

## Lösung Übungsblatt 3

Christoph van Heteren-Frese (Matr.-Nr.: 4465677), Julien Stengel

Tutor: Ruhland, eingereicht am 8. November 2013

---

### Aufgabe 4

a)

**Wieso haben die Variablen `a` und `b` zu unterschiedlichen Zeitpunkten verschiedene Werte?**

Die Variablen `a` und `b` wurden mehrfach deklariert; zum einen *global* in den Zeilen 3 und 4, sowie *lokal* in den Funktionen `diff` (Zeile 14 und 15) und `main` (Zeile 23 und 24). Die Ausgabe des Programms besteht aus fünf Zeilen und wird durch mehrere Aufrufe der Bibliotheksfunktion `printf` erzeugt:

- (1) Zeile eins der Ausgabe wird durch den Aufruf von `printf` in Zeile 27 erzeugt, der die Werte der zur Funktion `main` *lokalen* Variablen `a` und `b` ausgibt
- (2) Zeile zwei der Ausgabe wird durch den Aufruf von `printf` innerhalb der Funktion `summe` (Zeile 8) erzeugt, die die *global* deklarierten Variablen `a` und `b` ausgibt
- (3) Zeile drei der Ausgabe wird durch den Aufruf von `printf` in Zeile 28 erzeugt. Die *lokalen* Variablen `a` und `b` werden als Parameter an die Funktion `summe` übergeben, die die Summe der beiden Werte zurück gibt.
- (4) Zeile vier der Ausgabe wird durch den Aufruf von `printf` in Zeile 17 innerhalb der Funktion `diff` erzeugt. Die Werte der zur Funktion `diff` *lokalen* Variablen `a` und `b` werden hier ausgegeben.
- (5) Zeile fünf der Ausgabe wird durch den Aufruf von `printf` in Zeile 31 erzeugt, der den Wert der zur Funktion `main` *lokalen* Variable `c` ausgibt. Die Speicheradresse der Variable `c` ist zuvor durch den Aufruf in Zeile 30 als Parameter an die Funktion `diff` übergeben und dort in Folge der Zuweisung in Zeile 18 überschrieben worden.

**Welchen Rückgabewert hat die Funktion `diff()`?**

Die Funktion `diff` ist vom Typ `void` und hat somit keinen Rückgabewert.

**b)**

```
/* compiliert mit gcc (Ubuntu/Linaro 4.7.3-1ubuntu1) 4.7.3
*/

#include <stdio.h>

/* Definition von globalen Variablen.
 * Diese Variablen können überall im Programm verändert werden.
 */
int anz = 10;
int max = 0;
int array[10] = {4, 6, 2, 0, 9, 1, 5, 7, 8, 3};

/* Diese Funktion vergleicht und sortiert die beiden Feldeinträge
   "i1" und "i2" im Array "array".
 */
void exchange(int i1, int i2) {
    int tmp;
    if (array[i1] > array[i2]) {
        if (array[i1] > max)
            max = array[i1];
    } else {
        if (array[i2] > max)
            max = array[i2];
        tmp = array[i1];
        array[i1] = array[i2];
        array[i2] = tmp;
    }
}

int main()
{
    int i, j;
    for(i=0; i<anz; i++) {
        for(j=i+1; j<anz; j++)
            exchange(i, j);
    }
    printf("Die Zahlen in sortierter Reihenfolge:");
    for(i=0; i<anz; i++)
        printf(" %d", array[i]);

    printf("\nDas Maximum: %d\n", max);
    return 0;
}
```

c)

```
/* compiliert mit gcc (Ubuntu/Linaro 4.7.3-1ubuntu1) 4.7.3
 */

#include <stdio.h>

/* Diese Funktion vergleicht und sortiert die beiden Feldeinträge
   "i1" und "i2" im Array "array".
 */
void exchange(int i1, int i2, int array[], int * max) {
    int tmp;
    if (array[i1] > array[i2]) {
        if (array[i1] > *max)
            *max = array[i1];
    } else {
        if (array[i2] > *max)
            *max = array[i2];
        tmp = array[i1];
        array[i1] = array[i2];
        array[i2] = tmp;
    }
}

int main()
{
    int i, j;
    int max = 0;
    int anz = 10;
    int array[10] = {4, 6, 2, 0, 9, 1, 5, 7, 8, 3};

    for(i=0; i<anz; i++) {
        for(j=i+1; j<anz; j++)
            exchange(i, j, array, &max);
    }
    printf("Die Zahlen in sortierter Reihenfolge:");
    for(i=0; i<anz; i++)
        printf(" %d", array[i]);

    printf("\nDas Maximum: %d\n", max);
    return 0;
}
```