Dieser Übungszettel wird nicht bewertet und dient nur zu Ihrer eigenen Klausurvorbereitung.

Beispielklausur Die folgenden Aufgaben sind aus vorigen Klausuren zusammengestellt und dürften Ihnen einen ungefähren Eindruck von Umfang und Format der Klausur geben.

Übung 1 *Verständnisfragen*

a) Was ist die Frobeniusnorm der $n \times n$ Einheitsmatrix? Warum kann daher $\|A\|_F$ für $A \in \mathbb{R}^{n\times n}$ keine induzierte Operatornorm sein?

b) Zu paarweise verschiedenen Stützstellen $x_0, \ldots, x_n$ seien $L_i^{(n)}(x), i = 0, \ldots, n$ die zugehörigen Lagrange-Basispolynome. Warum gilt

$$\sum_{k=0}^{n} L_k^{(n)}(x) = 1 \quad \forall x \in \mathbb{R} \ ?$$

c) Gegeben sei eine "Punktewolke", d.h. eine Menge von n paarweise verschiedenen Stützstellen $x_0, \ldots, x_n$ mit Stützwerten $y_0, \ldots, y_n$. Beschreiben Sie qualitativ den Kurvenverlauf der folgenden Objekte:

1. Interpolationpolynom n-ten Grades $p \in P_n$
2. Lineare Spline-Interpolierende
3. Ausgleichsgerade

d) Warum liegt die Bézier-Kurve in der konvexen Hülle der Bézier-Punkte?

e) Nennen Sie zwei Gründe, warum negative Gewichte bei einer Quadraturformel nicht wünschenswert sind.

(- Punkte)

Übung 2 *Ausgleichsrechnung*
Gegeben seien die folgenden Stützpunkte x_i und Stützwerte y_i für $i \in \{1, 2, 3\}$:

x_i	-2	1	2
y_i	-6	2	-4

In dieser Aufgabe sollen Sie die Ausgleichsgerade $f(x) = a_1 x + a_2$ zu diesen Stützwerten finden, sodass $J(a_1, a_2) = \sum_{i=1}^{3}(f(x_i) - y_i)^2$ minimal wird. Gehen Sie dafür folgendermaßen vor:

a) Stellen Sie das Normalengleichungssystem auf.

b) Bestimmen Sie die Ausgleichsgerade indem Sie das in Teilaufgabe a) aufgestellte Gleichungssystem lösen.

(- Punkte)

Übung 3 *Polynominterpolation*

Lesen Sie bitte den gesamten Aufgabentext einschließlich des Hinweises durch bevor Sie mit der

Bearbeitung dieser Aufgabe beginnen!

Gegeben seien die paarweise verschiedenen Stützstellen $x_0 = -1$, $x_1 = 1$, $x_2 = 2$ und $x_3 = 3$ sowie die zugehörigen Werte $y_0 = 1$, $y_1 = 5$, $y_2 = 4$ und $y_3 = 17$.

a) Bestimmen Sie die Koeffizienten des zugehörigen Interpolationspolynoms $p(x)$ in der Newton Basis.

b) Bestimmen Sie die Koeffizienten des zugehörigen Interpolationspolynoms $p(x)$ in der Lagrange Basis.

c) Bestimmen Sie die Koeffizienten des zugehörigen Interpolationspolynoms $p(x)$ in der Monom Basis. Tipp: Sie können Teilaufgabe a) oder b) verwenden um ein Aufstellen und Lösen des Vandermonde-Gleichungssystems zu vermeiden.

Hinweis: In allen drei Fällen genügt es die Koeffizienten anzugeben. **(- Punkte)**

Übung 4 *LR-Zerlegung*

a) Sei $A = \begin{pmatrix} 2 & 3 & 2 & -1 \\ 4 & 8 & 4 & 1 \\ 6 & 5 & 8 & -10 \\ 4 & 12 & 4 & 10 \end{pmatrix}$.

Berechnen Sie für die Matrix A die LR-Zerlegung ohne Pivotisierung.

b) Ist die LR Zerlegung ohne Pivotisierung ein numerisch stabiler Algorithmus? (Keine Begründung erforderlich).

c) Ist die LR Zerlegung mit Spaltenpivotisierung ein numerisch stabiler Algorithmus? (Keine Begründung erforderlich).

