CMPS 251-Numerical Computing
Assignment 4
Due Monday, November 9, 2015

Reading Material:
- Cheney & Kincaid: Sections 3.1-3.3

Notes: You are encouraged in work individually on the assignment. Piazza can be used to ask questions (without requesting a solution !!).

Problem 1 *Practicing with root finding techniques*

Consider the following function $f(x) = x - 2e^{-x}$.
Determine the root of this equation using the following

1) Bisection method starting with $a = 0$ and $b = 1$. Show the first three iterations.

2) Regula Falsi method starting with $a = 0$ and $b = 1$. Show the first three iterations.

3) Secant method starting with $x_0 = 0$ and $x_1 = 1$. Show the first three iterations.

4) Newton method starting with $x_0 = 1$. Show the first three iterations.

Now find the exact value in MATLAB either using the `fzero()` command or using the following lines of code

```
syms x
f=x-2*exp(-x);
t=solve(f==0);
t=eval(t)
```

Calculate the relative error between the exact solution and the value obtained after three iteration. Comment on the results.

Problem 2 *System of nonlinear equations*

Using the Taylor series in two variables (x, y) of the form

$$f(x + h, y + k) = f(x, y) + hf_x(x, y) + kf_y(x, y) + \cdots$$

where $f_x = \partial f/\partial x$ and $f_y = \partial f/\partial y$, establish that Newton's method for solving the two simultaneous nonlinear equations

$$\begin{cases} f(x, y) = 0 \\ g(x, y) = 0 \end{cases}$$

can be described with the following formulas

$$x_{n+1} = x_n - \frac{fg_y - gf_y}{f_xg_y - g_xf_y} \qquad y_{n+1} = y_n - \frac{f_xg - g_xf}{f_xg_y - g_xf_y}$$

Here, the functions f, f_x, f_y, g, g_x, and g_y are evaluated at (x_n, y_n).

Now consider the following system of nonlinear equations

$$\begin{cases} -2x^3 + 3y^2 + 42 = 0 \\ 5x^2 + 3y^3 - 69 = 0 \end{cases}$$

With a sharp eye ☺, we find that the exact solution is $x = 3, y = 2$. Now, you need to solve it numerically.

1) Use Newton's method starting with $x_0 = 1$ and $y_0 = 1$. Show the first five iterations. Note that you can use the expressions derived above or the approach discussed in the slides (both are equivalent).

2) Use fixed point iterations. Use the following iteration function

$$y_{n+1} = \left(\frac{-5x_n^2 + 69}{3}\right)^{1/3} \qquad x_{n+1} = \left(\frac{3y_n^2 + 42}{2}\right)^{1/3}$$

   starting with $x_0 = 1$ and $y_0 = 1$. Show the first five iterations.

Problem 3 *Convergence of the fixed point iteration*

In class, we argued that the fixed point iteration technique converges if, in the region of interest, $|g'(x)| < 1$. In other words, convergence occurs if the magnitude of the slope of $g(x)$ is less than the slope of the line $y = x$.
The purpose of this problem is to prove this criterion. To this end, try to establish the relation $e_{n+1} = g'(\xi)e_n$ and then deduce the convergence criterion. The following properties are useful

- $x_{n+1} = g(x_n)$ and $r = g(r)$ (root of the equation which is a fixed point)
- $e_i = r - x_i$
- **Mean value theorem:** If $f(x)$ is a continuous function on the closed interval $[a, b]$ and differentiable on the open interval (a, b), there exists a number ξ such that $a < \xi < b$ and

$$f'(\xi) = \frac{f(b) - f(a)}{b - a}$$

Problem 4 *MATLAB: Moler-Morrison algorithm*

This is a typical MATLAB exam problem !
Computing an approximation to $\sqrt{x^2 + y^2}$ does not require square roots. It can be done as follows:

real function $f(x, y)$
integer n; **real** a, b, c, x, y
$f \leftarrow \max\{|x|, |y|\}$
$a \leftarrow \min\{|x|, |y|\}$
for $n = 1 : 3$
  $b \leftarrow (a/f)^2$
  $c \leftarrow b/(4 + b)$
  $f \leftarrow f + 2cf$
  $a \leftarrow ca$
end for
end function f

The output of the algorithm is the value of f. Test the algorithm on some simple cases such as $(x, y) = (3, 4)$, $(-5, 12)$, and $(7, -24)$. Then write a routine that uses the function $f(x, y)$ for approximating the Euclidean norm of the vector $\mathbf{x} = (x_1, \cdots, x_n)$ given as $||\mathbf{x}||_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$.

Problem 5 *MATLAB: Steffensen's method*

Steffensen's method is a scheme for finding a numerical solution of an equation of the form $f(x) = 0$ that is similar to Newton's method but does not require the derivative of $f(x)$. The solution process starts by choosing a point x_0, near the solution, as the first estimate of the solution. The next estimates of the solution x_{n+1} are calculated by:

$$x_{n+1} = x_n - \frac{(f(x_n))^2}{f(x_n + f(x_n)) - f(x_n)}$$

Write a MATLAB user-defined function that solves a nonlinear equation with Steffensen's method. Name the function `Xs = SteffensenRoot(Fun, Xest)`, where the output argument `Xs` is the numerical solution. The input argument `Fun` is a name for the function that calculates $f(x)$ for a given x (Remember to use the handle @ when calling the function which you define in a separate script), and `Xest` is the initial estimate of the solution. The iterations should stop when the estimated relative error given as $\left|\frac{x_{n+1}-x_n}{x_n}\right|$ is smaller than 10^{-6}. The number of iterations should be limited to 100 (to avoid an infinite loop). If a solution with the required accuracy is not obtained in 100 iterations, the program should stop and display an error message. Test your code using the function from Problem 1 with a starting point $x_0 = 1$. Display the output at each iteration in a tabulated format with the iteration number, x_n, $f(x_n)$, and the relative error.

Problem 6 *OPTIONAL but important*

The speed of convergence of a root finding technique is a very important parameter. The different proposed algorithms generate a sequence of values $\{x_n, n = 0, \cdots, \infty\}$ that converges to a root r.
If positive constants α and λ exist with

$$\lim_{n\to\infty} \frac{|x_{n+1} - r|}{|x_n - r|^\alpha} = \lambda$$

then the sequence x_n converges to r of order α with asymptotic error constant λ.

- If $\lambda = 0$, the convergence is superlinear
- If $\lambda = 1$, the convergence is sublinear
- If $\alpha = 1$ and $\lambda < 1$, the convergence is linear
- If $\alpha = 2$ and $\lambda > 0$, the convergence is quadratic ($\alpha = 3$ is cubic, etc.)

Determine the order of convergence of the following sequences for $r = 0$

(a) $(1/3)^n$

(b) $10^{-3\cdot 2^n}$

(c) n^{-10}

(d) 10^{-n^2}