Theorie-Fragen

- Laufzeit von QR-Zerlegung

- Laufzeit von Skalarprodukt(x^T, y),
beide Vektoren haben gleich viele Elemente

- Wie nennt man Algorithmen die Fehler
in den Eingabedaten verstärken

- Nenne Sie ein direktes und ein iteratives Verfahren
zur Lösung von LGS

- Was ist der Unterschied zwischen dem
Gauss-Seidel- und dem Jakobiverfahren

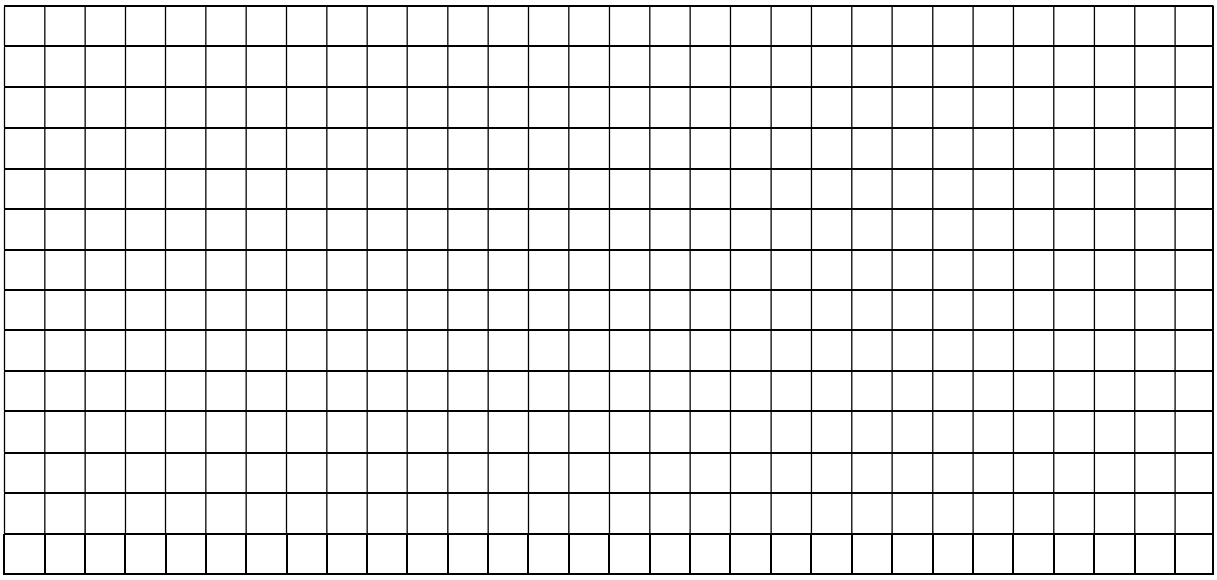
- IEEE 754 Bestandteile außer Vorzeichen

- 3 Merkmale von Bézierkurven

Aufgabe 1 - Matrix Formatierung (11 Punkte)