(- Punkte)

Übung 5 *"Programmieraufgabe"*

Sie haben von einem Kommilitonen folgenden Codeabschnitt bekommen. Sie wissen dass alle abgebildeten Funktionen Interpolationspolynome auswerten und dass sie korrekt implementiert sind.

```
double auswertung1(const std::vector<double>& nodes,
                   const std::vector<double>& coefficients,
                   double x)
{
  double basis(1.0);
  double eval = coefficients[0];
  for (std::size_t i=1; i<coefficients.size(); ++i){
    basis *= (x-nodes[i-1]);
    eval += coefficients[i] * basis;
  }
  return eval;
}

double auswertung2(const std::vector<double>& nodes,
                   const std::vector<double>& coefficients,
                   double x)
{
  double eval(0.0);
  for (std::size_t i=0; i<coefficients.size(); ++i){
    double basis(1.0);
    for (std::size_t j=0; j<coefficients.size(); ++j){
```

```
      if (j!=i){
        basis *= (x-nodes[j])/(nodes[i]-nodes[j]);
      }
    }
    eval += coefficients[i] * basis;
  }
  return eval;
}

double auswertung3(const std::vector<double>& coefficients,
                   double x)
{
  double basis(1.0);
  double eval = coefficients[0];
  for (std::size_t i=1; i<coefficients.size(); ++i){
    basis *= x;
    eval += coefficients[i] * basis;
  }
  return eval;
}
```

a) In welcher Basis werden die Interpolationspolynome jeweils ausgewertet? (Es müssen natürlich zur Basis passende Koeffizienten übergeben werden, damit die Auswertung Sinn macht).

b) Wie viele Rechenoperationen werden jeweils benötigt? Gehen Sie von $n+1$ Stützstellen aus, d.h. `nodes` und `coefficients` haben auch jeweils $n+1$ Einträge. Rechenoperationen mit Indizes (z.B `i-1`) werden nicht mitgezählt.

(- Punkte)

Übungsaufgaben Dies ist eine Sammlung von Übungsaufgaben aus vorigen Semestern, die dieses Mal nicht gestellt wurden. Entsprechend decken sie den Inhalt der Vorlesung nicht systematisch ab, bieten Ihnen aber hoffentlich zusätzliche Übungsmöglichkeiten.

Übung 6 *Eigenschaften von $A^T A$*

Sei $A \in \mathbb{R}^{m\times n}$ mit $m \geq n$. Sei $\text{Rang}(A) = n$. Zeige, dass dann gilt:

a) $A^T A \in \mathbb{R}^{n\times n}$ ist symmetrisch positiv definit.

b) $\text{cond}_2(A^T A) = \text{cond}_2(A)^2$.

Bemerkung: Für eine rechteckige Matrix A ist die Kondition allgemein definiert als

$$\text{cond}(A) := \frac{\max\limits_{\|x\|=1} \|Ax\|}{\min\limits_{\|y\|=1} \|Ay\|} .$$

(4 Punkte)

Übung 7 *Rundungsfehler einer quadratischen Gleichung*
Auf einer Rechenmaschine sei 4-stellige Dezimal-Gleitpunktarithmetik mit korrekter Rundung möglich ($\epsilon = 0.5 \cdot 10^{-3}$).

- Zeigen Sie, dass der relative Fehler der Operation $rd\left(\sqrt{rd(p)}\right)$ in erster Näherung durch $\frac{3}{2}\epsilon$ beschränkt ist.
- Berechnen Sie unter diesen Bedingungen möglichst gute Näherungen für die Wurzeln der quadratischen Gleichung
$$p^2 - 12.8\,p + 6.2 = 0.$$
Geben Sie Schranken für den relativen Fehler an.

(3 Punkte)

Übung 8 *Rundungsfehler Skalarprodukt*
Führen Sie eine Rundungsfehleranalyse des Skalarprodukts im $\mathbb{R}^n$

$$F_n(\vec{x},\vec{y}) := \vec{x}\cdot\vec{y} = \sum_{i=1}^{n} x_i y_i \qquad \vec{x},\vec{y} \in \mathbb{R}^n$$

durch, wobei wir annehmen, dass der Algorithmus hierzu durch

$$\begin{aligned} z_0 &:= 0 \\ z_i &:= z_{i-1} + x_i y_i \end{aligned}$$

gegeben ist und z_n die Lösung enthält. Leiten Sie hierzu zunächst eine rekursive Beziehung für $\Delta F_n\,(\,\Delta F_{n-1}\,, rd\,(F_{n-1})\,, x_n,\, y_n\,)$ her.

