the smallest positive root. Estimate the number of midpoints \( c \) needed to obtain an approximate root that is accurate with an error tolerance of \( 10^{-10} \). Use the bisection method to find this root.

4. (3.1.7 & 3.2.10) Using bisection, solve the equation

\[ f(x) \equiv x^3 - 3x^2 + 3x - 1 = 0 \]

with an accuracy of \( \varepsilon = 10^{-6} \). (Be sure to print your computed root with at least six digits of precision.) Experiment with the following ways of evaluating \( f(x) \): use (i) the given form, (ii) the nested form

\[ f(x) = -1 + x(3 + x(-3 + x)) \]

Try the following initial intervals \([a, b]\), for example, \([0, 1.5], [0.5, 2.0], \) and \([0.5, 1.1]\). Explain the results. Note that \( \alpha = 1 \) is the only root of \( f(x) = 0 \).

Also solve this equation using Newton’s Method. Experiment with the choice of the initial guess \( x_0 \). Also experiment with different ways of evaluating \( f(x) \) and \( f'(x) \). Note any unusual behavior in the iteration. For at least one of your choices for \( x_0 \) make a table similar to Table 3.2 showing how quickly the \( x_i \) converge to \( \alpha \).
5. The equation
\[ f(x) = x + e^{-Bx^2} \cos(x) = 0, \quad B > 0 \]
has a unique root, and it is in the interval (-1,0). Use Newton’s method to find it as accurately as possible. Use values of \( B = 1, 5, 10, 25, 50 \). Among your choices of \( x_0 \), choose \( x_0 = 0 \), and explain the behavior observed in the iterates for the larger values of \( B \).

*Hint:* Draw a graph of \( f(x) \) to better understand the behavior of this function.

The following require some basic proofs using convergence, limits, and mathematical notation. Each of the following problems may be substitute for one of problems 1-5.

6. (2.3.6) Consider evaluating \( p = a_1a_2 \cdots a_n \) with \( a_i = \text{fl}(a_i), \ i = 1, 2, \ldots, n \). Define
\[
p_2 = \text{fl}(a_1a_2), \quad p_3 = \text{fl}(p_2a_3), \quad \ldots, \quad p_n = \text{fl}(p_{n-1}a_n)
\]
Using the type of argument applied in deriving (2.50), derive an estimate for \( p_n - p \) and \( \text{Rel}(p_n) \), showing the effect of the rounding or chopping errors that occur in forming \( p_2, \ldots, p_n \).

7. (3.1.14) Imagine finding a root \( \alpha \) satisfying \( 1 < \alpha < 2 \). If you are using a binary computer with \( m \) binary digits in its significand, what is the smallest error tolerance that makes sense in finding an approximation to \( \alpha \). If the original interval \( [a, b] = [1, 2] \), how many interval halvings are needed to find a n approximation \( c_n \) to \( \alpha \) with the maximum accuracy possible for this computer?

8. (3.2.3) On most computers, the computation of \( \sqrt{a} \) is based on Newton’s method.

(a) Set up the Newton iteration for solving \( x^2 - a = 0 \), and show that it can be written in the form
\[
x_{n+1} = \frac{1}{2} \left( x_n + \frac{a}{x_n} \right), \quad n \geq 0
\]

(b) Derive the error and relative error formulas
\[
\sqrt{a} - x_{n+1} = -\frac{1}{2x_n} \left( \sqrt{a} - x_n \right)^2
\]
\[
\text{Rel}(x_{n+1}) = -\frac{\sqrt{a}}{2x_n} \left( \text{Rel}(x_n) \right)^2
\]

(c) For \( x_0 \) near \( \sqrt{a} \), the last formula becomes
\[
\text{Rel}(x_{n+1}) \approx -\frac{1}{2} \left( \text{Rel}(x_n) \right)^2, \quad n \geq 0
\]
Assuming \( \text{Rel}(x_0) = 0.1 \), use this formula to estimate the relative error in \( x_1, x_2, x_3, \) and \( x_4 \).