

GIT 2 Probeklausur SS 18 - Lösung

Dies ist keine offizielle Lösung von Herr Rieß!

André Kuhlmann - [KuhlTime/Informatik-Probeklausur](https://www.kuhlmann.de/KuhlTime/Informatik-Probeklausur)

Aufgabe 1:

1. Wie lautet der Befehl, um ein C-Programm "wurzel.c" zu kompilieren, unter Beachtung des ANSI-Standarts?
 - `gcc -ansi wurzel.c -o wurzel`
2. Was bewirkt der folgende Quellcode?
 - Die Variable `i` wird von 1 an dauerhaft hochgezählt und in die Konsole ausgegeben.
3. Nennen Sie die zwei fehlenden Bussysteme der „Von-Neumann-Rechnerarchitektur“:
 - Adressbus
 - **Datenbus**
 - **Steuerbus**
4. Was bewirkt die folgende Zeile?

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

- Mit dieser Codezeile kann die `stdio` Library, die Standard-Input-Output Library im folgenden Programm verwendet werden.
5. Was bedeutet die Werteübergabe mittels „call by value“?
 - Beim Aufrufen einer Funktion erhalten die Parameter der Funktion eine temporäre **Kopie** des Argumentwertes.
 6. Wofür steht die Abkürzung RAM?
 - Die Abkürzung RAM steht für Random-Access-Memory oder auf deutsch für den Arbeitsspeicher.
 7. Führen Sie jeweils eine bitweise Verknüpfung der angegebenen Zahlen über die logischen Operationen **AND**, **OR** und **XOR** durch. Geben Sie das Ergebnis in dezimaler Darstellung an.
Zahlen: **37, 89**

AND=1, OR=125, XOR=124

	2⁶	2⁵	2⁴	2³	2²	2¹	2⁰
	64	32	16	8	4	2	1
37	0	1	0	0	1	0	1
89	1	0	1	1	0	0	1
AND	0	0	0	0	0	0	1
OR	1	1	1	1	1	0	1
XOR	1	1	1	1	1	0	0

Aufgabe 2:

1. Welche Bedeutung haben die Zeilen 4-12?
Es werden zwei Funktionen definiert. `sub1` und `sub2`.
 - `sub1`: Die Funktion `sub1` nimmt einen Integer Wert als Übertragungsparameter, welchen die Funktion im inneren um einen Zähler erhöht und wieder zurück gibt.
 - `sub2`: Die zweite Funktion `sub2` nimmt einen pointer auf einen character und setzt den Pointer auf den Nullzeiger. Von der Funktion wird erwartet, dass ein integer Wert zurück gegeben wird, was allerdings nicht passiert.
2. Welche Auswirkung haben die Zeilen 21-28?
 - In diesen Zeilen wird der `m`'te Character des `String` Arrays mit einem Character mit der selben Platzierung von hinten getauscht.
 - `m` wird einen hochgezählt.
 - Dies geschieht solange `m < 3` ist.
3. Wie lauten die Ausgaben von Zeilen 30 und 31?
 - Zeile 30: `!t11o we1aH`
 - Zeile 31: `11`
4. Welche Auswirkung hat die Zeile 33 auf den Programmverlauf?
 - Die Zeile 33 hat keinen weiteren Einfluss auf das Programm, da die Variable `m` nicht weiter verwendet wird.
5. Wie lautet die Ausgabe von Zeile 37?
 - Zeile 37: `Ha11o`

Aufgabe 3:

1. Schreiben Sie ein Programm mit dem im Diagramm festgelegten Ablauf. Die Funktion `checkTemperatur()` brauchen Sie nicht erneut zu implementieren.

```
#include <stdio.h>

/* float checkTemperatur() */

int main() {
    float zielTemperatur = 25.0;
    int fanspeed = 3;
    int shutdown = 0;

    while(shutdown == 0) {
        float istTemperatur = checkTemperatur();
        printf("\nistTemp: %f", istTemperatur);

        if(istTemperatur < zielTemperatur) {
            fanspeed--;
        } else if (istTemperatur > zielTemperatur) {
            fanspeed++;
        }

        printf("\nFanspeed: %i", fanspeed);

        if(istTemperatur >= 70.0 || fanspeed > 10) {
            shutdown = 1;
        }
    }

    printf("System wird heruntergefahren...");
}
```

2. Beschreiben Sie stichpunktartig die Funktion des Programms und nennen Sie ein Beispiel wo ein solches Programm eingesetzt werden kann.
 - Das Programm kann als eine Steuerung für eine Ventilator betriebene Kühlung verwendet werden.
 - In dem Programm wird eine angestrebte `zielTemperatur` festgelegt. Welche das Programm durch Regelung der Lüftergeschwindigkeit versucht an zu peilen.
 - Ist die `istTemperatur` niedriger als die `zielTemperatur` so wird die Geschwindigkeit des Lüfters reduziert. Ist die `istTemperatur` jedoch größer als die `zielTemperatur` so wird die Geschwindigkeit des Lüfters hoch geregelt.
 - Sobald die `istTemperatur` über 70 steigt oder die `fanspeed` größer als 10 ist so wird das System heruntergefahren, wahrscheinlich als eine Art Überlastungsschutz.

Aufgabe 4

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

struct student {
    char name[10];
    float note;
};

void einlesen(struct student *neuerStudent) {
    printf("Nachname: ");
    scanf(" %s", &neuerStudent->name);
    printf("Note: ");
    scanf(" %f", &neuerStudent->note);
}

int main() {
    int anzahl, i;
    printf("Anzahl an Studenten: ");
    scanf(" %i", &anzahl);

    struct student *studenten = malloc(anzahl * sizeof(struct
student));

    for (i=0; i < anzahl; i++) {
        einlesen(&studenten[i]);
    }

    /* Beste Note finden */
    float besteNote = 5.0;
    for(i=0; i < anzahl; i++) {
        float aktuelleNote = studenten[i].note;
        if(aktuelleNote < besteNote) {
            besteNote = aktuelleNote;
        }
    }

    /* Studenten mit bester Note ausgeben */
    printf("\nBesten Schüler:\n");
    for(i=0; i < anzahl; i++) {
        if(studenten[i].note == besteNote) {
            printf("\n%s - %f", studenten[i].name,
studenten[i].note);
        }
    }
}
```