1. Assume that a computer solves a 1000-variable, upper-triangular, linear system by back substitution in 0.5 seconds. Estimate the time needed to solve a general (full) system by Gaussian elimination (LU-factorization). Use the counts from the notes: $2n^3/3 + O(n^2)$ for GE and n^2 for backward substitution.

2. Consider the linear system

$$\begin{pmatrix} 3 & 1 & 2 \\ 6 & 3 & 4 \\ 3 & 1 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 3 \end{pmatrix}.$$

(a) Solve (by hand) the system by finding the LU-factorization and then carrying out the two-step triangular solves. **(b)** Solve the problem numerically using the $PA = LU$ factorization. In MATLAB use `[L, U, p] = lu(A,"vector")`. **(c)** From the vector p returned in part **(b)**, how can the corresponding permutation matrix P be constructed in MATLAB?

3. Consider the linear system

$$\begin{pmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}.$$

(a) Write a function `x = trisolve1(b)` which solves the system $T\mathbf{x} = \mathbf{b}$, where T is the above tridiagonal matrix. Use the MATLAB backslash operator (`x = T\b`). Test your script for $\mathbf{b} = (1, 0, 0, 0, 1)^T$ corresponding to the exact solution $\mathbf{x} = (1, 1, 1, 1, 1)^T$.
(b) Show that an LU decomposition of T consists of two bi-diagonal matrices,

$$L = \begin{pmatrix} 1 & & & & & \\ \ell_2 & 1 & & & & \\ & \ell_3 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & \ell_{n-1} & 1 & \\ & & & & \ell_n & 1 \end{pmatrix}, \qquad U = \begin{pmatrix} u_1 & -1 & & & & \\ & u_2 & -1 & & & \\ & & u_3 & -1 & & \\ & & & \ddots & \ddots & \\ & & & & u_{n-1} & -1 \\ & & & & & u_n \end{pmatrix},$$

where $\ell_i = -(i-1)/i$ and $u_i = (i+1)/i$ and no permutations are required. You may verify this by matrix mutiplication. Design an algorithm which takes advantage of zero elements to solve $LU\mathbf{x} = \mathbf{b}$, implement your algorithm as the function `x = trisolve2(b)`. Test your script on the same example in **(a)**.
(c) Use both `trisolve1` or `trisolve2` to solve for `x` when `b = rand(n,1)` for $n = 200$, 400, 800, 1600, 3200, 6400, 12800, and make a table with the solution times (use `tic/toc`). Also represent the data in your table graphically (one plot). Interpret the results.