

Übungsblatt 9: Programmieren in C (WS 2019/20)

Abgabe: Montag, 20.01.20, 12:00

Zur Vorbereitung auf die Klausur finden Sie die Aufgaben der Klausuren aus früheren Semestern in der elektronischen Klausurausleihe der Fachschaft Informatik: <https://kai.cs.uni-kl.de>
Hinweis: Die Klausurausleihe ist nur innerhalb des Campusnetzes bzw. über VPN erreichbar!

Aufgabe 1 *Rekursive Funktionen verstehen*

Abgabe: rekursiv.txt (3 Punkte)

Gegeben sei folgende – mit Absicht unkommentiertes – C-Funktion:

```
int g(int m, int n)
{
    if (m == n)
        return m;
    if (m > n)
        return g(m-n, n);
    return g(m, n-m);
}
```

Was ist das Ergebnis bzw. der Effekt eines Aufrufs mit folgenden Parametern?

Geben Sie an, ob der Aufruf ein undefiniertes Verhalten hat (mit Erklärung!) oder andernfalls, was der Rückgabewert des Aufrufs ist!

Aufruf	Undefiniertes Verhalten bzw. Ergebnis
$g(64, 16)$	
$g(3, -7)$	
$g(48, 27)$	
$g(5, 0)$	
$g(0, 5)$	
$g(0, 0)$	

Aufgabe 2 *Rekursion und Iteration*

Abgabe: vergleich.c (8 Punkte)

Geben Sie für die folgenden rekursiven Funktion `f_rek` bzw. `g_rek` eine iterative Variante `f_it` bzw. `g_it` an! Bitte verwenden Sie dazu die Vorlage `vergleich.c` von der Vorlesungsseite und ergänzen Sie dort die Funktion.

```
int f_rek(int x)
{
    if (x < 0)
    {
        return 0;
    } else {
        return x + f_rek(x - 2);
    }
}

int g_rek(int *a, int n, int m, int x, int y)
{
    if (m < n) {
        return a[m] == x?
            g_rek(a,n,m+1,x,y+1) : g_rek(a,n,m+1,x,y);
    } else {
        return y;
    }
}
```

Aufgabe 3 *Ellipse*

Abgabe: ellipse.c (8 Punkte)

Gegeben sei folgende Grundstruktur eines C-Programm, das eine Funktion `summe()` enthält. Diese Funktion soll die Summe **beliebig vieler** Integer-Werte berechnen. Die aufzusummierenden Integer-Werte werden als Parameter übergeben, weshalb `summe()` eine variable Anzahl von Parametern benötigt. Um der Funktion mitzuteilen, dass die Liste zu Ende ist, wird als letzter Parameter eine 0 übergeben.

```
#include <stdio.h>
#include <stdarg.h>

int summe1(int zahl1, ... ) {
    int sum = 0;
    // hier ergaenzen!
    return sum;
}

int main(void) {
    printf("%d\n", summe1(12, 14, 16, 18, 0));
    printf("%d\n", summe1(88, 66, 42, 33, 27, 19, 0));

    return 0;
}
```

Ergänzen Sie die Implementierung der Funktion `summe()`!

Implementieren Sie außerdem eine weitere Funktion `summe2()`, welche die Anzahl der Zahlen, die aufsummiert werden sollen als Parameter enthält. Diese soll folgendermaßen verwendet werden:

```
printf("%d\n", summe2(4, 12, 14, 16, 18));
printf("%d\n", summe2(6, 88, 66, 42, 33, 27, 19));
```

Erläutern Sie, was passiert, wenn die Liste länger bzw. kürzer ist, als die Anzahl vorgibt!

Hinweis: Bitte fügen Sie die Erklärung als Kommentar zu `summe2` Ihrem Program hinzu.

Aufgabe 4 Sichtbarkeit von Variablen

Abgabe: sichtbar.txt (8 Punkte)

Gegeben sei folgendes C-Programm, bestehend aus zwei Modulen:

```
1 // Datei module1.c
2 #include "module2.h"
3 #include <stdio.h>
4 #include <stdlib.h>
5
6 int x = 2;
7 int main(void)
8 {
9     static int y = 3;
10    x = sub1(x);
11    x = sub1(x);
12    printf("main:%d %d\n", x, y);
13    x = sub2();
14    x = sub2();
15    printf("main:%d %d\n", x, y);
16    return 0;
17 }
```

```
1 //Datei module2.h
2 int sub1(int x);
3 int sub2(void);
```

```
1 // Datei module2.c
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 int sub1(int x);
6 int sub2(void);
7
8 extern int x;
9 int y = 8;
10
11 int sub1(int x)
12 {
13     int r;
14     static int y = 2;
15     r = x + y++;
16     printf("sub1: %d %d\n", x, y);
17     return r;
18 }
19
20 int sub2(void)
21 {
22     int r;
23     extern int y;
24     r = x + y++;
25     printf("sub2:%d %d\n", x, y);
26     return r;
27 }
```

- Auf welche Variablendefinitionen (nicht Variablendeklarationen) verweisen die Variablen `x` und `y` in den Anweisungen der Funktionen `sub1` (Zeile 15) und `sub2` (Zeile 24) in `module2.c`? Bitte geben Sie jeweils die Datei, die Zeilennummer sowie eine kurze Erklärung an!
- Was gibt das Programm aus? Bestimmen Sie die Ausgabe ohne das Programm auszuführen!¹

¹Dies ist eine Vorbereitung auf die Klausur: Dort steht Ihnen auch kein Rechner zur Verfügung!