Serie 6.

Woche: **26.10 – 29.10.2026**

Aufgabe 1 (Vektoren | 1 Punkt).

Löse die Aufgaben ohne `for`- oder `while`-Schleifen.

(a) Sei

$$\mathbf{x} = \begin{pmatrix} -1 \\ 2 \\ -3 \\ 4 \end{pmatrix} \quad \text{und} \quad \mathbf{y} = \begin{pmatrix} 1 & 2 & 3 & 4 \end{pmatrix}.$$

Berechne $\mathbf{x} + 3\mathbf{y}^\top$ und $\mathbf{x} + 3\mathbf{y}^\top + 2\mathbf{w} - \mathbf{z}$, wobei $w_i = \cos(x_i)$ und $z_i = x_i^2$ für $i = 1, \ldots, 4$.

(b) Erzeuge einen Vektor $\mathbf{x} = (2, 2.01, 2.02, \ldots, 2.99, 3)$ und berechne $\mathbf{y} = (y_1, \ldots, y_{101})$ mit

$$y_i = \frac{\pi^{3x_i}}{\cos(x_i)x_i}, \quad i = 1, \ldots, 101.$$

Aufgabe 2 (Matrizen | 1 Punkt).

Löse die Aufgaben ohne `for`- oder `while`-Schleifen.

(a) Für

$$\mathbf{A} = \begin{pmatrix} -1 & 0 & 1 \\ 0 & 1 & -1 \\ 1 & -1 & 0 \end{pmatrix} \quad \text{und} \quad \mathbf{M} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 6 & 5 \\ 7 & 8 & 9 \end{pmatrix}$$

berechne $(\mathbf{A} + \mathbf{M})^2 - \mathbf{A}^2 - \mathbf{M}^2 - 2\mathbf{A}\mathbf{M}$.

(b) Erzeuge die Matrix

$$\mathbf{Z} = \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array}\right) \in \mathbb{R}^{8\times 8}$$

mit $\mathbf{A}$ wie oben, $\mathbf{B}$ die (3×5)-Nullmatrix (besteht aus lauter Nullen), $\mathbf{C}$ die (5×3)-Einsmatrix und $\mathbf{D}$ die (5×5)-Einheitsmatrix und berechne ihre Spur (Summe der Hauptdiagonaleinträge). Benutze dazu die Befehle `zeros`, `ones`, `eye` und `trace`.

(c) Bringe das folgende lineare Gleichungssystem zunächst in die Form $\mathbf{A}\mathbf{v} = \mathbf{b}$ mit $\mathbf{v} = (w, x, y, z)^\top$ und löse es anschliessend mit dem \-Operator von Matlab (s.a. `doc mldivide`):

$$\begin{array}{rcrcrcrcr} & & 2x & + & 4y & - & 3z & = & -23 \\ 3w & & & + & 4y & & & = & 10 \\ w & + & x & - & 2y & + & 3z & = & 27 \\ -4w & - & 3x & + & 5y & + & 6z & = & 51 \end{array}$$

Aufgabe 3 (MV- und MM-Produkte | 2 Punkte).

Das Matrix-Vektor-Produkt $\mathbf{Ax} = \mathbf{y} \in \mathbb{R}^m$ einer Matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ und eines Vektors $\mathbf{x} \in \mathbb{R}^n$ ist gegeben durch

$$y_i = \sum_{j=1}^{n} a_{ij} x_j.$$

Das Matrix-Matrix-Produkt (d.h. die Matrixmultiplikation) ist wie folgt definiert: Sei $\mathbf{A} \in \mathbb{R}^{m\times n}$ mit Einträgen a_{ij} und $\mathbf{B} \in \mathbb{R}^{n\times \ell}$ mit Einträgen b_{jk}. Dann ist $\mathbf{C} = \mathbf{AB}$ in $\mathbb{R}^{m\times \ell}$ und die Einträge c_{ik} berechnen sich durch

$$c_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk}.$$

Schreibe die Produkte als Matlab-Funktionen. Teste die Funktionen anhand der folgenden Beispiele:

$$\mathbf{A} = \begin{pmatrix} 9 & 8 & -1 & 4 \\ 7 & 10 & 6 & -8 \\ 5 & -4 & -3 & 2 \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} 1 & -1 \\ -9 & 3 \\ 5 & 8 \\ 1 & 3 \end{pmatrix} \quad \mathbf{x} = (4, -3, 5, 1)^\top.$$

Dabei sollen die Funktionen so geschrieben sein, dass sie für allgemeine Matrizen bzw. Vektoren gültig sind. Dazu sind die Befehle `length` und `size` nützlich.

Aufgabe 4 (Matrixmanipulation | 2 Punkte).

Erzeuge eine 100×100-Matrix $\mathbf{A}$, deren Einträge nur aus Nullen bestehen, und führe nacheinander die folgenden Operationen aus (du kannst im Workspace-Fenster auf die Matrix doppelklicken, um sie dir anzeigen zu lassen und um einfach zu überprüfen, ob du die Operation richtig ausgeführt hast). Löse die Aufgaben ohne `for`- oder `while`-Schleifen.

1. Setze die Einträge $a_{4,43}$ und $a_{91,100}$ auf 33.
2. Setze alle Einträge der 90. Zeile von $\mathbf{A}$ auf 44.
3. Setze jeden vierten Eintrag der 31. Spalte auf 55.
4. Weise den Einträgen der 88. Zeile absteigend die Werte 199 bis 100 zu.
5. Setze jeden Eintrag $a_{i,j}$, wobei i und j beides Primzahlen sind, auf 13.
6. Setze jeden Eintrag $a_{i,j}$, wobei i und j beide durch 5 teilbar sind, auf 66.
7. Weise jedem Eintrag $a_{i,66}$, wobei i durch 3 und/oder durch 7 teilbar ist, aufsteigend die Werte 1 bis 43 zu.
8. Addiere zu jedem zweiten Eintrag auf der Hauptdiagonalen 21 hinzu.

Hinweis: Zur Lösung dieser Aufgabe sind der Junktor `|` und die Befehle `isprime`, `mod`, `eye` hilfreich. Hast du alles richtig gemacht, sollte `sum(sum(A))` die Zahl 55770 ergeben.