Algorithmische Mathematik I

Wintersemester 2017/18
Prof. Dr. Ira Neitzel
AR. Dr. Tino Ullrich

universitätbonn

Übungsblatt 4.

Abgabe am **06.11.2017 vor der Vorlesung**.

Aufgabe 1. (Kondition)

Erklären Sie den Begriff „Kondition“. Berechnen Sie die absoluten und relativen Konditionszahlen der folgenden Funktionen und geben Sie an, wo die Funktionsauswertung qualitativ gut bzw. schlecht konditioniert ist. Geben Sie einen Algorithmus zur Berechnung von f_1 an, der Auslöschung vermeidet.

a. $f_1(x) = \frac{1-\cos x}{x} \quad , \quad x \neq 0\,,$

b. $f_2(x) = \sqrt[3]{x} \quad , \quad x \geq 0\,.$

Hinweis: Benutzen Sie Ableitungsregeln, die Sie aus der Schule kennen und für a) die Tatsache, dass $\sin(x)/x \approx 1$ für $x \approx 0$. Verwenden Sie außerdem Additionstheoreme vom Typ $\sin(x+y) = \sin(x)\cos(y)+\cos(x)\sin(y)$ und $\cos(x+y) = \cos(x)\cos(y)-\sin(x)\sin(y)$ sowie den trigonometrischen Pythagoras $\sin^2(x) + \cos^2(x) = 1\,.$

$(2 + 2 + 2 = 6$ Punkte)

Aufgabe 2. (Rekursion)

In Analogie zum Beispiel aus der Vorlesung betrachten wir jetzt die Folge $(y_i)_{i\in\mathbb{N}}$ mit

$$y_i = \frac{1}{e}\int_0^1 e^x x^i dx.$$

a. Bestimmen Sie je eine Vorwärts- und eine Rückwärtsrekursionsformel, mit der der Wert y_i aus y_{i-1} bzw. y_{i+1} bestimmt werden kann. Nutzen Sie dabei partielle Integration

$$\int_a^b f'(x)g(x)dx = [f(x)g(x)]_a^b - \int_a^b f(x)g'(x)\,dx\,.$$

Mit der Vorwärtsrekursionsformel lassen sich somit sukzessiv die Folgenglieder $y_1, y_2, y_3, \ldots$ berechnen, falls der Startwert y_0 bekannt ist. Entsprechend lassen sich mit der Rückwärtsrekursionsformel die Folgenglieder $y_{k-1}, y_{k-2}, \ldots, y_0$ berechnen, falls ein Startwert y_k mit $k \in \mathbb{N}$ bekannt ist.

b. Anstelle der exakten Startwerte y_0 bzw. y_k seien genäherte Startwerte $\tilde{y}_0$ bzw. $\tilde{y}_k$ gegeben. Wie lauten die absoluten Fehler $\Delta y_i = y_i - \tilde{y}_i$ bei Vorwärts- und Rückwärtsrekursion in Abhängigkeit von Δy_0 bzw. Δy_k?

$(4 + 4 = 8$ Punkte)

Aufgabe 3. Für gegebene Daten $x_i \in \mathbb{R}$, $(i = 1, \ldots, M)$ berechnet sich der *Mittelwert* als

$$\bar{x} = \frac{1}{M}\sum_{i=1}^{M} x_i.$$

Die *Abweichung* der Daten ist definiert als

$$s_x^2 = \frac{1}{M} \sum_{i=1}^{M} (x_i - \bar{x})^2 \,. \tag{1}$$

a. Zeigen Sie zunächst

$$s_x^2 = \left(\frac{1}{M} \sum_{i=1}^{M} x_i^2 \right) - \bar{x}^2. \tag{2}$$

Aus der Vorlesung wissen Sie, dass Assoziativität und Distributivität für Operationen mit Gleitkommazahlen im Allgemeinen nicht gelten. Hierzu sehen wir uns folgendes Beispiel an. Es seien

$$\epsilon = 0.01 \qquad M = 2 \qquad x_1 = 1 - \epsilon \qquad x_2 = 1 + \epsilon.$$

b. Bestimmen Sie s_x^2 exakt (normales Rechnen).

c. Bestimmen Sie s_x^2 in Gleitpunktarithmetik zur Basis 10 mit 3 gültigen Stellen (also nach jeder Operation auf 3 Stellen runden, nicht nach der dritten Nachkommastelle runden !) mit beiden Formeln (1) und (2).

d. Welchen Vorteil hinsichtlich der Implementation bietet Formel (2) für s_x^2 ?

(2 + 1 + 2 + 1 = 6 Punkte)

Programmieraufgabe 1. (Rekursion)

Schreiben Sie ein C/C++-Programm, das die Werte $y_0, y_1, \ldots, y_k$ aus Aufgabe 2 für $k = 30$ mittels Vorwärts- und Rückwärtsrekursion berechnet. Verwenden Sie als Startwert für die Vorwärtsrekursion den exakten Wert

$$y_0 = \frac{e-1}{e}$$

und als Startwert für die Rückwärtsiteration die Werte

$$y_k = \frac{1}{2} \left(\frac{1}{e(k+1)} + \frac{1}{k+1} \right) \qquad \text{und} \qquad y_k = 10^6,$$

also einen guten und einen vollkommen schlechten Startwert. Stellen Sie die Ergebnisse tabellarisch dar und beurteilen Sie sie im Hinblick auf Aufgabe 2 b) .

(2 + 2 + 1 = 5 Punkte)

Beide Programmieraufgaben werden in der Woche vom 06.11.2017 bepunktet.

Programmieraufgabe 2. (Die harmonische Reihe)

Die *harmonische Reihe* $s = \lim_{n \to \infty} s_n$ mit

$$s_n = \sum_{k=1}^{n} \frac{1}{k}$$

is eine bekannte divergente Reihe.

a. Schreiben Sie ein C/C++ Programm, welches für gegebenes $n \in \mathbb{N}$ den Wert s_n bestimmt. Dies soll im Format `float` geschehen. Testen Sie Ihr Programm für $n = 2^i$, $i = 5, 10, 30$. Was fällt Ihnen dabei auf?

b. Modifizieren Sie Ihr Programm so, dass das erste Folgenglied als Startwert gesetzt wird und anschließend jeweils blockweise die Summe der Folgenglieder für Blöcke der Größe 2^k für $k = 1, 2, ..., m$ berechnet und schließlich aufsummiert, d.h

$$s_{2^n} = 1 + \underbrace{\frac{1}{2}} + \underbrace{\left(\frac{1}{3} + \frac{1}{4}\right)} + \underbrace{\left(\frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8}\right)} + ...$$

Vergleichen Sie die Ergebnisse mit den Ergebnissen aus Teil a). Was fällt auf und wie erklären Sie sich das Verhalten?

c. Bestimmen Sie eine Approximation der *Euler-Mascheronischeschen Konstante*. Diese ist definiert durch

$$\gamma := \lim_{n\to\infty} (s_n - \ln(n)) .$$

(2 + 2 + 1 = 5 Punkte)

Beide Programmieraufgaben werden in der Woche vom 06.11.2017 bepunktet.