

# Numerical Analysis and Optimization Homework Project 1

---

This assignment is done using the Julia programming language. To ease the task, we load Julia's linear algebra standard library:

```
1 using LinearAlgebra
```

## Problem 1

---

Here we define the function `lufact` that takes as input a square matrix and compute the non-pivoted LU factorization and its grow factor:

`lufact` (generic function with 1 method)

```
1 function lufact(A::AbstractMatrix)
2     # checks that the given matrix is squared
3     if size(A, 1) != size(A, 2)
4         throw(ArgumentError("The matrix is not squared."))
5     end
6     U = copy(A)
7     L = UnitLowerTriangular(copy(A))
8     γ = 0
9     # algorithm
10 end
```