Zürcher Hochschule für Angewandte Wissenschaften

Name	
Vorname	
Klasse	

Aufgabe	**1**	**2**	**3**	**4**	**5**	**6**	**Total**
Maximum	10	10	10	10	10	10	60
Erreichte Punkte							

Note	

- **Dauer:** 120 Minuten
- **Hilfsmittel:** Gemäss Kursvereinbarung
- **Lösungsweg:** Der Lösungsweg muss vollständig (d.h. inklusive relevanter Zwischenschritte) angegeben und nachvollziehbar sein. Resultate ohne Zwischenschritte geben keine Punkte. Der Python-Code muss als *.py vorliegen, vollständig und lauffähig sein, das Resultat muss eindeutig erkennbar und gekennzeichnet sein.
- **Bewertung:** Es hat insgesamt 6 Aufgaben. Jede Aufgabe wird mit 10 Punkten gleich bewertet.
- **Abgabe:** Sämtliche von Ihnen beschriebene Lösungs-Blätter müssen mit Namen angeschrieben sein und in diesen Prüfungsbogen gelegt werden. Den verlangten Python-Code (inkl. der aufgerufenen Unterfunktionen) geben Sie auf Moodle im Verzeichnis SEP -> Abgabe SEP als *.zip Datei ab.

Aufgabe 1

Sie arbeiten auf einem Rechner mit 5-stelliger Gleitpunktarithmetik im Dualsystem. Für den Exponenten haben Sie zusätzlich zum Vorzeichen 3 Stellen zur Verfügung. Geben Sie sämtliche Resultate im Dezimalsystem an.

a) (2 Punkte) Was ist der grösste Exponent, den Sie speichern können?

b) (4 Punkte) Welches ist die kleinste darstellbare positive Zahl, welche die grösste? Approximieren Sie diese durch normierte 4-stellige Maschinenzahlen im Dezimalsystem.

c) (4 Punkte) Vergleichen Sie nun Ihren Rechner mit dem Ihres Kollegen, der mit einem 2-stelligen Hexadezimalsystem arbeitet. Wer rechnet genauer?

Aufgabe 2

Gegeben ist die Funktion

$$f(x) = x^2 \cdot \sin(x)$$

mit iherer Ableitung

$$f'(x) = 2x\sin(x) + x^2\cos(x)$$

und $x \in \mathbb{R}$ im Bogenmass.

a) (1 Punkt) Bestimmen Sie die Konditionszahl von $f(x)$ in Abhängigkeit von x.

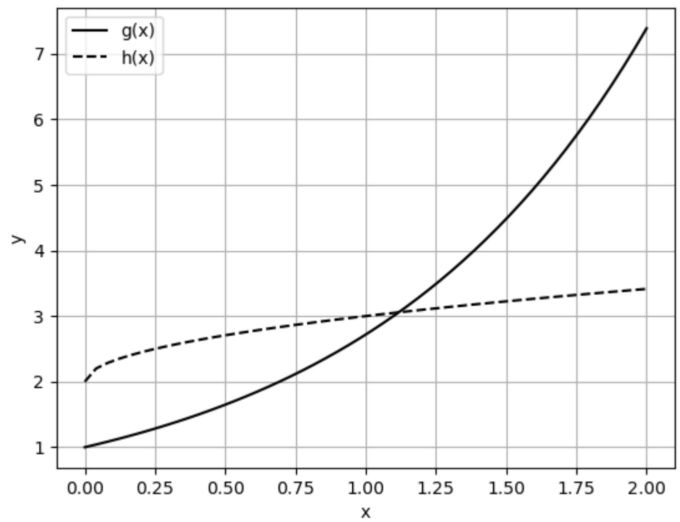
b) (3 Punkte) Berechnen Sie näherungsweise, mit welchem absoluten Fehler $x_0 = \pi/3$ höchstens behaftet sein darf, damit der relative Fehler von $f(x_0)$ höchstens 10% beträgt.

c) (3 Punkte) Bestimmen Sie numerisch das Verhalten der Konditionszahl von $f(x)$ für $x \to 0$ und geben Sie an, was das Ergebnis über die Konditionierung von $f(x)$ für $x = 0$ aussagt.

d) (3 Punkte) Plotten Sie die Konditionszahl von $f(x)$ halblogarithmisch im Bereich $x \in [-2\pi, 3\pi]$ und geben Sie an, was das Ergebnis über die Konditionierung von $f(x)$ für $x \in \mathbb{R}$ aussagt.

Aufgabe 3

Gesucht ist der Schnittpunkt der Funktionen $g(x) = \exp(x)$ und $h(x) = \sqrt{x} + 2$.



a) (5 Punkte) Bestimmen Sie mit dem Newton-Verfahren und dem Startwert $x_0 = 0.5$ den Schnittpunkt bis auf einen absoluten Fehler von höchstens 10^{-7} genau.

b) (5 Punkte) Zeigen Sie, dass der Fixpunkt der Iteration

$$x_{k+1} = \ln(\sqrt{x_k} + 2)$$

gerade dem Schnittpunkt von $g(x)$ und $h(x)$ entspricht, und dass die Iteration für jeden Startwert x_0 im Intervall $[0.5, 1.5]$ konvergiert. Prüfen Sie dazu die Voraussetzungen des Banachschen Fixpunktsatzes.

