Calcul Numeric
Test Laborator – Calculatoare şi Tehnologia Informaţiei, Anul I

INSTRUCŢIUNI:

1. Toate problemele sunt **obligatorii**.
2. Comentaţi şi explicaţi toate rezolvările trimise. Codurile necomentate/neexplicate nu se punctează.
3. **TIMP DE LUCRU: 2 ore**
4. Rezolvările problemelor corespunzătoare acestui test vor fi trimise prin email:
   - **ca fişier .txt**, cu denumirea `Nume_Prenume_Grupa_Test.txt`
   - la adresa `alexandru.ghita@unibuc.ro`;
   - vor avea următoarea **linie de subiect**:
     `Test Laborator CN - Nume şi prenume student, Grupa 16X`
5. **Termenul limită** de trimitere prin email a rezolvărilor problemelor: **8 iunie 2021, orele 20:30.**

Algorithm 1: Interpolare Lagrange (Metoda Neville)

Input: $\mathbf{X} \in \mathbb{R}^{n+1}$, $\mathbf{Y} \in \mathbb{R}^{n+1}$, $\mathbf{z} \in \mathbb{R}$
Result: $\mathbf{t} \in \mathbb{R}$

Pasul 1: Construieşte matricea $\mathrm{Q} = (q_{ij})_{i,j=\overline{1,n+1}} \in \mathcal{M}_{n+1}(\mathbb{R})$:

- Se iniţializează prima coloană a matricei Q cu Y;
- Pentru $i = \overline{2, n+1}$ şi $j = \overline{2, i}$ calculează termenii matricei Q:

$$q_{ij} \longleftarrow \frac{q_{i,j-1}(z - x_{i-j+1}) - q_{i-1,j-1}(z - x_i)}{x_i - x_{i-j+1}};$$

Pasul 2: $t \longleftarrow q_{n+1,n+1}$

Pasul 3: OUTPUT($\mathbf{t}$)
STOP.

Factorizarea QR:

Fie $A \in \mathbb{R}^{n\times n}$. Numim descompunere QR a matricei A, descompunerea de forma $A = QR$, unde $Q \in \mathbb{R}^{n\times n}$ este o matrice ortogonală, i.e. $Q^TQ = QQ^T = I$, iar $R \in \mathbb{R}^{n\times n}$ este o matrice superior triunghiulară.

Dacă $A \in \mathbb{R}^{n\times n}$ este o matrice inversabilă, atunci există şi este unică descompunerea QR a matricei A cu $Q \in \mathbb{R}^{n\times n}$ o matrice ortogonală şi $R \in \mathbb{R}^{n\times n}$ o matrice superior triunghiulară având componentele pe diagonala principală $r_{kk} > 0, k = \overline{1,n}$.

Sistemul $Ax = b$ devine $QRx = b$. Cum Q este ortogonală ($Q^TQ = I$), înmulţind relaţia $QRx = b$ cu Q^T obţinem $Rx = Q^Tb$. Cum R este superior triunghiulară sistemul $Rx = Q^Tb$ se rezolvă conform metodei substituţiei descendente.

Algorithm 2: Metoda de descompunere QR

Input: $\mathbf{A} \in \mathbb{R}^{n\times n}$
Result: $\mathbf{Q} \in \mathbb{R}^{n\times n}$, $\mathbf{R} \in \mathbb{R}^{n\times n}$

Pasul 1: Determină prima coloană a matricei Q şi prima linie a matricei R:

- $r_{11} \longleftarrow \sqrt{\sum_{i=1}^{n} a_{i1}^2}$;
- $q_{i1} \longleftarrow \dfrac{a_{i1}}{r_{11}}, \quad \forall\, i = \overline{1,n}$;
- $r_{1j} \longleftarrow \sum_{s=1}^{n} q_{s1}a_{sj}, \quad \forall\, j = \overline{2,n}$;

Pasul 2: Pentru $k = \overline{2,n}$ completează coloana k a matricei Q şi linia k a matricei R:

- $r_{kk} \longleftarrow \sqrt{\sum_{i=1}^{n} a_{ik}^2 - \sum_{s=1}^{k-1} r_{sk}^2}$;
- $q_{ik} \longleftarrow \dfrac{1}{r_{kk}}\left(a_{ik} - \sum_{s=1}^{k-1} q_{is}r_{sk}\right), \quad \forall\, i = \overline{1,n}$;
- $r_{kj} \longleftarrow \sum_{s=1}^{n} q_{sk}a_{sj}, \quad \forall\, j = \overline{k+1,n}$;

Pasul 3: OUTPUT($\mathbf{Q}$, $\mathbf{R}$)
STOP.

Ex. 1

(a) Rezolvaţi numeric ecuaţia $x^2 - 17 = 0$. Folosind comentarii se va argumenta aplicabilitatea metodei alese.

(b) Să se ilustreze grafic funcţia şi punctul/punctele de intersecţie cu axa OX. Graficul va include minim notarea axelor OX şi OY şi legenda.

Ex. 2

Presupunem că avem datele cunoscute **X** (date de client) în punctele obţinute din discretizarea intervalului $[-1, 1]$ în 23 puncte echidistante. Valorile corespunzătoare punctelor rezultate **Y** sunt obţinute prin evaluarea funcţiei $f(x) = e^{2x}$ în acele puncte.

(a) Implementează în **python** *metoda Neville de interpolare Lagrange* cu numele **interp_neville** care determină, conform metodei Neville, polinomul Lagrange $P_n(x)$. Pentru implementare, urmăreşte algoritmul de mai sus;

(b) Clientul doreşte aproximarea valorilor funcţiei în toate punctele din discretizarea cu 75 de puncte echidistante a domeniului. Pentru aproximarea valorilor lipsă, foloseşte datele oferite de client şi *metoda Neville de interpolare Lagrange*. Într-o figură, afişează datele clientului, graficul funcţiei exacte cât şi graficul aproximării obţinute;

(c) Într-o figură nouă, generează graficul erorii de interpolare $e_t = |P_n(x) - f(x)|$.

Ex. 3

(a) Implementează in **python** factorizarea QR cu numele **fact_qr_new**. Pentru implementare, urmăreşte algoritmul de mai sus. În implementarea metodei **fact_qr_new**, verifică dacă:

   (i) Matricea A este pătratică;

   (ii) Matricea A este inversabilă.

(b) Pentru implementare, verifică rezolvarea sistemului $A \cdot \underline{\boldsymbol{x}} = \underline{b}$, unde:

$$A = \begin{bmatrix} 0 & 1 & 5 & 1 \\ -10 & 4 & 0 & 6 \\ 4 & -5 & -10 & -5 \\ 1 & 5 & -7 & 7 \end{bmatrix}, \quad \underline{b} = \begin{bmatrix} 28 \\ 22 \\ -72 \\ 24 \end{bmatrix}. \tag{1}$$

Help: Ţine cont că rezolvarea sistemului $A \cdot \underline{\boldsymbol{x}} = \underline{b}$ atunci când $A = QR$ folosind algoritmul de mai sus se reduce la a rezolva sistemul $R \cdot \underline{\boldsymbol{x}} = Q^T \cdot \underline{b}$.