Problem Set 3

1. (10 pts.) Exercise 2.1, problem 2b, 2c

2. (5 pts.) Given that the approximate operation count for Gaussian elimination is $2n^3/3$ where $n$ is the size of the matrix, estimate how long it would take to solve a $4n \times 4n$ linear system if we know that it takes 1 second to solve an $n \times n$ system.

3. (5 pts.) Exercise 2.2, problem 2b

4. (5 pts.) Exercise 2.2, problem 4b

5. (20 pts.)
   
a) Write a program to calculate the $L$ and $U$ in the the $LU$ factorization of an $n \times n$ matrix $A$ (do not include pivoting). Your program should check for nearly zero pivot elements, print an error message, and cease operation if one is encountered. Also your code should implement the algorithm using loops, not MATLAB matrix routines. Run your code on a random matrix generated with $A=\text{rand}(100,100)$. Confirm your factorization of $A$ is correct by multiplying $L$ and $U$ together and comparing with $A$ (hint, use $\text{norm}(A-L*U)$).

   b) Assuming the $L$ and $U$ from above have been computed, write codes (again using loops not MATLAB matrix routines) to perform the back substitutions required to solve $Ax = LUx = b$. Use your codes to solve the system $Ax = b$ using the random matrix $A$ from above and a randomly generated right hand side $b$. Confirm the correctness of the result by comparing $Ax$ with $b$ (hint, use $\text{norm}(A*x-b)$).

   c) Using the codes form parts a and b above, solve a linear system generated by the $\text{rand}(n,n)$ command with $n = 100$. Report the number of seconds the code required to factor the matrix, and to perform the back substitutions (hint use the MATLAB commands $\text{tic}$ and $\text{toc}$). Repeat this procedure for $n = 200$ and $n = 400$. Comment on the increasing cost of the LU factorization and the back substitutions as the size of the matrix increases.