**Practical Worksheet 02 – Solving Triangular Systems**

**Exercise 01**

1. Write a Matlab function "is\_upper\_triangular" that takes a matrix ***A*** as an argument and returns 1 if matrix ***A*** is an upper triangular matrix, and 0 otherwise (Matrix ***A*** is said to be upper triangular if and only if all elements below its diagonal are zero). A diagonal matrix is ​​also considered an upper triangular matrix.
2. Write a Matlab function "is\_lower\_triangular" that takes a matrix ***A*** as an argument and returns 1 if matrix ***A*** is a lower triangular matrix, and 0 otherwise (Matrix ***A*** is said to be lower triangular if and only if all elements above its diagonal are zero). A diagonal matrix is ​​also considered a lower triangular matrix.
3. Write a Matlab function "matrix\_type" that takes a matrix ***A*** as an argument and returns one of the following results:
	* 0 if matrix ***A*** is not triangular.
	* 1 if matrix ***A*** is upper triangular.
	* 2 if matrix ***A*** is lower triangular.
	* 3 if matrix ***A*** is a diagonal matrix.

Test the function "matrix\_type" on a non-triangular matrix, an upper triangular matrix, a lower triangular matrix, and a diagonal matrix of your choice and check if it returns the correct result in each situation.

**Hint:** to check that a matrix is ​​diagonal, you have to test if it is upper and lower triangular at the same time.

**Exercise 02**

1. Write a Matlab function "determinant\_triangulaire" that takes a matrix A as an argument and calculates and returns the determinant of A if it is triangular (or diagonal), otherwise the function should display the following message: "Matrix A is not triangular".
	* Test your function on a triangular matrix of your choice, and compare the result obtained by your function to that obtained by the predefined Matlab function **"det"** to check if it returns the right result.
2. Write an "invertible" Matlab function that takes a matrix A as an argument and returns 1 if A is invertible, and 0 otherwise.

**Hint:** use the functions declared in exercise 01.

**Exercise 03**

1. Write a Matlab function “solve\_upper\_triangular” that takes as an argument an upper triangular matrix with coefficients A and a right-hand side vector b and that solves the upper triangular system Ax = b.
2. Write a Matlab function “solve\_lower\_triangular” that takes as an argument a lower triangular matrix with coefficients A and a right-hand side vector b and that solves the lower triangular system Ax = b.
3. Write a Matlab script that allows the user to:
	* Enter n the size of a linear system to be solved.
	* Enter the elements of the matrix A and the right-hand side b.
	* If the matrix A is triangular (or diagonal) and invertible, solve the triangular system Ax = b according to the nature of the matrix A (upper or lower triangular) and display the result. Otherwise, the script must display one of the following messages:
		1. “Matrix A is not triangular”: if the matrix A is not triangular.
		2. “Matrix A is not invertible”: if matrix A is triangular but not invertible.
		3. Test the script on a matrix A and a vector b of your choice, and compare the result to that obtained by the Matlab instruction “x = A\b” to verify that your code is correct.

**Hint:** use the functions defined in exercises 01 and 02.