a) PLR-Zerlegung von

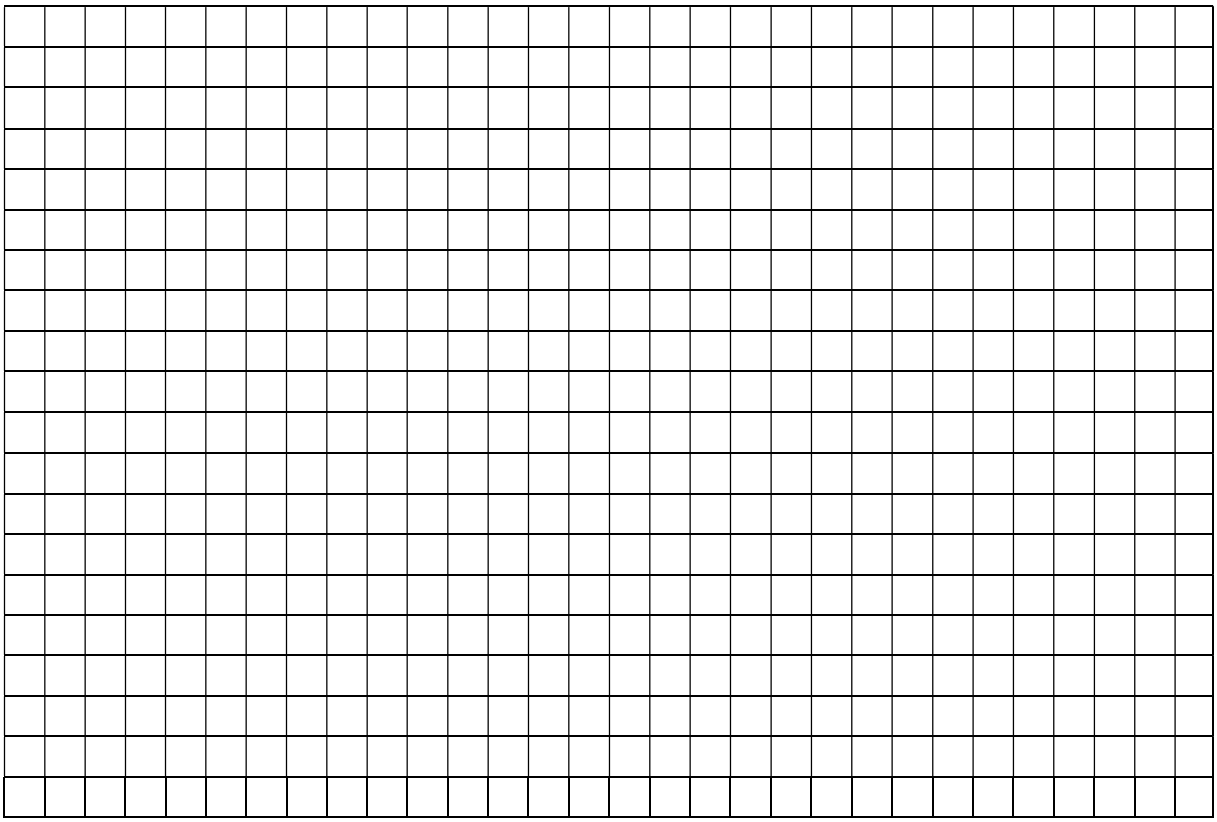
$$A = \begin{bmatrix} 3 & 3 \\ 6 & 9 \end{bmatrix}$$



b) Gegeben ist eine Matrix B und eine Rotationsmatrix S, sodass $S*B = \begin{bmatrix} * & * \\ 0 & * \end{bmatrix}$ gilt.

$$B = \begin{bmatrix} 6 & -4 \\ 8 & 2 \end{bmatrix},\ S = \begin{bmatrix} 0.6 & 0.8 \\ -0.8 & 0.6 \end{bmatrix}$$

Bestimmen Sie die QR-Zerlegung der Matrix B, indem Sie die Rotationsmatrix S verwenden.



c) Von der Matrix C ist folgende QR-Zerlegung bekannt:

$$B = \begin{bmatrix} 0.6 & -0.8 \\ 0.8 & 0.6 \end{bmatrix} * \begin{bmatrix} 10 & 7 \\ 0 & -1 \end{bmatrix}$$

Q R

Lösen sie das lineare Gleichungssystem Cx = b für b = [10, 10]

Aufgabe 3 – Interpolation (10 P)

a) Es ein die Punkte

$$(x_1, y_1) = (0, 1)$$
$$(x_2, y_2) = (1, 2)$$
$$(x_3, y_3) = (2, 4)$$

Gegeben. Bestimmen sie mittels der Monombasis, und der entsprechenden Vendermon… den kleinsten Grades das alle Punkte interpoliert.

b) Warum ist die Bestimmung eines Interpationspolynoms, mittels der Vandermoun…, problematisch? Nennen sie auserdem ein besseres Verfahren, zur Bestimmung, von Interpol…

c) Gegeben sein die Punkte

$$P_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad P_2 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix},$$

$$P_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \qquad P_4 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix},$$

