# System of linear equations

Numerical methods used to solve such problem allow to introduce and experiment on Time_complexity, considering cubic time behavior of standard algorithms and *i.e.* quadratic time solutions using LU decomposition.

## Theory

- System_of_linear_equations
- Gaussian_elimination, Gauss and Gauss-Jordan eliminations (diagonalization, triangularization)
- Pivot_element, pivoting
- LU_decomposition
  - Triangular_matrix#Forward_and_back_substitution
- Chapter 2 in the book “Numerical Recipes” :
  - 2.0 Introduction
  - 2.1 Gauss-Jordan Elimination
  - 2.2 Gaussian Elimination with Backsubstitution
  - 2.3 LU Decomposition and Its Application
- Python NumPy library : NumPy Reference
  - Linear algebra (numpy.linalg) : numpy.linalg.solve
- Time complexity analysis
  - Hint : in Python, use the timeit module

## Exercices and applications

- Exercices :
  - write a python function for diagonalisation with partial pivoting
  - random numbers → linear systems
  - comparison with numpy standard library
  - measurements of execution time to check cubic complexity

### 1D problems with neigbours

- Thermal diffusion and chemical diffusion (transient or stationary) on a regular 1D space with equidistant steps. ODE equations can be writen such a given evolution equation for node # i only imlies nodes i+1 and i-1
- Using tridiagonal Thomas algorithm allows to save computational time thanks to n complexity
- ? Python library with Thomas algorithm

# What you must have learned in this chapter

- Except ill-conditionned, linear systems can be solved “exactly” using linear algebra algorithms in a finite and known number of arithmetic operations.
- The accuracy is determined by the number of numerical figures which are encoded in floating point description
- For a general system of n equations, diagonalisation requires of the order of $n^3$ operations. Also for solving a system using these method.
- If the coefficient matrix is the same for different systems (only the independent coefficients are different), it is possible to solve systems with the order of $n^2$ operations, if the matrix of coeeficients is decomposed in the product of two triangular matrix (Lower-Upper decomposition). This $n^3$ step is realised only once.

# References :


- Numerical recipes, The Art of Scientific Computing 3rd Edition, William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, 2007, isbn: 9780521880688
  - http://numerical.recipes/
    - in C : http://apps.nrbook.com/c/index.html
  - http://www2.units.it/ipl/students_area/imm2/files/Numerical_Recipes.pdf
  - http://apps.nrbook.com/empanel/index.html#