Assignment 5.

Due Monday, Feb. 24, at 2:30pm PST.

Reading: Ch 4 in the Gautschi book.

1. Use MATLAB (or another programming language) to plot the function $f(x) = (5 - x)\exp(x) - 5$, for x between 0 and 5. (This function is associated with the *Wien radiation law*, which gives a method to estimate the surface temperature of a star.)

   (a) Write a bisection routine to find a root of $f(x)$ in the interval $[4, 5]$, accurate to six decimal places. At each step, print out the endpoints of the smallest interval known to contain a root. Without running the code further, answer the following: How many steps would be required to reduce the size of this interval to 10^{-12}? Explain your answer.

   (b) Write a routine to use Newton's method to find a root of $f(x)$, using initial guess $x_0 = 5$. Print out your approximate solution x_k and the value of $f(x_k)$ at each step and run until $|f(x_k)| \leq 10^{-8}$. Without running the code further, but perhaps using information from your code about the rate at which $|f(x_k)|$ is reduced, can you estimate how many more steps would be required to make $|f(x_k)| \leq 10^{-16}$ (assuming that your machine carried enough decimal places to do this)? Explain your answer.

   (c) Modify your routine for doing Newton's method to run the secant method. Repeat the run of part (b), using, say, $x_0 = 4$ and $x_1 = 5$, and again predict (without running the code further) how many steps would be required to reduce $|f(x_k)|$ below 10^{-16} (assuming that your machine carried enough decimal places to do this) using the secant method. Explain your answer.

2. p. 294, problem 11 in [Gautschi].

3. If you enter a number into a handheld calculator and repeatedly press the cosine button, what number (approximately) will eventually appear? Provide a proof. [Note: Set your calculator to interpret numbers as radians rather than degrees; otherwise you will get a different answer.]

4. In hw4, problem 4, you were asked to construct the 2-point weighted Gauss quadrature formula for the interval $[0, 1]$ with weight function $w(x) = x$; that is, to find a formula of the form
$$\int_0^1 x f(x)\, dx \approx a_0 f(x_0) + a_1 f(x_1)$$

that is exact for all polynomials of degree 3 or less. Suppose you did not know that x_0 and x_1 are roots of the second degree orthogonal polynomial for the weight function x and you instead decided to solve the nonlinear system of equations

$$\int_0^1 x \cdot x^j \, dx = a_0 x_0^j + a_1 x_1^j, \quad j = 0, 1, 2, 3$$

for a_0, a_1, x_0, and x_1. Write this set of equations in the form $\vec{f}(\vec{y}) = \vec{0}$, where the entries of $\vec{y}$ are the unknown values a_0, a_1, x_0, x_1, and write a formula for the Jacobian of $\vec{f}$. Determine (either analytically or numerically with Matlab) if the Jacobian matrix is nonsingular at the solution: $a_0, a_1 = 1/4 \pm \sqrt{6}/36$, $x_0, x_1 = (6 \pm \sqrt{6})/10$. Also determine some values for a_0, a_1, x_0, x_1 that would lead to a singular Jacobian matrix and hence, for which Newton's method with those values as initial guesses would fail.

5. The equation of motion of a pendulum of length L meters is

$$\theta''(t) = -(g/L)\sin(\theta(t)),$$

where θ is the angle of the pendulum with the downward vertical axis and $g \approx 9.8 m/s^2$ is the gravitational constant. For simplicity, assume that $g/L = 1$. Suppose we wish to set the pendulum in motion, swinging from some given initial position $\theta(0) = \alpha$ with some unknown angular velocity $\theta'(0)$ in such a way that the pendulum ends up at a desired position β at time $t = T$. Then we must solve the following 2-point boundary value problem:

$$\theta''(t) = -\sin(\theta(t)), \quad 0 < t < T,$$
$$\theta(0) = \alpha, \quad \theta(T) = \beta.$$

We saw in a previous homework how to approximate the second derivative with a finite difference quotient:

$$\theta''(t) \approx \frac{\theta(t+h) + \theta(t-h) - 2\theta(t)}{h^2}.$$

If we divide the interval $[0, T]$ into n subintervals, each of width $h = T/n$, and let θ_i denote the approximate value of θ at $t = ih$, $i = 1, \ldots, n-1$, then we end up with a system of $n-1$ equations in $n-1$ unknowns:

$$\frac{1}{h^2}(\theta_{i-1} - 2\theta_i + \theta_{i+1}) + \sin(\theta_i) = 0, \quad i = 1, \ldots, n-1,$$

where $\theta_0 = \alpha$ and $\theta_n = \beta$.

Taking $T = 2\pi$, $\alpha = \beta = 0.7$, write down the Jacobian matrix corresponding to this nonlinear system of equations and the Newton iteration that you would use to solve this system. Write a code to solve this system using $\theta_i^{(0)} = 0.7\cos(t_i) + 0.5\sin(t_i)$, $i = 1, \ldots, n-1$ as an initial guess. Plot the approximate solution at each step of Newton's method. Try some different initial guesses and see if you can converge to different solutions, again plotting the approximations from Newton's method at each step. Turn in your plots and discuss your results.