(4 Punkte)

Übung 9 *Gauß Approximation*
Auf dem Intervall $[-1,1]$ betrachten wir die Polynome $p_0(x) = 1$ und $p_1(x) = -x+1$. Diese bilden den Raum $S := \text{span}\{p_0, p_1\} = \{g \mid g = \alpha p_0 + \beta p_1, \alpha, \beta \in \mathbb{R}\}$ mit dem Skalarprodukt und mit der Norm

$$(f,g) = \int_{-1}^{1} f(x)g(x)dx, \quad \|f\| = \sqrt{(f,f)}.$$

Bestimmen Sie die Funktion $g \in S$ zu $f : [-1,1] \ni x \mapsto \frac{1}{2}(x^2 - 4x + 2)$, so dass

$$\|f - g\| \to \min .$$

(5 Punkte)

Übung 10 *Gauss-Elimination und Zeilenvertauschung*
Betrachten Sie die Matrix

$$A = \begin{pmatrix} 10^{-20} & 1 \\ 1 & 1 \end{pmatrix},$$

(a) Berechnen Sie die Kondition von A in der Spektralnorm.

(b) Berechnen Sie die echte Lösung von $Ax = b$ mit $b = (1,0)^T$.

(c) Berechnen Sie die Lösung numerisch mittels Gauss-Elimination *ohne* Zeilenvertauschung.

(d) Berechnen Sie die Lösung numerisch mittels Gauss-Elimination *mit* Zeilenvertauschung.

Bei den numerischen Berechnungen nehmen wir an, dass wir eine Maschinengenauigkeit von 10^{-16} hätten.

(4 Punkte)

Übung 11 *Störungsanalyse*
Man betrachte das lineare Gleichungssystem

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Der relative Fehler in den Matrixelementen betrage höchstens $\pm 1\%$, derjenige der Komponenten der rechten Seite höchstens $\pm 3\%$.

a) Schätzen Sie die relativen Fehler $||\delta x||_1/||x||_1$ und $||\delta x||_\infty/||x||_\infty$ ab.

b) Zeichnen Sie jeweils die Punktemenge in $\mathbb{R}^2$, in der die Lösung $x + \delta x$ des gestörten Systems liegt.

(4+1 Punkte)

Übung 12 *Horner Schema*
Zur Auswertung eines Polynoms der Ordnung k mit

$$f(x) = \sum_{i=0}^{k} a_i x^i$$

wird gewöhnlich das Horner Schema verwendet. Hierbei berechnet man im Schritt n:

$$z_n = x z_{n-1} + a_{k-n}$$

wobei $z_0 := a_k$ gesetzt wird. Dann gilt $z_k = f(x)$. Zeigen Sie das dieser Algorithmus numerisch stabiler ist als das einzelne Auswerten der Summanden von $f(x)$ indem sie eine Rundungsfehleranalyse für beide Verfahren durchführen.

(3 Punkte)

Übung 13 *Interpolation*

a) Gegeben seien die Wertepaare:

i	0	1	2
x_i	-4	- 1	0
y_i	1	-1	5

Bestimme die Koeffizienten des interpolierenden Polynoms $p(x)$ in der Monom-Darstellung (also die c_i aus $p(x) = \sum_{i=0}^{2} c_i x^i$, wobei wir mit $\{1, x, x^2\}$ die Monombasis bezeichnen).

b) Warum ist die Bestimmung der zugehörigen Koeffizienten bei der Lagrange-Interpolation trivial?

c) Interpolieren Sie die Funktion $f(t) = \frac{1}{1+t^2}$ mit Hilfe des Newton'schen Interpolationspolynoms vom Grad 2 ($p \in P_2$) zwischen den Stützstellen $t_0 = 0$, $t_1 = 1$, und $t_2 = 2$.

