1. **Gaussian elimination.** Write a subroutine that finds the solution $x = (x_1, \ldots, x_N)^T$ of the linear set of equations

   $$Ax = b, \tag{1}$$

   for given square matrix $A = (a_{ij})_{N\times N}$ and vector $b = (b_1, \ldots, b_N)^T$ using Gaussian elimination.

   In order to test your subroutine, write a main program that reads from a file the size and elements of the matrix A and the components of vector b, and prints the components of the solution x to the screen. Compare your results to those given by Numerical Recipes subroutines `ludcmp` and `lubksp` which are based on the LU decomposition. Your subroutine should be able to appropriately deal with possible divisions by small values! Three example problems can be found at

   `http://cc.oulu.fi/~tf/tiedostot/pub/atkIV/harjoitukset/Ex008/`

   The files contain the dimension of the matrix, N, the matrix A with one row on each line, followed by vector b in the same form.

2. **Matrix inverse and determinant.** Write a program that finds the inverse A^{-1} and determinant $\det A$ of a square matrix $A = (a_{ij})_{n\times n}$ using LU–decomposition. Numerical Recipes routines `ludcmp` and `lubksb` are at your disposal.