CSC336 Assignment 2 Due Mon Oct 21

Please **underline** your last name. The assignment is due at the beginning of tutorial.
For each programming question, hand in the script file and output.

1. [10] We would like to solve the following system $Ax = b$ for x, where

$$A = \begin{bmatrix} 1 & 1 & -1 & 0 \\ 1 & 1 & 4 & 3 \\ 2 & -1 & 2 & 4 \\ 2 & -1 & 2 & 3 \end{bmatrix}, b = \begin{bmatrix} 1 \\ -2 \\ -2 \\ 1 \end{bmatrix}$$

   First, find the permutation matrix P and the lower and upper triangular matrices L and U, respectively, such that $PA = LU$. Now, use the factoring $PA = LU$ to solve for x. Show all your steps.

2. [10] Consider the $2n \times 2n$ system

$$\begin{bmatrix} A & Z \\ A & B \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},$$

   where A, B are $n \times n$ non-singular matrices, Z is an $n \times n$ matrix with entries all zero, b_1, b_2 are $n \times 1$ vectors, and x_1, x_2 are $n \times 1$ unknown vectors. Describe an efficient method to solve for x_1 and x_2. What is the number of flops used?

3. [10] Assume that an $n \times m$ sparse matrix A is stored in arrays a, ja of size n, and ia of size $n + 1$, according to the Profile storage scheme. Write an efficient pseudo-code that computes the matrix-vector multiplication $p = Av$ using the Profile storage scheme.

4. [10] We would like to solve the following system $Ax = b$ for x, where

$$A = \begin{bmatrix} 1 & 1+\epsilon \\ 1-\epsilon & 1 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

   and $\epsilon = 0.01$. An unknown algorithm that uses high precision arithmetic computes the approximate solution $\hat{x}$. How many digits of accuracy do we need from the residual $r = b - A\hat{x}$ in order to guarantee that the relative error $\|x - \hat{x}\|/\|x\| \leq 10^{-2}$? Use the infinity norm.

5. [10] The function

$$f(x) = 230x^4 + 18x^3 + 9x^2 - 221x - 9$$

   has two real roots, one in $[-1, 0]$ and the other in $[0, 1]$. Use Newton's method, the secant method, and the bisection method to find the roots to within 10^{-3}. Perform at most 5 iterations for each method. Use the midpoint as the initial approximation for Newton's method and use the two end points as the initial approximations for the secant method. Show all the iterative approximations for the three methods. How accurate are the methods?

6. [20] Write an efficient MATLAB program `function B = SolveTri(A)` that takes in an $n \times n$ lower triangular matrix A and returns its inverse B. Show your derivation. Test the program on the matrix whose elements are $a_{ij} = (i+j)^2$ when $i \geq j$. Use $n = 10$. Form the product AA^{-1} as a test of the computed inverse.

7. [20] Implement Newton's method for systems in MATLAB with `function [X, k] = NewtSys(t, maxit, tol)`, where X is the solution vector, k is the number of iterations used, t is the initial guess, $maxit$ is maximum number of iterations, and tol is the tolerance precision.

   You may assume that your MATLAB function can globally access the multi-dimensional functions $f(X)$ and it's Jacobian $J(X)$ when solving the non-linear equation $f(X) = 0$. Furthermore, you should use the infinity norm.

   Let $f(X) = (f_1, f_2, \ldots, f_n)^T$, where $f_i = x_{i-1}^2 + x_{i-1}x_{i+1} + x_{i+1}^2 - 3x_i^2$ and $X = (x_1, x_2, \ldots, x_n)^T$. Find $J(X)$. Use Newton's method to solve this non-linear system with $n = 3, x_0 = 0$, and $x_4 = 1$. Find X and k for inputs $t = [0.3, 0.6, 0.9]^T, maxit = 10$, and $tol = 10^{-5}$.

Total Marks 90