(Bonus 1+1+1 Punkte)

Übung 14 *Äquidistante Stützstellen*
Beweisen Sie, dass man bei äquidistanten Stützstellen $x_i = x_0 + jh$, $j = 0, \ldots, n$, $h > 0$ die Newton-Darstellung des Interpolationspolynoms in

$$p(x) = \sum_{k=0}^{n} \binom{s}{k} k! \cdot h^k \cdot f[x_0, \ldots, x_k] \qquad \text{mit } s = \frac{x - x_0}{h}$$

umwandeln kann, wobei der Binomialkoeffizient durch

$$\binom{s}{k} = \frac{s \cdot (s-1) \cdots (s-k+1)}{k!} \tag{1}$$

auch für eine reelle Zahl $s \in \mathbb{R}$ definiert ist.

(5 Punkte)

Übung 15 *Newton Interpolation Fehler (Bonusaufgabe)*
Auf dem Aufgabenblatt 10 in der Übung 1 hat man die Funktion $f(t) = \sqrt{t}$ mit Hilfe des Newton'schen Interpolationspolynoms vom Grad 2 ($p \in P_2$) zwischen den Stützstellen $t_0 = \frac{1}{4}$, $t_1 = 1$, und $t_2 = 4$ interpoliert.

Vergleichen Sie den Interpolationsfehler im Punkt $t = 2$ mit der Fehlerabschäatzung für Interpolationspolynome glatter Funktionen.

Hinweis: Rechnen Sie den maximalen Wert von $f^{(3)}$ in $[\frac{1}{4}, 4]$ aus.
Als Bonuspunkt könnten Sie die Nullstelle von $F_x^{(3)}$ finden und den Interpolationsfehler exakt berechnen.
Das Interpolationspolynom ist $p(t) = -\frac{4}{45}t^2 + \frac{7}{9}t + \frac{14}{45}$.

(2+1 Punkte)

Übung 16 *LR-Zerlegung*
Gegeben sei die Matrix

$$A = \begin{pmatrix} 1 & 2 & 3 & -4 \\ 2 & 8 & 6 & -14 \\ 3 & 6 & \alpha & -15 \\ -4 & -14 & -15 & 30 \end{pmatrix}$$

mit einem reellen Parameter α. Berechnen sie die zugehörige LR-Faktorisierung (ohne Pivotisierung), beziehungsweise geben Sie an, für welchen Wert des Parameters α diese nicht existiert.

(Bonus 2 Punkte)

Übung 17 *Matrix Kondition und Störungstheorie*

Wir betrachten das lineare Gleichungssystem $Ax = b$

$$\begin{pmatrix} 100 & 2 \\ 2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

a) Berechnen Sie die zur Zeilensummennorm gehördende Kondition $\mathrm{cond}_\infty(A)$ der Matrix A.

b) Geben Sie eine Abschätzung für den relativen Fehler $\|\delta x\|_\infty / \|x\|_\infty$, wenn der relative Fehler in den Matrixelementen höchstens $\pm 2\%$ und der in den Komponenten der rechten Seite höchstens $\pm 5\%$ beträgt?

(Bonus 1+1 Punkte)

Übung 18 *Zeilenäquilibrierung*

Eine Matrix $A \in \mathbb{R}^{n\times n}$ heißt *zeilenäquilibriert*, wenn die Zeilensumme $\sum_{j=1}^{n} |a_{ij}|$ für jedes $i \in \{1, ..., n\}$ identisch ist.

(a) Zeigen Sie: $A \in \mathbb{R}^{n\times n}$ sei regulär. Dann existiert eine reguläre Diagonalmatrix $D \in \mathbb{R}^{n\times n}$, so dass die Matrix $D \cdot A \in \mathbb{R}^{n\times n}$ zeilenäquilibriert ist.

(b) Zeigen Sie: Ist $A \in \mathbb{R}^{n\times n}$ bereits zeilenäquilibriert und ist $D \in \mathbb{R}^{n\times n}$ irgendeine reguläre Diagonalmatrix, so gilt:

$$\mathrm{cond}_\infty(DA) \geq \mathrm{cond}_\infty(A)$$

(4 Punkte)

Übung 19 *Überbestimmte Gleichungssysteme*

Gegeben seien die folgenden Datenpunkte

x	-2	-1	0	1	2
y	3	3	1	-1	-1

Bestimmen Sie die Gaußsche Ausgleichsgerade $y = ax + b$ zu diesen Punkten, also die Gerade, die die Punkte im Sinne von kleinsten Fehlerquadraten am besten beschreibt. **(Bonus 2 Punkte)**

Übung 20 *Curve-Fitting*

Gegeben seien die folgenden Wertepaare:

z_i	0	0.15	0.31	0.5	0.6	0.75
y_i	1.0	1.004	1.031	1.117	1.223	1.422

Gesucht ist ein reelles Polynom $P(z)$ ersten oder zweiten Grades, das den Fehler

$$\chi^2 = \sum_{i=1}^{6} |P(z_i) - y_i|^2$$

minimiert.