Bestimmen Sie den bilinearen Interpolanten $F_{st}(s,t)$ der Eckpunkte P1, P2, P3, P4 sodass

$$F_{st}(0, 0) = P_1, \quad F_{st}(1, 0) = P_2, \quad F_{st}(0, 1) = P_3, \quad F_{st}(1, 1) = P_4$$

4. Python

a) Median (L)

b) count_even(L)

c) set_one(L , e)
-> alle Elemente die e sind auf 1 gesetzt

d) remove_smallest(L , e)
-> alle Elemente kleiner e sollten entfernt werden

e) every_third(L)
-> Kopie der Liste zurück, in der nur jedes 3. Element der ursprünglichen Liste drin ist

Aufgabe 4 – Python

Aufgabe 5 – Faltung (8 P)

a) Geben sie die Definition der Faltung (f * g)(x) zweier Funktionen *f, g* : D -> R an

b) Nun seien die Funktionen

$$f(x) = \begin{cases} 1 & falls\ |x| \leq 0.5 \\ 0 & sonst \end{cases}$$

$$g(x) = \begin{cases} 0.5 & falls\ |x| \leq 0.25 \\ 0 & sonst \end{cases}$$

gegeben. Werten Sie ihre Faltung (f * g)(x) an den Stellen x = 0 und x = 1 aus.

c) Gegeben sei der 2D Filterkern

$$|w(i, j)| = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 1 & 1 \end{bmatrix}$$

Ist der Filterkern separierbar? Begründen Sie ihre Antwort

d) Wenden Sie den Filterern w aus der vorigen Teilaufgebe auf das folgende 2D Array (Ursprungsarray) an und notieren Sie das Ergebnis im leeren 2D Array (Zielarray) darunter. Wenden Sie den Filterkern an dass stets das komplette Operationsfenster im Ursprungsarray liegt. Die Dimensionen des Zielarray dementsprechend kleiner als die des Ursprungsarrays.

Ursprungsarray:

0	0	0	0	0
0	0	0	0	0
0	1	2	3	0
0	0	0	0	0
0	0	0	0	0

Zielarray (3 x 3):

_	_	_
_	_	_
_	_	_

Aufgabe 6 – Numerik (10 Punkte)

a) Gleichungssystem aufstellen: Zu einer Differentialgleichung sei die Diskretisierungsvorschrift

$2x_i + 3x_{i+1} - x_{i-1} = f_i \quad i = (1, 2, 3)$

Sowie Randbedingungen $x_0 = x_1 = 1$, und ein Kraftterm $f_1 = f_3 = 0$, $f_2 = -\cos(\theta t)$

Stellen Sie ein Gleichungssystem mit Lösungsvektor x auf

$$\begin{bmatrix} \_ & \_ & \_ \\ \_ & \_ & \_ \\ \_ & \_ & \_ \end{bmatrix} \qquad \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} \quad = \quad \begin{bmatrix} \_ \\ \_ \\ \_ \end{bmatrix}$$

b) Betrachten Sie nun folgendes Gleichungssystem

$$\underset{A}{\begin{bmatrix} 6 & -3 \\ 1 & 3 \end{bmatrix}} \begin{bmatrix} x1 \\ x2 \end{bmatrix} = \underset{b}{\begin{bmatrix} 9 \\ 5 \end{bmatrix}}$$

Ist die Matrix singulär? Begründen Sie

Gauß-Seidel-Verfahren Führen Sie eine Iteration den Verfahrens auf dem System Ax = b aus der vorherigen Teilaufgabe mit dem Startwert $x^0 = [x^0_1, x^0_2]^T = [0, 1]^T$ durch. Geben sie den Resultierenden Vektor x^1 an.

Residuum Berechnen Sie die L2-Norm des Residuums für den Vektor $x = [1, 3]^T$ und b sowie die Matrix A.

