This assignment is worth a total of 15 points. If you have questions or concerns regarding the course content, please email Leroy at *leroy_jia@brown.edu* or visit his office hours at 170 Hope St.

1 Triangular systems

a) Write a MATLAB program that solves a linear system represented by an *upper triangular* matrix (and a vector representing the RHS) via the backsolving technique. Your code should first check if the input is square and upper triangular. Do not use the backslash operator for this problem.

b) Now, write a MATLAB program that solves a linear system represented by a *lower triangular* matrix (all entries above the main diagonal are zero). You may use your code from part a.

2 Toeplitz matrices

A Toeplitz matrix is a matrix with the property that each element on a given diagonal is constant. For example, the 6×6 matrix

$$\begin{bmatrix} 0 & 8 & -1 & \pi & -4 & 3 \\ 12 & 0 & 8 & -1 & \pi & -4 \\ 7 & 12 & 0 & 8 & -1 & \pi \\ -2 & 7 & 12 & 0 & 8 & -1 \\ 5 & -2 & 7 & 12 & 0 & -8 \\ 3 & 5 & -2 & 7 & 12 & 0 \end{bmatrix}$$

is a Toeplitz matrix. Write a MATLAB function `toeplitz(vec)` that constructs the square Toeplitz matrix whose elements on each diagonal are the elements of `vec`, with the first element of `vec` corresponding to the lowest diagonal of the matrix, the second element of `vec` corresponing to the second lowest diagonal, and so on until the last element, which corresponds to the highest diagonal. Note that the length of `vec` must be odd or else the matrix will not be defined.

3 Tridiagonal systems

Use your tridiagonal solver from recitation to solve the system

$$\begin{bmatrix} 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} = \begin{bmatrix} 12 \\ -4 \\ 8 \\ -3 \\ 0 \\ -19 \\ 2 \\ -15 \\ -27 \\ 38 \end{bmatrix}$$

Again, do not use the MATLAB backslash operator. You will need to use your own code for full credit.