Calcul Numeric – Laborator #3

Ex. 1 Să se construiască în Python procedura **SubsDesc** conform sintaxei x=**SubsDesc**(A, b) care rezolvă numeric sisteme liniare superior triunghiulare conform algoritmului (metoda substituţiei descendente). Procedura va testa dacă matricea este pătratică, superior triunghiulară şi dacă sistemul este compatibil determinat.

Ex. 2 Să se construiască în Python procedura **SubsAsc** conform sintaxei x=**SubsAsc**(A, b) care rezolvă numeric sisteme liniare inferior triunghiulare conform algoritmului (metoda substituţiei ascendente). Procedura va testa dacă matricea este pătratică, inferior triunghiulară şi dacă sistemul este compatibil determinat.

Ex. 3

a. Să se construiască în Python procedurile **GaussFaraPiv** şi **GaussPivPart** cu sintaxa

GaussFaraPiv(A, b, tol)

GaussPivPart(A, b, tol)

care returneaza soluţia sistemului $Ax = b$ conform metodelor de eliminare Gauss fără pivotare şi Gauss cu pivotare parţială.

Obs.: Variabila *tol* se folosește în algoritmi în cazul în care dorim să verificăm dacă un număr este nenul. Condiția $a \neq 0$ va fi înlocuită cu $|a| > tol$, cu *tol* foarte mic.

b. Să se rezolve sistemele de mai jos, apelând funcţiile create la subpunctul a. Se alege $tol = 10^{-16}$.

$$\begin{cases} x_2 + x_3 = 3 \\ 2x_1 + x_2 + 5x_3 = 5 \\ 4x_1 + 2x_2 + x_3 = 1 \end{cases} \qquad \begin{cases} x_2 - 2x_3 = 4 \\ x_1 - x_2 + x_3 = 6 \\ x_1 - x_3 = 2 \end{cases} \tag{1}$$

c. Să se aplice metodele Gauss fără pivotare şi cu pivotare parţială pentru rezolvarea sistemului

$$\begin{cases} \varepsilon\, x_1 + x_2 = 1 \\ \quad x_1 + x_2 = 2 \end{cases} \tag{2}$$

unde $\varepsilon = \mathsf{O}(10^{-20}) \ll 1$ şi $tol = 0$. Modificaţi $tol = 10^{-16}$. Ce observaţi în cazul metodei Gauss fără pivotare?

Ex. 4 Să se afle manual rangul matricei A folosind algoritmul de determinare al rangului cu GPP

$$A = \begin{pmatrix} 2 & -2 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ -4 & 4 & 0 & 0 \\ 4 & -1 & 10^{-20} & 10^{-20} \end{pmatrix}$$

Se vor folosi succesiv valorile $tol = 0, 10^{-20}, 10^{-10}$. Ce observaţi?

Ex. 5 Să se construiască în Python procedura **Rang**(A, tol) conform algoritmului de calcul al rangului unei matrice folosind metoda GPP. Procedura **Rang** returnează rangul matricei A. Să studieze natura următoarelor trei sisteme, afişându-se unul dintre mesajele:

- Sistem compatibil determinat;

- Sistem compatibil nedeterminat;
- Sistem incompatibil.

$$\begin{cases} x_2 - 2x_3 = 4 \\ x_1 - x_2 + x_3 = 6 \\ x_1 - x_3 = 2 \end{cases} \qquad \begin{cases} x_1 - 2x_2 + 3x_3 = 4 \\ -2x_1 + 3x_2 - 4x_3 = -5 \\ 3x_1 - 4x_2 + 5x_3 = 6 \end{cases} \qquad \begin{cases} x_2 + x_3 = 3 \\ 2x_1 + x_2 + 5x_3 = 5 \\ 4x_1 + 2x_2 + x_3 = 1 \end{cases}$$

Se va considera $tol = 10^{-10}$.

Ex. 6 Să se afle rangul următoarelor matrice folosind procedura **Rang**.

$$A = \begin{pmatrix} 0 & -3 & -6 & 4 \\ -1 & -2 & -1 & 3 \\ -2 & -3 & 0 & 3 \\ 1 & 4 & 5 & -9 \end{pmatrix}, \quad A = \begin{pmatrix} 2 & -2 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ -4 & 4 & 0 & 0 \\ 4 & -1 & 10^{-20} & 10^{-20} \end{pmatrix}$$

Se vor folosi succesiv valorile $tol = 0, 10^{-20}, 10^{-10}$. Ce observaţi?

Ex. 7 Se va repeta Ex. 6 folosind funcția *linalg.matrix_rank(A, tol= None)* din modulul Numpy.