Zur Erinnerung: Ein reelles Polynom der Ordnung k hat die Gestalt: $P_k(z) = x_0 + x_1 z + ... + x_k z^k$ mit Koeffizienten $x_0, ..., x_k \in \mathbb{R}$.

a) Formulieren Sie das Problem um, in die Gestalt: Finden Sie ein $x = (x_0, x_1, \ldots, x_k) \in \mathbb{R}^{k+1}$ so, dass $\|Ax - b\|_2^2$ minimal ist und geben Sie (für $k = 1$ und $k = 2$) die zugehörige Matrix A sowie den Vektor b an.

b) Stellen Sie die Normalengleichung für $k = 1$ und $k = 2$ auf.

c) Ist die Lösung der Normalengleichung eindeutig?

Sie brauchen die Normalengleichung aus b) hier nicht zu lösen.

(5 Punkte)

Übung 21 *Maschinengenauigkeit (Praktische Übung)*

Schreibe ein C++-Programm namens `precision`, das folgende Aufgaben erledigt:
Wenn der Benutzer die Kommandozeile

```
./precision
```

aufruft, soll zunächst eine beliebige Zahl z abgefragt werden. Dann soll (näherungsweise) ermittelt werden, für welche Fließkommazahl x_0 gerade noch die Bedingung

$$z < z + x_0$$

erfüllt ist.
Der Algorithmus könnte z.B. so aussehen:

```
1.) Setze x=1.0
2.) Berechne c=z+x
3.) Solange die Bedingung z<c erfüllt ist,
    halbiere x
    und
    berechne wieder c=z+x.
```

Verwende bedingte Anweisungen und Schleifen. Benutze als Datentyp für Fließkommazahlen den Typ `double`. Nach jedem Halbierungsschritt soll das Programm die Werte von x und $z + x$ am Bildschirm ausgeben und am Ende auch den Wert von x_1, für welches z praktisch identisch ist mit $z + x_1$.
Teste das Programm für $z = 10^{-5}$, $z = 1$ und $z = 10^5$ und vergleiche die jeweiligen x_0 und x_1 für die verschiedenen Werte von z.

(5 Punkte)

Übung 22 *QR-Zerlegung von A*
Berechnen Sie die QR-Zerlegung der Matrix

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

per Hand mit Hilfe des Householder-Verfahrens.

(5 Punkte)

Übung 23 *QR-Zerlegung Analyse*
Analysieren Sie die Anzahl der Elementaroperationen und den Speicheraufwand, die zur Durchführung des Householder-Algorithmus zur Berechnung der QR-Zerlegung einer Matrix $A \in \mathbb{R}^{m \times n}$ notwendig sind. Diskutieren Sie die Fälle $m \approx n$ und $m \gg n$. Vergleichen Sie die Werte mit denen von Gauss-Algorithmus.

(4 Punkte)

Übung 24 *Splines*
Seit $X = (x_0, x_1, \cdots, x_n)$ mit $a = x_0 < x_1 < \cdots < x_n = b$ eine Zerlegung des Intervalles $[a, b]$.

a) Sei $f \in C^2([a,b])$. Für den zugehörigen interpolierenden linearen Spline $s \in S^1(X)$ beweisen Sie mithilfe der Taylorschen Formel die folgende Fehlerabschätzung:

$$|s'(x) - f'(x)| \leq \frac{1}{2}\|f''\|_\infty h_{max} \quad \text{für } x \in [a,b], \quad x \notin X,$$

wobei $h_{max} = \max\limits_{j=0,\cdots,n-1} |x_{j+1} - x_j|$ den maximalen Knotenabstand bezeichnet.

b) Geben Sie die Bedingungen für quadratische Splines $S^2(X)$ mit periodischen Randbedingungen an.

c) Warum sind die quadratische Splines in der Praxis für nichtperiodische Randbedingungen unbeliebt?

(Bonus 1+1+1 Punkte)