Dirk Praetorius, Samuel Ferraz-Leite
Christoph Überhuber

Sommersemester 2008
02. April 2008

Übungen zur Vorlesung
Einführung in das Programmieren für TM

Serie 2

Die Aufgaben mit Stern (*) sind bis zur nächsten Übung vorzubereiten und werden dort abgeprüft. Die übrigen Aufgaben dienen nur Ihrer Übung und mir als zusätzliche Grundlage für den Prüfungsstoff in den schriftlichen Tests. Kopieren Sie bitte den Source-Code auf Ihren Account auf der `cad.zserv.tuwien.ac.at` *in ein Unterverzeichnis* `serie02` *Ihres Home-Verzeichnisses. Überprüfen Sie bitte vor der Übung, ob Ihre Source-Codes mit dem* `gcc` *compiliert werden können. In den folgenden Aufgaben sollen* ***Verzweigungen*** *und* ***Zählschleifen*** *geübt werden.*

Aufgabe 11*. Man schreibe eine Funktion `vektorprodukt`, die zu gegebenen Vektoren $x, y \in \mathbb{R}^3$ das Vektorprodukt $z = x \times y$ mit

$$z_1 = x_2 y_3 - x_3 y_2$$
$$z_2 = x_3 y_1 - x_1 y_3$$
$$z_3 = x_1 y_2 - x_2 y_1$$

berechnet. Ferner schreibe man ein aufrufendes Hauptprogramm, das die Vektoren x und y einliest und $x \times y$ ausgibt. Speichern Sie den Source-Code `vektorprodukt.c` ins Verzeichnis `serie02`.

Aufgabe 12*. Man schreibe eine Prozedur `kurvendiskussion`, die für eine Parabel $p(x) = a_0 + a_1 x + a_2 x^2$ mit Koeffizientenvektor $a \in \mathbb{R}^3$ eine Kurvendiskussion durchführt. Wenn vorhanden, berechne man das Extremum (und Art) und die Nullstellen. Anderenfalls gebe man aus, dass die Parabel kein Extremum bzw. keine Nullstelle besitzt. Man schreibe ferner ein aufrufendes Hauptprogramm, das den Vektor a einliest und die Prozedur aufruft. Speichern Sie den Source-Code unter `kurvendiskussion.c` ins Verzeichnis `serie02`.

Aufgabe 13*. Gegeben sei ein Kreis in Form seines Mittelpunkts $(x, y) \in \mathbb{R}^2$ und seines Radius $r > 0$. Gegeben sei ferner ein Punkt $(u, v) \in \mathbb{R}^2$. Man schreibe eine Funktion `locate`, die zurückgibt, ob der Punkt (u, v) im Kreis (Rückgabe `-1`), auf der Kreislinie (Rückgabe `0`) oder außerhalb des Kreises (Rückgabe `1`) liegt. Ferner schreibe man ein aufrufendes Hauptprogramm, das die Zahlen $x, y, r, u, v \in \mathbb{R}$ einliest, die Funktion `locate` aufruft und danach in der Shell ausgibt, wie der Punkt (u, v) im Verhältnis zu Kreis (x, y, r) liegt.

Aufgabe 14*. Man schreibe eine Funktion `binomial`, die mittels *einer* geeigneten Schleife den Binomialkoeffizienten $\binom{n}{k}$ berechnet. Dazu realisiere man die gekürzte Form

$$\binom{n}{k} = \frac{n \cdot (n-1) \cdots (n-k+1)}{k \cdot (k-1) \cdots 1}.$$

Man schreibe ein aufrufendes Hauptprogramm, in dem $k, n \in \mathbb{N}_0$ mit $k \leq n$ eingelesen werden und $\binom{n}{k}$ ausgeben wird. Den Source-Code speichere man unter `binomial.c` ins Verzeichnis `serie02`. Weiters schreibe man eine Lösung mit *zwei* Schleifen, bei der Zähler und Nenner getrennt berechnet werden. Welche der zwei Implementierungen ist klüger und warum?

Aufgabe 15. Man schreibe eine Funktion `minabs`, die von einem gegebenem Vektor $x \in \mathbb{R}^n$ das Element x_k mit minimalem Absolutbetrag $|x_k| = \min_{j=1}^n |x_j|$ zurückgibt. Ferner schreibe man ein aufrufendes Hauptprogramm, das den Vektor x einliest und x_k ausgibt. Die Länge des Vektors soll eine Konstante im Hauptprogramm sein, die Funktion `minabs` ist für beliebige Länge n zu programmieren. Den Source-Code speichere man unter `minabs.c` ins Verzeichnis `serie02`.

Aufgabe 16. Man schreibe eine Funktion `skalarprodukt`, die zu gegebenen Vektoren $x, y \in \mathbb{R}^n$ das Skalarprodukt $x \cdot y := \sum_{j=1}^n x_j y_j$ berechnet. Ferner schreibe man ein aufrufendes Hauptprogramm, das die Vektoren x und y einliest und $x \cdot y$ ausgibt. Die Länge n der Vektoren soll eine Konstante im Hauptprogramm sein, die Funktion `skalarprodukt` ist für beliebige Länge n zu programmieren. Speichern Sie den Source-Code unter `skalarprodukt.c` ins Verzeichnis `serie02`.

Aufgabe 17. Gegeben seien drei Punkte (x, y), (u, v) und (a, b) in $\mathbb{R}^2$. Man schreibe eine Funktion `punkte`, die überprüft, ob die 3 Punkte auf einer Geraden liegen. Ferner schreibe man ein aufrufendes Hauptprogramm, in dem die 6 Parameter eingelesen werden und das Resultat ausgegeben wird.

Aufgabe 18. Gegeben seien zwei Geraden $f(x) = ax + b$ und $g(x) = cx + d$. Man schreibe eine Prozedur `schnittpunkt`, die überprüft, ob sich die beiden Geraden f und g schneiden, und die ggf. den Schnittpunkt berechnet. Das Ergebnis soll geeignet ausgegeben werden. Ferner schreibe man ein Hauptprogramm, in dem die Parameter $a, b, c, d \in \mathbb{R}$ eingelesen werden.

Aufgabe 19. Man schreibe eine Funktion `mean`, die den Mittelwert $\frac{1}{n} \sum_{j=1}^n x_j$ eines ganzzahligen Vektors $x \in \mathbb{Z}^n$ zurückgibt. Ferner schreibe man ein aufrufendes Hauptprogramm, das den Vektor x einliest und den Mittelwert ausgibt. Die Länge n des Vektors soll eine Konstante im Hauptprogramm sein, die Funktion `mean` ist für beliebige Länge zu programmieren.

Aufgabe 20. Man erweitere *Selection Sort* aus der Vorlesung um einen Parameter `type`, sodass die Funktion einen Vektor $x \in \mathbb{R}^n$ wahlweise aufsteigend (`type = 1`) oder absteigend (`type = -1`) sortiert. Man schreibe ein aufrufendes Hauptprogramm, in dem $x \in \mathbb{R}^n$ und die Sortierrichtung eingegeben werden und der sortierte Vektor ausgegeben wird. Die Länge n des Vektors soll eine Konstante im Hauptprogramm sein, die Funktion `sortvector` ist für beliebige Länge n zu programmieren.