Der Startwert sei $x_0 = 0.5$. Bestimmen Sie mit Hilfe der a priori Abschätzung die Anzahl der benötigten Schritte, wenn der absolute Fehler der Näherung kleiner als 10^{-7} sein soll.

Aufgabe 4

Betrachten Sie das lineare Gleichungssystem $\boldsymbol{Ax} = \boldsymbol{b}$ mit

$$\boldsymbol{A} = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 2+\varepsilon & 5 \\ 4 & 6 & 8 \end{pmatrix}$$

und

$$\boldsymbol{b} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

wobei ε eine reelle Zahl ist.

(a) (5 Punkte) Berechnen Sie **manuell** die LR−Zerlegung **ohne Zeilenvertauschung** der Matrix $\boldsymbol{A}$ für ein allgemeines $\varepsilon \neq 0$.

(b) (2 Punkte) Bestimmen Sie mit Hilfe der Matrizen $\boldsymbol{L}$ und $\boldsymbol{R}$ aus (a) die Lösung von $\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}$ für $\varepsilon = 2^{-52}$ (Maschinengenauigkeit). Schreiben Sie dazu ein Python-Skript und verwenden Sie numpy.linalg.solve().

(c) (3 Punkte) Lösen Sie nun das lin. Gleichungssystem $\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}$ für $\varepsilon = 2^{-52}$ direkt mit numpy.linalg.solve(). Weshalb erhalten Sie nicht das gleiche Resultat wie bei b)? Begründen Sie!

Aufgabe 5

Mit Hilfe des Jacobiverfahrens soll die Lösung eines linearen Gleichungssystems mit einer n-dimensionalen *tridiagonalen* Systemmatrix $\mathbf{A}$ bestimmt werden. Die Matrix und die rechte Seite sind definiert mit $c > 0$

$$\mathbf{A} = \begin{pmatrix} c & -1 & & & & \\ -1 & c & -1 & & & \\ & -1 & c & -1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & c & -1 \\ & & & & -1 & c \end{pmatrix} \quad \mathbf{b} = \begin{pmatrix} c \\ 0 \\ \vdots \\ 0 \\ c \end{pmatrix}$$

Das Jacobiverfahren kann als Fixpunktiteration $\mathbf{x}_k = \mathbf{B} \cdot \mathbf{x}_{k-1} + \mathbf{v}, \ k = 1, 2, \ldots$ geschrieben werden. Verwenden Sie in der Folge den Nullvektor als Startvektor $\mathbf{x}_0 = \mathbf{0}$.

(a) (2 Punkte) Bestimmen Sie die Matrix $\mathbf{B}$ für $n = 6$, $c = 4.0$ und daraus $||\mathbf{B}||_\infty$.
Für welche $c \in \mathbb{R}$ gilt $||\mathbf{B}||_\infty < 1$?

(b) (2 Punkte) Welche Anzahl Iterationen benötigt man maximal, um eine numerische Lösung mit einer Genauigkeit von 10^{-3} in der ∞-Norm zu erhalten? (a priori Abschätzung)

(c) (5 Punkte) Bestimmen Sie die Lösung mit Hilfe einer Implementierung des Jacobiverfahrens. Führen Sie die vorher bestimmte Anzahl von Iterationen aus!

(d) (1 Punkt) Bestimmen Sie die Genauigkeit der errechneten Lösung mit Hilfe der a posteriori Abschätzung?

Aufgabe 6

Gegeben ist die Matrix

$$A = \begin{pmatrix} 13 & -4 \\ 30 & -9 \end{pmatrix}.$$

a) (4 Punkte) Bestimmen Sie mit Hilfe des charakteristischen Polynoms die Eigenwerte von A, und bestimmen Sie die zugehörigen Eigenräume.

b) (2 Punkte) Bestimmen Sie eine zu A ähnliche Diagonalmatrix D sowie eine zugehörige Basiswechselmatrix T, deren Spalten bezüglich der 2-Norm auf die Länge 1 normiert sein sollen. Es soll dann also gelten: $D = T^{-1}AT$ (Sie dürfen zur Überprüfung T^{-1} mit numpy.linalg.inv() berechnen).

c) (4 Punkte) Bestimmen Sie mit Hilfe der von Mises Iteration numerisch den betragsmässig grössten Eigenwert von A, sowie einen zugehörigen Eigenvektor. Iterieren Sie dabei 40 Mal, ausgehend vom Startvektor $v = (1,\ 0)^T$. Stellen Sie den absoluten Fehler der numerischen Näherung des Eigenwertes in Abhängigkeit der Iterationszahl halblogarithmisch dar.
Hinweis: Verwenden Sie f́ur alle numerischen Berechnungen in dieser Teilaufgabe den Datentyp `numpy.float64`.