

Programmieren in C – Wintersemester 2019 / 20

Aufgabenblatt 1

Abgabe: Montag, 14.10.2019, 08:30 Uhr (schriftlich und elektronisch)

Hinweise

- Die Aufgaben sollen in Teams mit jeweils 2 Teilnehmern bearbeitet werden.
- Jede von Ihnen geschriebene Funktion muss mit Funktionskopf (Beschreibung der Funktionalität, Parameter und Rückgabewert) dokumentiert sein.
- Filename für das Hochladen: GruppeX_TeamY_BlattZ.pdf
(siehe auch die Hinweise in [Anmerkungen_Labor_C.pdf](#))
- **Achtung:** Zu spät eingereichte Lösungen gelten als nicht abgegeben. Gleiches gilt für nachträgliche Korrekturen.
- Kommentieren Sie Ihre abgegebenen Programme (siehe als weitere – aber nicht verpflichtende – Anregung auch die Hinweise in [PRG_WieSindProgrammeZuKommentieren.pdf](#))

Umfang der Ergebnisse (siehe [Anmerkungen_Labor_C.pdf](#)):

- Bei Programmieraufgaben: Laden Sie den Source Code zu den einzelnen Aufgaben hoch; der jeweilige Dateiname entspricht der Aufgabennummer
Beispiel: Source Code zu Aufgabe 1a. Steht in „Aufgabe1a.c“
ACHTUNG: NICHT „Aufgabe1a.c.txt“
- Zusätzlich: Kopieren Sie jeden Source Code in eine Datei (in der Reihenfolge der Aufgaben) und ergänzen Sie Texte, Interpretationen etc. (soweit in der Aufgabe gefordert); diese Gesamtlösungsdatei trägt den
Filnamen: GruppeX_TeamY_BlattZ.pdf

Aufgabe 1 (Pseudo-Programm zum Auswählen und Sortieren von Zahlen aus einer Liste):

Schreiben Sie für das in der Vorlesung beschriebene einfache Maschinenmodell eines Taschenrechners mit 4 Registern ein Pseudo-Programm (s. Vorlesung unter "Weiter verbesserte Version des Programms"), das folgende Aufgabe löst:

Aus einer gegebenen Liste mit Zahlen sollen die drei größten gefunden und in aufsteigend sortierter Reihenfolge ausgegeben werden.

Strukturieren und kommentieren Sie Ihr Programm, insbesondere die Verwendung der Register und evt. getroffene Annahmen über die Beschaffenheit der Liste.

Aufgabe 2 (Layout von Programmen):

Gegeben sei folgendes Programm (siehe auch Skriptordner, „Aufgabe2.c“):

```
/* ****
 * Projekt: Programmieren in C Dateiname: ??? Beschreibung: ???
 * Autor: Computerknacki Erstellt: Sept. 2011
 **** */
#include <stdio.h>
#include <stdlib.h> /* fuer system("PAUSE") */
int main(void) { int a, b; float erg; printf("Ein erstes Programm...\n");
printf("\n"); printf("Bitte die erste (ganze) Zahl
eingeben:\n"); scanf("%d", &a);
printf("Bitte die zweite (ganze) Zahl eingeben:\n"); scanf("%d",
&b); erg = a + b;
printf("%d + %d = %f\n", a, b, erg); erg = a - b; printf("%d - %d =
%f\n", a, b, erg);
erg = a * b; printf("%d * %d = %f\n", a, b, erg); erg = 1.0 * a /
b; printf("%d / %d = %f\n", a, b, erg);
system("PAUSE"); return 0; }
```

- a) Bringen Sie das Programm zum Laufen, d.h. kompilieren Sie es und führen es aus. Geben Sie die Programmausgabe an, wenn Sie die Zahlen 4 und 6 eingeben (in dieser Reihenfolge).

Hinweis: Wenn Sie unter Linux statt Windows arbeiten, sollten Sie die Zeile

```
system("PAUSE");
```

entfernen.

- b) Ohne die Befehle im einzelnen zu verstehen: Verbessern Sie die Lesbarkeit des Programms, indem Sie es durch Einfügen von Einrückungen, Zeilenumbrüchen, Leerzeichen und Leerzeilen strukturieren (Beispiel: siehe Vorlesung). Auf jeder Zeile sollte höchstens ein Befehl stehen. Kopieren Sie Ihre Lösung in die schriftliche Ausarbeitung und laden Sie das Programm elektronisch hoch.
- c) Beschreiben Sie kurz die Funktion des Programms.

Aufgabe 3 (Einfache printf-Anweisungen):

- a) Ergänzen Sie die folgende **printf**-Anweisung, damit auf dem Bildschirm die darunter gestellte Ausgabe erscheint:

```
#include <stdio.h>

int main(void)
{
    printf("\nDieser ...
}
```

Programmausgabe:

```
Dieser Text
enthaelt
viele
Zeilenumbrueche
```

Geben Sie zwei verschiedene Lösungen – inklusive der Programmausgabe – an.

- b) Was gibt das folgende Programm am Bildschirm aus? Versuchen Sie zunächst, die Aufgabe zu lösen, ohne das Programm laufen zu lassen, und überprüfen Sie Ihre Antwort anschließend.

```
#include <stdio.h>

int main(void)
{
    printf("Kommentare werden in C mit /*...*/ geklammert.\n");
    printf("Beachte jedoch:");
    printf("\n - Kommentare innerhalb von Anfuehrungszeichen\n");
    printf(" werden als Text und nicht als Kommentar interpretiert\n");
    /* printf("Ende\n"); */
    return(0);
}
```

Aufgabe 4 (Charly Brown als Bildschirmausgabe):

- a) Schreiben Sie ein Mini-Programm, das die folgende Zeichen nacheinander auf dem Bildschirm ausgibt (jeweils in einer neuen Zeile):

- (Punkt)
- (Minuszeichen)
- ' (Hochkomma)
- & (Kaufmanns-„Und“)
- / (Slash)
- \ (Backslash)
- _ (Unterstrich)
- | (Strich)
- ((Klammer auf)
-) (Klammer zu)
- " (Anführungsstriche)
- ` (einfacher Anführungsstrich)
- o (kleines o)

Dabei müssen Sie beachten, daß Sie zur Ausgabe einiger dieser Zeichen ein „\“ Backslash vor das Zeichen in der printf-Anweisung setzen müssen, da sie sonst nicht richtig interpretiert werden. Beispiel: Ausgabe der Anführungsstriche:

```
printf("\n\n");
```

Auf welche dieser Zeichen trifft dies zu?

- b) Schreiben Sie nunmehr ein Programm, das Charly Brown auf dem Bildschirm ausgibt. Die Lösung soll das Programm und die Programmausgabe umfassen.