Aufgabe 7 – SVD (11 Punkte)

a) **Aufwärmübung** Sei $A \in R^{m \times n}$ eine reelle Matrix und $A = U \Sigma V^T$ ihre Singulärwertzerlegung. Welche fundamentale Eigenschaft haben die Matrizen U und V

Und Σ

b) Bestimmen der SVD (Teil 1) Sei nun

$$A = 1/5 \begin{bmatrix} 0 & 0 \\ 4 & -6 \\ 3 & 8 \end{bmatrix}$$

gegeben. Bestimmen Sie Σ und V. Geben Sie Ihren Rechenweg an (Nutzen Sie die Struktur von A^TA)

c) Bestimmen der SVD (Teil 2) Für die Zerlegung $A = U \Sigma V^T$ der gegebenen Matrix A gilt $u_1 = 1/5\ [0, -3, 4]^T$ und $u_2 = 1/5\ [0, 4, 3]^T$ (u_i ist der i-te Spaltenvektor der Matrix U). Geben sie u_3 an.

d) Eigenschaften von A Geben Sie den Rang und die Konditionszahl von A an. Verwenden Sie die Ergebnisse der vorherigen Teilaufgaben oder $u_3 = 3^{-1/2}\,(1, 1, 1)^T$,

$\Sigma = \begin{bmatrix} 4 & 0 \\ 0 & 2 \\ 0 & 0 \end{bmatrix}$, $V = 1$.

Aufgabe 8 – Python (12 Punkte)

Hinweis: In dieser Aufgabe dürfen keine numpy Funktionen verwendet werden. Die Matrizen und Vektoren sind 2D bzw 1D. Der Zugriff auf Elemente einer Matrix A erfolgt mit zB A[4] [1]. In diesem Fall wird auf Zeile 5 und Spalte 2 zugegriffen, da die Indizierung bei 0 beginnt. Bei einem Vektor a erfolgt der Zugriff mit a[2] für das dritte Element.

a) **Polynom auswerten (Hornerschema)** Implementieren Sie die Funktion polynom_eval(cofficients, x). Die Funktion wertet ein Polynom deren Koeffizienten in coefficient gespeichert sind an der Stelle x mit dem Hornerschema aus und gibt das Ergebnis zurück. Der erste Eintrag in coefficiets gehört zur höchsten Potenz. Coefficients enthält einen Eintrag für jede Potenz auch wenn dieser 0 sein sollte. Beispiel: Für das Polynom $2 * x^3 + 5 * x + 8$ wären coefficients = [2, 0, 5, 8] . Die Liste coefficients hat mindestens ein Element.

   def polynom_eval(cofficients, x):

b) Matrix-Norm Implementieren Sie die Funktion row_sum_norm(A). Die Funktion bekommt eine Matrix A übergeben. Die Funktion gibt die Zeilensummennorm der Matrix A zurück. Die Zeilensummennorm ist die maximal Betragssumme der Zeilen. Es muss also die Summe der Beträge der Einträge jeder Zeile berechnet und anschließend das Maximums gefunden werden. Mathematisch bedeutet dies: $\max_{i=1,\dots,m} \sum_{j=1}^{n} |aij|$ für eine Matrix $A \in R^{m \times n}$ also m Zeilen und n Spalten. Sie können Annehmen dass m,n > 0.

def row_sum_norm(A):

c) Matrix-Matrix Multiplikation Implementieren Sie die Funktion mat_mult(A, B). Die Funktion bekommt zwei Matrizen A und B übergeben. Die Funktion gibt das Ergebnis der Multiplikation als 2D-Liste zurück Sie können davon ausgehen, dass die Matrizen quadratisch sind und mindestens eine Dimension von 1x1 haben ($A,B \in R^{n \times n}$, $n \in N$).

```
def mat_mult(A, B):
```

10. BCRS

a) Matrix war gegeben, man sollte die Arrays
val, col_idx und row_idx bestimmen
- Indizierung bei 0 in der Klausur begonnen!

b) Speicherbedarf Differenz „ohne BCRS“
und „mit BCRS“ ausrechnen