Scientific Computing Homework
Given: Wednesday, Nov. 1, Due: Nov. 8

Programming assignment: This assignment will use Matlab to solve for the four roots of unity, that is, to find a solution to $w^4 = 1$. We will use Newton's method to find a root. There are four roots – we will keep track of different starting values, which root they converge to using Newton's method, and plot the results. The resulting figure is a beautiful example of a two dimensional fractal, and is related to something called the *Mandelbrot* set.

1. First, write an m file that iterates using Newton's method. It should call a function to evaluate $f(x)$ and a different function to evaluate the derivative $f'(x)$, rather than having them hardwired into your driver program.

   Test your program on a problem with a known answer, and demonstrate that it converges quadratically. Explain this all in your writeup. (The parameters for the final computation below may take a long time, so do not use it for debugging).

2. Next, we will use many discrete points in the complex plan as initial values for finding a root of $w^4 = 1$. Let $z = a + bi$ be the initial starting value for Newton's method, where a goes from -1 to 1 in steps of .01, and the same for b (don't use these initially until after your program is debugged). Note that Matlab treats complex numbers the same way it handles real values, so your same Newton routine should work without change. To form complex numbers, Matlab routines *real* and *imag* may be useful, or simply use the built-in number i. Each starting value will converge to one of the four roots of unity (if it converges at all - so make sure you have an maximum number of iterations in your program. Which starting values do not converge?). Assign a value to each initial guess that indicates which root it converged to. For example, if we label the four roots $w_i, i = 1, 4$, then if z_k converges to w_2, assign $c(z_k) = 2$. When it's all done, plot c. Experiment with using the graphics routines in Matlab called *contourf* or *surf* for this. For a faster program using vector Matlab, (this is much faster than iterating one value as a time), look at the function *meshgrid*.