# Praktische Mathematik I

Wintersemester 2004/05
Prof. Dr. Karl Scherer, Marcel Arndt

Rheinische Friedrich-Wilhelms-Universität

## Aufgabenblatt 2

Ausgabe: 21.10.2004, Abgabe der Lösungen: 28.10.2004, 10:10
Abgabe der Programmieraufgabe: 8.-12.11.2004, genauer Termin nach Vereinbarung.

**Klausur: Samstag, 5.2.2005, 9:00 Uhr im Wolfgang-Paul-Hörsaal der Physik, Kreuzbergweg.**

**Aufgabe 5:**
Gemäß Vorlesung wird die Berechnung des Skalarprodukts $\sum_{i=1}^n x_i y_i$ mit $x_i, y_i \in \mathbb{R}$ und $n \in \mathbb{N}$ in die Berechnung der Teilsummen

$$S_r = \mathrm{gl}(\mathrm{gl}(x_r y_r) + s_{r-1})$$

aufgespalten. Bezeichnet eps die Maschinengenauigkeit, so läßt sich $S_r$ schreiben als

$$S_r = (x_r y_r (1 + \eta_r) + s_{r-1})(1 + \xi_r)$$

mit $|\eta_r|, |\xi_r| \le \mathrm{eps}$.
a) Zeige, daß sich $S_n$ zwecks Rückwärtsfehleranalyse als

$$S_n = \sum_{k=1}^n x_k y_k (1 + \varepsilon_k)$$

schreiben läßt, also exakt als ohne Rundungsfehler berechnetes Skalarprodukt zu den gestörten Daten $\tilde{x}_k = x_k(1 + \varepsilon_k)$ anstelle von $x_k$. Wie sieht $\varepsilon_k$ aus?
b) Folgere aus der Darstellung von $\varepsilon_k$ die Abschätzung

$$(1 - \mathrm{eps})^{n-k+2} \le 1 + \varepsilon_k \le (1 + \mathrm{eps})^{n-k+2}$$

und zeige für genügend kleines eps

$$|\varepsilon_k| \le \frac{(n-k+2)\mathrm{eps}}{1 - (n-k+1)\mathrm{eps}}.$$

(10 Punkte)

**Aufgabe 6:**
a) Bestimme die Binär-, Oktal- und Hexadezimaldarstellung der Dezimalzahlen 108 und -30 sowie ihre Bitdarstellungen als 8-bit signed integer gemäß Vorlesung.
b) Betrachte die (binäre) Gleitkommadarstellung bestehend aus normalisierten Gleitkommazahlen mit Exponent $e \in \{-3, \dots, 3\}$ und denormalisierten Gleitkommazahlen mit Exponent $e = -3$. Die Mantissenlänge sei $m = 4$. Bestimme die Darstellungen der Zahlen

$$\frac{5}{2}, \quad \frac{1}{3} \quad \text{und} \quad \frac{-1}{10}.$$

Berechne (in der Gleitkommadarstellung)

$$\frac{5}{2} - \frac{1}{10} \quad \text{und} \quad \frac{5}{2} \cdot \frac{1}{3}.$$

(10 Punkte)

**Aufgabe 7:**
Zu berechnen sei die Summe $f(x_1, \dots, x_n) = \sum_{i=1}^n x_i$ positiver Maschinenzahlen $x_i$. Schätze den relativen Fehler bei der Berechnung $\tilde{f}$ von $f$ in der Form

$$\frac{\tilde{f}(x_1, \dots, x_n) - f(x_1, \dots, x_n)}{f(x_1, \dots, x_n)} \le C(x_1, \dots, x_n) \cdot \mathrm{eps}$$

ab, wobei eps die Maschinengenauigkeit bezeichnet. In welcher Reihenfolge sollte die Summation durchgeführt werden, um den Fehler möglichst gering zu halten? (10 Punkte)

**Aufgabe 8:**
Gegeben seien die Integrale
$$I_k = \frac{1}{e}\int_0^1 e^x x^k \mathrm{d}x$$
für $k \in \mathbb{N}$.
a) Zeige die Abschätzungen
$$\frac{1}{e(k+1)} < I_k < \frac{1}{k+1} \qquad \text{und} \qquad 0 < I_{k+1} < I_k.$$
b) Zeige die Rekursionsformel
$$I_{k+1} + (k+1)I_k = 1.$$
Damit lassen sich also die Integrale $I_0, I_1, \dots, I_K$ rekursiv berechnen:

| | | |
|---|---|---|
| Vorwärtsrekursion: | $I_{k+1} := 1 - (k+1)I_k$ | $k = 0, 1, 2, \dots, K-1$ |
| Rückwärtsrekursion: | $I_{k-1} := \frac{1 - I_k}{k}$ | $k = K, K-1, \dots, 1$ |

falls Startwerte $I_0$ bzw. $I_K$ bekannt sind.
c) Anstelle der exakten Startwerte $I_0$ bzw. $I_K$ seien genäherte Startwerte $\tilde{I}_0$ bzw. $\tilde{I}_K$ gegeben. Wie lauten die absoluten Fehler $\Delta I_k = I_k - \tilde{I}_k$ bei Vorwärts- und Rückwärtsrekursion? (10 Punkte)

**Programmieraufgabe 1:**
Schreibe ein C-, C++- oder Javaprogramm, das die Integrale $I_0, I_1, \dots, I_K$ aus Aufgabe 8 für $K = 30$ mittels Vorwärts- und Rückwärtsrekursion berechnet. Verwende als Startwert für die Vorwärtsrekursion den exakten Wert
$$I_0 = \frac{e-1}{e}$$
und als Startwert für die Rückwärtsiteration die Werte
$$I_K = \frac{1}{2}\left(\frac{1}{e(K+1)} + \frac{1}{K+1}\right) \qquad \text{und} \qquad I_K = 10^6,$$
also das arithmetische Mittel der Abschätzungen oben und einen vollkommen schlechten Wert. Stelle die Ergebnisse tabellarisch dar und beurteile sie, auch im Hinblick auf Aufgabe 8 c). (10 Punkte)

**Beispielprogramm in C:**
Das folgende kleine Beispielprogramm berechnet die Sinusfunktion an einigen Stellen und gibt die Werte tabellarisch aus.

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>

int main()
{
    /* Variablen deklarieren */
    int k;
    double x;
    double y[9];
    /* Schleife: k = 0, 1, 2, ..., 8 */
    for (k=0; k<9; k=k+1)
        {
        /* Werte berechnen */
        x = k * M_PI / 8;
        y[k] = sin(x);

        /* Ausgabe */
        printf("k=%d x=%e y[k]=%e\n", k, x, y[k]);
        }
    return 0;
}
```

**Kurzanleitung zur Verwendung:**
Starte an der Kommandozeile einen Editor:
```
nedit beispiel.c &
```
Gib das Programm ein und speichere es ab.
Kompiliere es mit
```
gcc beispiel.c -o beispiel -lm
```
Führe es aus:
```
beispiel
```