

# Informatik 1 für Regenerative Energien

## Klausur vom 5. Februar 2013

Jörn Loviscach

Versionsstand: 22. Februar 2013, 18:58



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Germany License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/3.0/de/> or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

*Drei Punkte pro Aufgabe. Mindestpunktzahl zum Bestehen: 15 Punkte. Hilfsmittel: maximal vier einseitig oder zwei beidseitig beschriftete DIN-A4-Spickzettel beliebigen Inhalts, möglichst selbst verfasst oder zusammengestellt; kein Skript, keine anderen Texte, kein Taschenrechner, kein Computer, kein Handy und Ähnliches.*

Name	Vorname	Matrikelnummer	E-Mail-Adresse

### Fingerübungen

1. Welches Bitmuster (acht Bit mit Zweierkomplement) steht für die Zahl 18? Welches für die Zahl  $-2$ ? Wie rechnet man dann  $18 - 2$  aus? Schreiben Sie das mit Bits hin.
2. Gegeben sind die C-Variablen `unsigned char a = 0xF2;` und `unsigned char b = 0x13;`. Geben Sie in Hexadezimal an, welche Werte die drei C-Ausdrücke `a&b` und `a|b` und `a^b` bei Rechnung mit acht Bit haben.
3. Geben Sie durch Klammern an, wie die Bedingung im `if` des folgenden C-Fragments ausgewertet wird. Was sind jeweils die Teilergebnisse?

```
bool a = true;
int b = 14;
int c = 42;
if(! a || b % 4 + 3 < 9 && c == 7)
{
    // ...
}
```

4. Die folgende C-Funktion soll die Zahl der Tage im Monat (1 = Januar, ..., 12 = Dezember) als Ergebnis liefern. Januar: 31 Tage, Februar: 28 oder im Schaltjahr 29 Tage, und dann weiter 31, 30, 31, 30, 31, 31, 30, 31, 30, 31. Die Nummer des Monats und die Information, ob es sich um ein Schaltjahr handelt, wird der Funktion übergeben. Es sind aber drei Fehler in der Funktion. Finden und korrigieren Sie diese.

```
int zahlDerTage(int monat, bool istSchaltjahr) // hat drei Fehler
{
    if(istSchaltjahr && monat = 2)
    {
        29;
    }
    int d[] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
    return d[monat];
}
```

5. Gegeben sind diese Datenstruktur:

```
struct Ort
{
    char ortsname[32];
    unsigned long postleitzahlVon;
    unsigned long postleitzahlBis;
};
typedef struct Ort Ort;
```

und dieses Array:

```
Ort ortsliste[] =
{
    {"Bielefeld", 33602, 33739},
    {"Guetersloh", 33334, 33335},
    // usw. durch alle Städte
    {"", -1, -1}
};
```

Schreiben Sie eine C-Funktion `bool pruefePLZ(Ort ortsliste[], char ortsname[], long plz)`, die als Ergebnis zurückgibt, ob die angegebene Postleitzahl `plz` zu dem angegebenen Ortsnamen `ortsname` passt. In `ortsliste` wird der Funktion das obige Array übergeben. Sie können `strcmp` aus `string.h` verwenden. Die Funktion `strcmp` liefert dann nur und nur dann den Wert 0 zurück, wenn die beiden Strings, die man ihr übergibt, gleich sind.

6. Schreiben Sie dieses C-Programmfragment mit `switch` statt mit `if`:

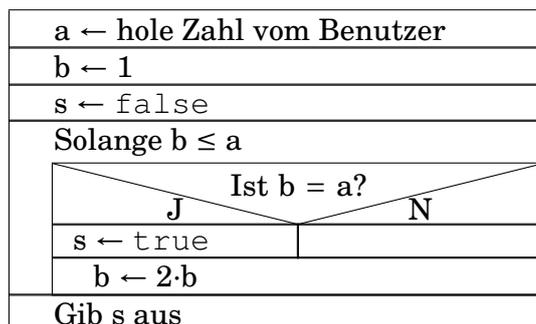
```

int x;
int y;
// ...
if(x >= 2 && x <= 4)
{
    y = x + 13;
}
else if(x == 10)
{
    y = x + 42;
}
else
{
    y = 7;
}

```

### Kreative Anwendung

7. Zeichnen Sie ein Flussdiagramm für das Programm, das durch dieses Struktogramm beschrieben wird:



8. Schreiben Sie eine Funktion `bool pruefePasswort(char s[])`, die eine nullterminierte C-Zeichenkette entgegen nimmt und prüft, ob mindestens ein Buchstabe A...Z und mindestens eine Ziffer 0...9 in der Zeichenkette enthalten ist. Wenn das der Fall ist, soll die Funktion `true` zurückliefern, sonst `false`. Beispiel: Mit der Zeichenkette "1234" aufgerufen, soll die Funktion den Wert `false` zurückliefern, ebenso für "abcABC". Für "4ab2C" liefert sie dagegen `true` zurück. Achtung: Die Länge der Zeichenkette wird nicht gesondert übergeben. Die Funktion `strlen` ist erlaubt.
9. Ein Algorithmus erhält eine Liste von  $n$  Elementen und soll überprüfen, ob die Liste jedes Element nur einmal enthält. Dazu vergleicht er das erste Element mit den zweiten, dem dritten, usw. bis zum  $n$ -ten. Dann vergleicht er das zweite Element mit dem dritten, dem vierten usw. bis zum  $n$ -ten. Und so weiter. Ist die Laufzeit  $O(n)$  und/oder ist sie  $O(n^2)$  und/oder ist sie  $O(n^3)$ ?

10. Welche Werte stehen nach Ausführung dieser Zeilen C-Code in den Variablen u, v, w?

```
int x = 42;
int y[7] = {3, 4};
int *z = &x;
(*z)++;
int u = z[0];
int v = *(y + 1);
z = y + 2;
int w = z[0];
```

11. Diese Funktion ist gegeben:

```
int f(int x)
{
    if(x <= 1)
    {
        return 42;
    }
    return x + f(x - 2);
}
```

Was steht nach dem Aufruf `int a = f(3);` in der Variablen a? Geben Sie möglichst auch Zwischenschritte an, damit Ihr Gedankengang nachvollziehbar ist.

12. Schreiben Sie diese C-Funktion so um, dass man sofort sehen kann, was sie tut. Geben Sie möglichst auch Zwischenschritte an, damit Ihr Gedankengang nachvollziehbar ist.

```
unsigned int f(unsigned int x)
{
    while(x < 7)
    {
        x += 2;
    }
    while(x > 8)
    {
        x -= 2;
    }
    return x;
}
```