

Übungen zur Vorlesung Einführung in das Programmieren für TM

Serie 1

Aufgabe 1.1*. Legen Sie in Ihrem Home-Verzeichnis auf der `lva.student.tuwien.ac.at` ein Unterverzeichnis `serie01` an. In dieses Verzeichnis kopieren Sie bitte den Source-Code der folgenden obligatorischen Programmieraufgaben, damit Sie in der Übung nicht lange suchen müssen. Machen Sie sich in diesem Zusammenhang die Verwendung des `scp`-Befehls klar.

Aufgabe 1.2*. Schreiben Sie ein Programm `kreis.c`, das aus gegebenem Radius r den Flächeninhalt des korrespondierenden Kreises ausgibt. Speichern Sie den Source-Code unter `kreis.c` in das Verzeichnis `serie01`.

Aufgabe 1.3*. Schreiben Sie ein `void`-Funktion `vektorprodukt`, die zu gegebenen Vektoren $\mathbf{u} = (a, b, c)^T$ und $\mathbf{v} = (x, y, z)^T$ das Vektorprodukt $\mathbf{w} = \mathbf{u} \times \mathbf{v}$ mit

$$w_1 = bz - cy$$

$$w_2 = cx - az$$

$$w_3 = ay - bx$$

berechnet und ausgibt. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem die Parameter a, b, c, x, y, z eingelesen und die Funktion aufgerufen werden. Speichern Sie den Source-Code unter `vektorprodukt.c` in das Verzeichnis `serie01`.

Aufgabe 1.4*. Ändern Sie das Programm aus Aufgabe 1.2 dahingehend, dass auch das Volumen eines Prismas mit einem Kreis als Grundfläche berechnet werden kann. Ändern Sie Ihren Code zunächst so ab, dass r über die Tastatur eingelesen werden kann. Zusätzlich soll eine zweite Größe h eingelesen werden, welche die Höhe des Prismas darstellt. Für $h \neq 0$ soll das Programm nun das Volumen des Prismas ausgeben. Im Fall $h = 0$ bleibt die Ausgabe, wie zuvor, der Flächeninhalt des Kreises. Speichern Sie den Source-Code unter `dreieck2.c` in das Verzeichnis `serie01`.

Aufgabe 1.5*. Schreiben Sie eine `void`-Funktion `sort3`, der drei Zahlen $x, y, z \in \mathbb{R}$ übergeben werden und die diese Zahlen fallend sortiert ausgibt, d.h. zuerst das Maximum $\max\{x, y, z\}$ und zuletzt das Minimum $\min\{x, y, z\}$. Schreiben Sie ferner ein aufrufendes Hauptprogramm in dem die Zahlen x, y, z eingelesen und die Funktion aufgerufen werden. Speichern Sie den Source-Code unter `sort3.c` in das Verzeichnis `serie01`.

Aufgabe 1.6. Was ist der Unterschied zwischen *lokalen* und *globalen* Variablen? Was könnten mögliche Vor- und Nachteile sein? Was meint man mit den Bezeichnungen *lifetime* und *scope*?

Aufgabe 1.7. Schreiben Sie eine `void`-Funktion `quadrant`, die für einen Punkt $(x, y) \in \mathbb{R}^2$ ausgibt, ob (x, y) auf einer der Achsen des Koordinatensystems liegt. Falls nicht, soll ausgegeben werden, in welchem

Quadranten (x, y) liegt. Schreiben Sie ferner ein Hauptprogramm, in dem $x, y \in \mathbb{R}$ eingelesen werden. Speichern Sie den Source-Code unter `quadrant.c` in das Verzeichnis `serie01`.

Aufgabe 1.8. Schreiben Sie eine Funktion `rundung`, die für eine gegebene Zahl $x \in \mathbb{R}$ die Zahl $n \in \mathbb{N}$ zurückliefert, die x am nächsten liegt. Die Zahl x , sowie die beiden Zahlen $n_1 \in \mathbb{N}$ und $n_2 \in \mathbb{N}$ mit $n_1 \leq x \leq n_2$ sollen hierbei über die Tastatur eingegeben werden. Falls x genau in der Mitte zwischen n_1 und n_2 liegt, werde die größere zurückgeliefert. Schreiben Sie ferner ein aufrufendes Hauptprogramm, das die Eingaben einliest und gerundet ausgibt. Speichern Sie den Source-Code unter `rundung.c` in das Verzeichnis `serie01`.

Aufgabe 1.9. Gegeben seien drei Punkte (x, y) , (u, v) und (a, b) in \mathbb{R}^2 . Schreiben Sie eine Funktion `punkte`, die überprüft, ob die 3 Punkte auf einer Geraden liegen. Schreiben Sie ferner ein aufrufendes Hauptprogramm, in dem die 6 Parameter eingelesen werden und das Resultat ausgegeben wird. Speichern Sie den Source-Code unter `punkte.c` in das Verzeichnis `serie01`.

Aufgabe 1.10. Schreiben Sie ein Programm welches eine Zahl $x \in \{0, 1, 2, 3, 4\}$ über die Tastatur einliest. Ihre Kommilitonen sollen nun die Zahl erraten. Dies ist über eine zweite Eingabe zu bewerkstelligen (Schaffen Sie es die vorherige Eingabe am Bildschirm zu löschen?). Sind die Eingaben gleich, so soll ein dickes Lob ausgegeben werden. Im anderen Fall ist an Lob nicht zu denken. Speichern Sie den Source-Code unter `guesswhat.c` in das Verzeichnis `serie01`.