

C-PROGRAMMIERKURS
WS 2011/2012

Bitte geben Sie Ihre Lösungen in Moodle ab, d.h. laden Sie Ihre C-Programme als .c-Datei bzw. Text — insbesondere die Ausgabe Ihres Programms — als .txt-Datei hoch. Ihre Dateien sollten im Dateinamen die Nummer der Aufgabe enthalten (Bsp. aufgabe02.c und aufgabe02.txt). Sie sollten Ihre Aufgaben zu zweit bearbeiten und abgeben. Schreiben Sie Ihre Namen, Matrikelnummern und das Datum oben als Kommentar in Ihre Programme. Sie können mehrere einzelne Dateien oder ein Archiv von Dateien abgeben.

Tipp: Testen Sie Ihre Abgaben. Sie dürfen Ihre zusätzlichen Testfälle gerne auskommentiert im Quelltext belassen.

Übungsblatt 6

Abgabe: **Freitag**, 25.11.2011 bis 09:15

Aufgabe 17 struct, Pointer, Funktionen, math.h (10 Punkte)

Geben Sie je Teilaufgabe eine kommentierte C-Quellcode-Datei ab. Ihr Vorgehen soll nachvollziehbar sein. Alle Programme sollen ihre Daten über die Kommandozeile bekommen, z.B. ./progname 3 4 5 und das Ergebnis ausgeben. Die geschriebenen Funktionen sollen nichts ausgeben, sondern nur die Ergebnisse zurückliefern.

1. Definieren Sie ein struct `dreieck`, das die Seitenlängen eines Dreiecks (a, b, c) aufnehmen kann.
2. Schreiben Sie eine Funktion mit der Signatur `double umfang (struct dreieck)`, welche aus der übergebenden Struktur den Umfang des Dreiecks berechnet.
3. Es ist eine Funktion zu erstellen, die einen übergebenen Pointer auf ein struct `dreieck` überprüft, ob das Dreieck rechtwinkelig ist.
4. Schreiben Sie eine Funktion `flaeche ()`, die ein Dreieck per Call-by-Reference übergeben bekommt, dessen Fläche bestimmt und diese zurückliefert.

Nutzen Sie die Funktion `double atof (const char *text)`. Beachten Sie auch die aus der Vorlesung bekannten mathematischen Funktionen.

Aufgabe 18 Arrays, Dynamischer Speicher, Datenstrukturen, Typecasts, Gehirnverknötungen (10 Punkte)

In dieser Aufgabe soll das Arbeiten mit Feldern und verketteten Listen mit dynamischer Speicherverwaltung geübt werden. Geben Sie je Teilaufgabe eine kommentierte C-Datei ab. Ihr Vorgehen soll nachvollziehbar sein. Die Daten sollen an die Programme über die Kommandozeile übergeben werden. Benutzen Sie ggf. Hilfsfunktionen um doppelten Code zu vermeiden.

1. Erstellen Sie je ein Programm, das über die Kommandozeile eine beliebige Menge von ganzen Zahlen einliest und in
 - a) ein dynamisch allokiertes Feld bzw.
 - b) eine verkettete Liste einliest.
2. Iterieren Sie in je einer eigenen Funktion über das obige Feld bzw. die Liste und berechnen Sie den Durchschnitt und lassen diesen ausgeben.
3. Schreiben Sie sich je eine Hilfsfunktion, welches jedes zweite Element aus dem Feld bzw. aus der Liste entfernt und anschliessend wieder den Durchschnitt der verbleibenden Werte bestimmt. Welche der verwendeten Datenstrukturen ist dafür besser geeignet und weshalb?