Übungsblatt 5.

Abgabe am **13.11.2017 vor der Vorlesung**.

Aufgabe 1. (Landau-Symbole)

Gegeben sei die Funktion

$$f(x) = 32x^4 - 12x^5 + 13x^7(\ln(x))^4 + x^9 \quad , \quad x > 0.$$

Geben Sie einfache Funktionen $g_1(x), g_2(x)$ an, so dass

a. $f = \mathcal{O}(g_1)$ für $x \to \infty$, d.h. $a = \infty$,

b. $f = o(g_2)$ für $x \to 0$, d.h. $a = 0$.

(2 + 2 = 4 Punkte)

Aufgabe 2. Beweisen Sie den folgenden Satz. Seien alle $|\varepsilon_i| \leq \varepsilon < 1/n$ für $1 \leq i \leq n$ sowie δ definiert durch

$$1 + \delta = \prod_{i=1}^{n} (1 + \varepsilon_i)^{\pm 1}$$

Dann gilt

$$|\delta| \leq \frac{n \cdot \varepsilon}{1 - n \cdot \varepsilon}.$$

Bemerkung: Dabei bedeutet "± 1" im Exponenten, daß bei $(1 + \varepsilon_i)$ für $1 \leq i \leq n$ der Exponent entweder $+1$ oder -1 lautet, aber nicht für alle $1 \leq i \leq n$ identisch zu sein braucht !

(4 Punkte)

Aufgabe 3. (Stabilitätsanalyse)

Es seien Maschinenzahlen $a_1, ..., a_n$ gegeben und

$$f(a_1, ..., a_n) = \sum_{i=1}^{n} a_i\,.$$

Der Computer habe die Maschinengenauigkeit ε und wir setzen $n \leq (2\varepsilon)^{-1}$ voraus. Es wird sukzessive addiert und nach jedem Zwischenergebnis gerundet, was in einer Funktion $\tilde{f}(a_1, ..., a_n)$ resultiert.

a. Zeigen Sie, dass $\tilde{f}$ rückwärtsstabil ist, d.h. betrachten Sie

$$\tilde{f}(a_1, ..., a_n) = f(a_1(1 + \delta_1), ..., a_n(1 + \delta_n))$$

und zeigen Sie, dass sich die relativen Eingabefehler δ_i wie $|\delta_i| \leq C\varepsilon$ verhalten. Dabei ist die Konstante C von den a_i unabhängig.

b. Zeigen Sie, dass für den absoluten Fehler zwischen $\tilde{f}$ und f gilt

$$|\tilde{f}(a_1, ..., a_n) - f(a_1, ..., a_n)| \leq 2\varepsilon \sum_{i=1}^{n} |n - i + 1||a_i|$$

gilt.

c. In welcher Reihenfolge sollte summiert werden, um den Fehler zu minimieren?

Hinweis: Verwenden Sie Aufgabe 2!

(3 + 2 + 1 = 6 Punkte)

Aufgabe 4. (Was macht dieses Programm? (schriftlich bearbeiten!))

Gegeben seien n Gleitkommazahlen $a_1, \ldots, a_n$. Damit wird folgendes Programm ausgeführt (dabei bezeichne := die Zuweisung):

$s := a_1$
$c := 0$
for $i := 2, \ldots, n$:
    $t_1 := a_i - c$
    $t_2 := s + t_1$
    $t_3 := t_2 - s$
    $s := t_2$
    $c := t_3 - t_1$
return s

a. Was berechnet das Programm, wenn alle Operationen ohne Rundungsfehler ausgeführt werden?

b. Was berechnet das Programm, wenn bei der Operation $t_2 := s + t_1$ Rundungsfehler auftreten, alle anderen Operationen aber exakt berechnet werden?

c. Geben Sie ein Beispiel an, wo dieser Algorithmus nützlich ist.

(2 + 2 + 2 = 6 Punkte)