# Lab Session 8

**MA-581 :** Numerical Computations Lab  July-November 2022  S. Bora

1. This is an exercise on handling matrices in MATLAB.

   (a). Generate the following square matrix which is known as the Wilkinson matrix without using any *for loops.*

   $$W_{ij} = \begin{cases} -1 & \text{if } i > j \\ 1 & \text{if } i = j \text{ or } j = n \\ 0 & \text{otherwise} \end{cases}$$

   Here $n$ is the size of the matrix. You may write a function program `W = Wilkinson(n)` which takes the size $n$ of the matrix as input for this.
   Hint: Use the MATLAB commands `eyes`, `ones` and `tril`.

   (b). A real $2n \times 2n$ matrix $H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & -H_{11}^T \end{bmatrix}$ is said to be Hamiltonian if $H_{12}$ and $H_{21}$ are $n \times n$ matrices such that $H_{12}^T = H_{12}$ and $H_{21}^T = H_{21}$. Here $T$ denotes the transpose of a matrix. Use concatenation and the `randn` command to generate a random real Hamiltonian matrix.

2. The solution of a system of equations $Ax = b$ can be obtained in MATLAB by setting `x = A\b`. MATLAB uses GEPP (Gaussian Elimination with Partial Pivoting) to find $x$ for this command. [Wait for a few more classes to know the details!]

   The same may also be found by setting `x = inv(A) * b`. Write an M-file which finds the time taken by both these commands for 20 matrices with sizes increasing from 200 to 1150 in steps of 50 and plots them on a semilog scale on the same graph. Use legends to distinguish between your curves.

3. Write the following function programs to solve triangular systems of equations.

   (a). `x = colbackward(U,b)` to solve an upper triangular system $Ux = b$ by column oriented back substitution.

   (b). `x = rowforward(L,b)` to solve a lower triangular system $Lx = b$ by row oriented forward substitution.

4. Write a MATLAB function program `[L,U] = genp(A)` which finds an $LU$ factorization $A = LU$ of an $n$-by-$n$ matrix $A$ by performing Gaussian Elimination with no pivoting (GENP).