Dirk Praetorius, Samuel Ferraz-Leite
Christoph Überhuber

Sommersemester 2010
30. März 2010

Übungen zur Vorlesung
Einführung in das Programmieren für TM

Serie 3

Die Aufgaben sind freiwillig bis zur Übung nach Ostern vorzubereiten. Kopieren Sie die Source-Codes vor der Übung auf Ihren Account auf der `lva.student.tuwien.ac.at` *und überprüfen Sie, ob diese mit dem* `gcc` *compiliert werden können. In den folgenden Übungsaufgaben sollen* ***Pointer*** *und* ***Schleifen*** *geübt werden.*

Aufgabe 3.1. Welchen Output liefert das folgende Programm und warum?

```
#include <stdio.h>

void square(double* x)
{
  double* y;
  *y=(*x)*(*x);
}

int main(){
  double x=2.1;
  square(&x);
  printf("x^2=%f\n",x);
  return 0;
}
```

Verändern Sie das Programm geeignet, sodass der Ouput den Erwartungen entspricht.

Aufgabe 3.2. Die Fibonacci-Folge ist definiert durch $a_0 := 0$, $a_1 := 1$ und $a_{n+1} := a_n + a_{n-1}$. Schreiben Sie eine Funktion `fibonacci`, die zu gegebenem Index n das Folgenglied a_n zurückgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das den Index n einliest und a_n ausgibt.

Aufgabe 3.3. Die Quotientenfolge $(a_{n+1}/a_n)_{n\in\mathbb{N}}$ zur Fibonacci-Folge $(a_n)_{n\in\mathbb{N}}$,

$$a_0 := 1, \quad a_1 := 1, \quad a_n := a_{n-1} + a_{n-2} \quad \text{für } n \geq 2,$$

konvergiert gegen den goldenen Schnitt $(1+\sqrt{5})/2$. Insbesondere konvergiert die Differenz

$$b_n := \frac{a_{n+1}}{a_n} - \frac{a_n}{a_{n-1}}$$

gegen Null. Schreiben Sie eine Funktion `cauchy`, die zu gegebenem $k \in \mathbb{N}$ die kleinste Zahl $n \in \mathbb{N}$ mit $|b_n| \leq 1/k$ zurückgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das die Zahl $k \in \mathbb{N}$ einliest und den zugehörigen Index $n \in \mathbb{N}$ ausgibt.

Aufgabe 3.4. Was ist ein Gleitkommazahlsystem? Aus welchen Bestandteilen setzt sich eine Gleitkommazahl zusammen? Wie bestimmt man daraus ihren Wert? Was verbirgt sich hinter den Symbolen

`Inf`, `-Inf` und `NaN`? Was ist die Maschinengenauigkeit `eps`? Was ist ein implizites erstes Bit? Kann es auch zur Basis 7 ein implizites erstes Bit geben?

Aufgabe 3.5. Gegeben sei ein Polynom $p(x) = \sum_{j=0}^{n} a_j x^j$ in Form seines Koeffizientenvektors $a = (a_0, \ldots, a_n) \in \mathbb{R}^{n+1}$. Schreiben Sie eine Funktion `evalpolynomial`, die für gegebenen Koeffizientenvektor a und Auswertungspunkt x den Funktionswert $p(x)$ berechnet. Die Funktion `pow` zur Berechnung von x^j soll *nicht* verwendet werden. Schreiben Sie eine Funktion, die möglichst nur *eine* Schleife verwendet. Der Grad $n \in \mathbb{N}$ des Polynoms soll eine Konstante im Hauptprogramm sein, die Funktion `evalpolynomial` soll aber beliebigen Grad zulassen. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem die Koeffizienten a_j sowie der Auswertungspunkt x eingelesen werden und $p(x)$ ausgegeben wird.

Aufgabe 3.6. Schreiben Sie eine Funktion `kalender(jahr1)`, die das nächste Jahr `jahr2` $>$ `jahr1` berechnet, an dem man einen Kalender von Jahr `jahr1` vollständig wiederverwenden kann, d.h. die Wochentage der Jahre `jahr1` und `jahr2` stimmen vollständig überein.

Aufgabe 3.7. Schreiben Sie eine Funktion `minmaxmean`, die von einem gegebenem Vektor $x \in \mathbb{N}^n$ das Minimum, das Maximum und den Mittelwert $\frac{1}{n}\sum_{j=1}^{n} x_j$ berechnet und geeignet zurückgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das den Vektor $x \in \mathbb{N}^n$ einliest und Minimum, Maximum und Mittelwert ausgibt. Die Länge $n \in \mathbb{N}$ des Vektors soll eine Konstante im Hauptprogramm sein, die Funktion `minmaxmean` ist für beliebige Länge n programmieren.

Aufgabe 3.8. Schreiben Sie eine Funktion `wurzelschranke`, die zu einer gegebenen Zahl $x \geq 0$ die natürliche Zahl $k \in \mathbb{N}$ mit $k \leq \sqrt{x} < k+1$ zurückgibt. Dabei dürfen weder die Wurzel-Funktion `sqrt`, noch Rundungsoperationen (z.B. `floor` oder `ceil` etc.) verwendet werden. Schreiben Sie ferner ein aufrufendes Hauptprogramm das $x \in \mathbb{R}$ einliest und $k \in \mathbb{N}$ ausgibt.

Aufgabe 3.9. Ein Tripel $(x, y, z) \in \mathbb{N}^3$ natürlicher Zahlen heißt *pythagoräisches Zahlentripel*, falls $x^2+y^2 = z^2$ gilt. Das wohl bekannteste Beispiel ist $(3, 4, 5)$. Offensichtlich gelten $z > \max\{x, y\}$ sowie $x \neq y$ und ohne Beschränkung der Allgemeinheit ferner $x < y$. Schreiben Sie eine `void`-Funktion `pythagoras`, die zu gegebener Schranke $n \in \mathbb{N}$ alle pythagoräischen Zahlentripel mit $x < y < z \leq n$ bestimmt und ausgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem die Schranke n eingelesen und `pythagoras` aufgerufen wird.

Aufgabe 3.10. Für $p \in [1, \infty)$ ist die ℓ_p-Norm auf $\mathbb{R}^n$ definiert durch

$$\|x\|_p := \Big(\sum_{j=1}^{n} |x_j|^p\Big)^{1/p}.$$

Schreiben Sie eine Funktion `pnorm`, die einen Vektor $x \in \mathbb{R}^n$, dessen Länge n sowie $p \in [1, \infty)$ übernimmt und $\|x\|_p$ zurückgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem x und p eingelesen werden und $\|x\|_p$ ausgegeben wird. Die Dimension $n \in \mathbb{N}$ soll eine Konstante im Hauptprogramm sein, die Funktion `pnorm` soll aber beliebige Dimension zulassen. Testen Sie Ihr Programm mit verschiedenen Werten für p bei festem Vektor x. Was beobachten Sie für $p \to \infty$?