Rapport du Solveur de Systèmes Linéaires

# Code :

import numpy as np  
import sys  
  
def input\_coefficients(matrixSize):  
 matrixInput = np.zeros((matrixSize, matrixSize + 1))  
 for i in range(matrixSize):  
 for j in range(matrixSize + 1):  
 matrixInput[i][j] = float(input("a[" + str(i) + "][" + str(j) + "]="))  
 return matrixInput  
  
def gaussian\_elimination(matrixInput, matrixSize):  
 for i in range(matrixSize):  
 if matrixInput[i][i] == 0.0:  
 sys.exit("Division par zéro détectée !")  
  
 for j in range(i + 1, matrixSize):  
 ratio = matrixInput[j][i] / matrixInput[i][i]  
  
 for k in range(matrixSize + 1):  
 matrixInput[j][k] = matrixInput[j][k] - ratio \* matrixInput[i][k]  
  
 return matrixInput  
  
def back\_substitution(matrixInput, matrixSize):  
 solution\_vector = np.zeros(matrixSize)  
 solution\_vector[matrixSize - 1] = (  
 matrixInput[matrixSize - 1][matrixSize]  
 / matrixInput[matrixSize - 1][matrixSize - 1]  
 )  
  
 for i in range(matrixSize - 2, -1, -1):  
 solution\_vector[i] = matrixInput[i][matrixSize]  
  
 for j in range(i + 1, matrixSize):  
 solution\_vector[i] = int(  
 solution\_vector[i] - matrixInput[i][j] \* solution\_vector[j]  
 )  
  
 solution\_vector[i] = solution\_vector[i] / matrixInput[i][i]  
  
 return solution\_vector  
  
def display\_solution(solution\_vector, matrixSize):  
 print("\nSolution requise : ")  
 for i in range(matrixSize):  
 print("X%d = %0.2f" % (i, solution\_vector[i]), end="\t")  
  
def solve\_linear\_system():  
 # Obtenir la taille de la matrice de l'utilisateur  
 matrixSize = int(input("Entrez la taille de la matrice : "))  
  
 # Entrer les coefficients de l'utilisateur  
 matrixInput = input\_coefficients(matrixSize)  
  
 # Appliquer l'élimination gaussienne  
 matrixInput = gaussian\_elimination(matrixInput, matrixSize)  
  
 # Effectuer la substitution arrière  
 solution\_vector = back\_substitution(matrixInput, matrixSize)  
  
 # Afficher la solution  
 display\_solution(solution\_vector, matrixSize)  
  
# Appel de la fonction principale pour résoudre le système linéaire  
solve\_linear\_system()

# Rapport :

Rapport de Code :  
1. Objectif et Aperçu :  
 - Le code résout un système d'équations linéaires en utilisant l'élimination gaussienne.  
 - Les étapes comprennent la saisie des coefficients, l'élimination gaussienne, la substitution arrière et l'affichage de la solution.  
  
2. Coefficients d'Entrée :  
 - La fonction input\_coefficients initialise et remplit une matrice avec les coefficients saisis par l'utilisateur.  
  
3. Élimination Gaussienne :  
 - La fonction gaussian\_elimination transforme la matrice en une forme triangulaire supérieure en utilisant l'élimination gaussienne.  
  
4. Substitution Arrière :  
 - La fonction back\_substitution applique la substitution arrière pour trouver le vecteur solution.  
  
5. Afficher la Solution :  
 - La fonction display\_solution imprime le vecteur solution obtenu.  
  
6. Fonction Principale - solve\_linear\_system :  
 - La fonction principale demande à l'utilisateur la taille de la matrice, saisit les coefficients, applique l'élimination gaussienne, la substitution arrière, puis affiche la solution.  
  
7. Améliorations et Suggestions :  
 - Gestion des erreurs pour la division par zéro.  
 - Commentaires dans le code pour améliorer la lisibilité.  
 - Validation des entrées pour s'assurer qu'elles sont numériques.  
 - Inclusion de cas de test pour la vérification.  
  
8. Conclusion :  
 - Le code implémente efficacement l'élimination gaussienne pour résoudre des systèmes linéaires.