0.1 Résolution de systèmes linéaires par des Méthodes Directes

Dans le cadre de ce premier TP, nous devions implémenter l'algorithme du *Pivot de Gauss* en utilisant le langage de programmation C.

0.1.1 Détail de l'algorithme

Soit A une matrice $\in \mathcal{M}_{m,m}$ $m \in \mathbb{N}$ et $b \in \mathcal{M}_{m,1}$.
L'algorithme de Gauss se décrit ainsi:

Pour $k = 1, \ldots, n-1$ Faire :
    Pour $i = k+1, \ldots, n$ Faire :

$$\alpha_i^{(k)} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

        Pour $j = k, \ldots, n$ Faire :

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \alpha_i^{(k)} a_{kj}^{(k)}$$

        FIN Pour j

$$b_i^{(k+1)} = b_i^{(k)} - \alpha_i^{(k)} b_k^{(k)}$$

    FIN Pour i
FIN Pour k

Après cette algorithme permettant l'échelonnage de la matrice, pour trouver les solutions du système, on appliquera la formule suivante:

$$x_n = \frac{b_n}{a_{m,m}}$$

et

$$\forall i = n-1, \ldots, 1,\ x_i = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1+i}^{n} a_{ij} x_j \right)$$

La complexité temporelle de cet algorithme est cubique soit $O(n^3)$ avec une compléxité exacte de $\frac{2n^3}{3}$.
Pour l'implémentation de cet algorithme, nous allons présenter deux façons de le conceptualiser avec une comparaison algorithmique des deux programmes.

0.1.2 Implémentation du pivot de Gauss sans matrice Augmentée

0.1.3 Implementation du pivot de Gauss avec matrice Augmentée