

Numerische Mathematik

Sommersemester 2022

Übungsblatt 1

Hausaufgaben (Abgabe: bis 19.04.2022 10⁰⁰ Uhr)

Abgabe paarweise — Achten Sie darauf, dass beide Namen auf der abgegebenen Lösung vermerkt sind.

Prüfungszulassungsvoraussetzung: Laut Modulbeschreibung sind 50% der Hausaufgabenpunkte zu erreichen, um zur Prüfung zugelassen zu werden.

Laden Sie Hausaufgabenlösungen als PDF (nicht zu große Dateien!) und Programme als Quellcode (.c, .py, ...) in Moodle hoch.

Hausaufgabe 1.1: *Approximation durch Taylorpolynome*

Berechnen Sie jeweils $T_2 f(x; 0)$, also das Taylorpolynom von f vom Grad 2 zum Entwicklungspunkt 0, und schätzen Sie $|f(1) - T_2 f(1; 0)|$ ab. **Anmerkung:** Der Arbeitsauftrag „Schätzen Sie Y ab.“ bedeutet, dass Sie $s, S \in \mathbb{R}$ mit $s \leq Y \leq S$ bestimmen sollen, ohne dafür den exakten Wert Y zu verwenden. Dabei soll s möglichst groß und S möglichst klein sein.

a) (2 P.) $f(x) := \frac{1}{1+x}$

b) (2 P.) $f(x) := \sqrt{1+x}$

Hausaufgabe 1.2: *Kombination von Fehlerabschätzungen*

Bekanntlich ist $\forall x \in \mathbb{C}$: $\sin(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{(2n+1)!}$ und $\cos(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n}}{(2n)!}$. Wir setzen in dieser Aufgabe nicht voraus, dass wir $\cos(x)$ und $\sin(x)$ exakt berechnen können, und brechen die Reihen daher jeweils nach 11 Summanden ab: $\sin(x) \approx s(x) := \sum_{n=0}^{10} \frac{(-1)^n x^{2n+1}}{(2n+1)!}$ und $\cos(x) \approx c(x) := \sum_{n=0}^{10} \frac{(-1)^n x^{2n}}{(2n)!}$. Sei $a := 3.140625$.

a) (2 P.) Berechnen Sie möglichst kleine $\delta_s, \delta_c \in \mathbb{R}_{>0}$ mit $s(a) - \delta_s \leq \sin(a) \leq s(a) + \delta_s$ und $c(a) - \delta_c \leq \cos(a) \leq c(a) + \delta_c$. **Hinweis:** Leibniz-Kriterium.

b) (2 P.) Bestimmen Sie mit Hilfe des Mittelwertsatzes ausgehend von $a, s(a), c(a), \delta_s$ und δ_c ein möglichst großes $L \in \mathbb{R}$ und ein möglichst kleines $U \in \mathbb{R}$, so dass $L \leq \pi \leq U$.

Bitte wenden

Hausaufgabe 1.3: *Gerundete Auswertung von Ausdrücken*

In dieser Aufgabe geht es bereits um das nächste Vorlesungsthema, nämlich Rundung in normierter Gleitkommadarstellung. Für $\ell \in \mathbb{N}^*$ sei \mathbb{G}_ℓ die Menge der von Null verschiedenen reellen Zahlen, deren normalisierte Gleitkommadarstellung zur Basis 10 eine Mantissenlänge ℓ hat. Für $x \in \mathbb{R}^*$ seien $x_-, x_+ \in \mathbb{G}_\ell$ so, dass $x_- < x_+$, $x_- \leq x \leq x_+$ und $\nexists y \in \mathbb{G}_\ell: x_- < y < x_+$. Man kann sich überlegen, dass die Mantissen von x_- und x_+ sich um genau $10^{-\ell+1}$ unterscheiden. Wir definieren nun $\text{Rd}_\ell(x) := x_-$, falls $|x - x_-| < |x - x_+|$ oder falls $|x - x_-| = |x - x_+|$ und die $(\ell - 1)$ -te Nachkommastelle der (normalisierten) Mantisse von x_- ist gerade; andernfalls sei $\text{Rd}_\ell(x) := x_+$. Zudem sei $\text{Rd}_\ell(0) := 0$.

Bsp: $\text{Rd}_2(-0.000344) = -3.4 \times 10^{-4} = -0.00034$, $\text{Rd}_2(205142) = 2.1 \times 10^5 = 210000$, $\text{Rd}_2(2.45) = 2.4$ (NICHT = 2.5!), $\text{Rd}_2(2.35) = 2.4$, $\text{Rd}_2(9.88) = 10$.

Wenn die Mantissenlänge ℓ vorgegeben ist, definieren wir für $a, b \in \mathbb{R}$ die gerundeten Grundrechenarten wie folgt: $a \boxplus b := \text{Rd}_\ell(a + b)$, $a \boxminus b := \text{Rd}_\ell(a - b)$, $a \boxtimes b := \text{Rd}_\ell(a \cdot b)$, $a \boxdiv b := \text{Rd}_\ell(a/b)$.

- a) (2 P.) Für $x \in \mathbb{R}$ sei $A_x := (x + 1) \cdot (x + 1) = (x + 2) \cdot x + 1$. Begründen Sie, warum für manche betragsmäßig kleine x gilt $|((x \boxplus 1) \boxminus (x \boxplus 1)) - A_x| > |(((x \boxplus 2) \boxtimes x) \boxplus 1) - A_x|$. Geben Sie dafür auch ein Beispiel, mit Mantissenlänge $\ell = 2$.
- b) (1 P.) Zeigen Sie durch Angabe von Gegenbeispielen für Mantissenlänge $\ell = 2$, dass weder \boxplus noch \boxtimes das Assoziativgesetz erfüllen.

Programmieraufgabe 1.4: *IEEE vs. Posit*

Es sei \mathbb{B}_{16} die Menge aller Bitlisten der Länge 16. Jedes $B \in \mathbb{B}_{16}$ kann man einerseits als 16-Bit-Gleitkommazahl $I_B \in \mathbb{Q}$ nach dem IEEE 754-Standard (half precision mit $r = 5$ $p = 10$) und andererseits als 16-Bit-Posit $P_B \in \mathbb{Q}$ nach dem Posit-Standard interpretieren. Zudem kann man B als **Zweierkomplement** $K_B \in \mathbb{Z}$ interpretieren.

(5 P.) Schreiben Sie Funktionen, die für jedes $B \in \mathbb{B}_{16}$ I_B , P_B und K_B berechnen. Stellen Sie bei geeigneter Skalierung der Achsen die Punktmengen

$$L_I := \{(K_B, \text{sgn}(I_B) \cdot \ln(|I_B|)) \mid B \in \mathbb{B}_{16}, I_B \in \mathbb{R}^*\} \subset \mathbb{R}^2$$

und

$$L_P := \{(K_B, \text{sgn}(P_B) \cdot \ln(|P_B|)) \mid B \in \mathbb{B}_{16}, P_B \in \mathbb{R}^*\} \subset \mathbb{R}^2$$

graphisch dar, L_I in blauer und L_P in roter Farbe.

Erreichbare Punktzahl: 16