28	29	30	Σ

Name: ____________________

Name: ____________________

Numerik I – 10. Übungsblatt

Hinweis: Verwenden Sie als Dateiname für die schriftliche Abgabe bitte

`Name1_Name2_Blatt10.pdf` bzw. `Name1_Blatt10.pdf`,

wobei Name1 (und Name2) Ihre Nachnamen sind. Dies erleichtert die Zuordnung Ihrer Abgaben.

Aufgabe 28:

Skizzieren Sie in Pseudo-Code eine Funktion `LGSsolve`, welche als Eingabe eine symmetrisch positiv definite Matrix $A \in \mathbb{R}^{n\times n}$ und eine rechte Seite $b \in \mathbb{R}^n$ erhält. Die Funktion soll die Lösung $x \in \mathbb{R}^n$ des linearen Gleichungssystems

$$Ax = b$$

zurückgegeben. Ihre Funktion soll dabei die folgenden drei, als gegeben angenommen, anderen Funktionen aufrufen:

(i) Eine Funktion `[ L, D ] = Cholzerlegung(A)`, die zur Eingabe A eine linke untere Dreiecksmatrix L und eine positiv definite Diagonalmatrix D mit $A = LDL^T$ gemäß Satz 17.2 zurückgibt.

(ii) Eine Funktion `x = Vorwaertssub(L, c)`, die als Eingabe eine untere Dreiecksmatrix $L \in \mathbb{R}^{n\times n}$ und eine rechte Seite $c \in \mathbb{R}^n$ erhält und die Lösung $x \in \mathbb{R}^n$ von $Lx = c$ zurückgibt.

(iii) Eine Funktion `x = Rueckwaertssub(R, c)`, die als Eingabe eine obere Dreiecksmatrix $R \in \mathbb{R}^{n\times n}$ und eine rechte Seite $c \in \mathbb{R}^n$ erhält und die Lösung $x \in \mathbb{R}^n$ von $Rx = c$ zurückgibt.

Wichtig:
Ihr Algorithmus soll effizient arbeiten, also *nicht* das Matrix-Matrix Produkt LD oder DL^T berechnen.
Kommentieren Sie Ihren Pseudocode.
Sprachspezifische Befehle, wie `np.beliebige_methode` gehören nicht in Pseudocode.

Aufgabe 29:

(a) Bestimmen Sie $\|A\|_1$, $\|A\|_2$ und $\|A\|_\infty$ für

$$A = \begin{bmatrix} -1 & 0 \\ 2 & -3 \\ -3 & -2 \end{bmatrix}.$$

(b) Sei $A \in \mathbb{R}^{m\times n}$. Zeigen Sie

$$\|A\|_1 = \|A^T\|_\infty.$$

(c) Zeigen Sie: Für $A \in \mathbb{C}^{m\times n}$ gilt

$$\|A\|_\infty = \max_{i=1,\dots,m} \sum_{j=1}^{n} |a_{ij}|.$$

(d) Sei nun $A \in \mathbb{R}^{n\times n}$ symmetrisch. Zeigen Sie

$$\|A\|_2 \le \|A\|_1.$$

Aufgabe 30:

(a) Sei $f : (\mathbb{R}, |\cdot|) \to (\mathbb{R}, |\cdot|)$ ein Problem mit $f \in C^1(\mathbb{R})$. Zeigen Sie, dass

$$\kappa_{\text{abs}}(f, x) = |f'(x)|.$$

(b) Sei $A \in \mathbb{R}^{n \times n}$ invertierbar. Zeigen Sie

(i) $\operatorname{cond}(A) \geq 1$

(ii) $\operatorname{cond}(\alpha A) = \operatorname{cond}(A)$ für alle $\alpha \in \mathbb{C}_{\neq 0}$.

Programmieraufgabe 10:

Es sollen Verfahren zur Auswertung von Interpolationspolynomen hinsichtlich ihrer Genauigkeit verglichen werden.

(a) Implementieren Sie eine Funktion `vandermonde_interpolate(x, y, X)`, die zu vorgebenen Daten das Interpolationspolynom in der Monombasis berechnet, was man laut Vorlesung nicht machen sollte, und an der Stelle `X` auswertet. `x` ist hierbei ein Vektor/numpy-array mit den Stützstellen x_i, und der Vektor/numpy-array `y` enthält die zu interpolierenden Werte.

Lösen Sie dazu das Gleichungssystem

$$Vc = y \quad \text{mit} \quad v_{ij} = x_i^j \quad i, j = 0 \ldots, n-1$$

für den Koeffizientenvektor c und werten Sie da Polynom $p(x) = \sum_{j=0}^{n} c_j x^j$ an der Stelle `X` aus.

(b) Ein entsprechendes Verfahren zur Newtoninterpolation ist in der Datei `p10.py` bereits implementiert. `newton_interpolate(x, y, X)`

(c) Das baryzentrische Interpolationspolynom, falls Sie es nicht selbst implementiert haben, können Sie durch `from scipy.interpolate import barycentric_interpolate` laden.

(d) Um die Geauigkeit der einzelnen Verfahren zu testen verwenden wir `sympy`, das es erlaubt, das Ergebnis mit nahezu beliebig vielen Stellen zu berechnen. Für das Newtoninterpolationspolynom und die Auswertung mit dem Hornerschema finden Sie in der Datei `p10.py` ein Beispiel. Gehen Sie analog für die Vandermonde Interpolation und die baryzentrischen Interpolation vor. Erhöhen Sie die Anzahl der Stützstellen und beobachten Sie wie sich der Fehler ändert.

Abgabe der Übungsaufgaben am Mittwoch, 14. Juni bis 10:30 Uhr per ILIAS unter „Übungen zu Numerik I".
Abgabe der Programmieraufgaben am Mittwoch, 14. Juni bis 10:30 Uhr per ILIAS unter „Programmierübungen zu Numerik I".