

# Informatik 1 für Regenerative Energien

## Klausur vom 24. September 2012

Jörn Loviscach

Versionsstand: 5. Oktober 2012, 20:13



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  $-11$ ? Welches für die Zahl  $12$ ? Wie rechnet man dann  $-11 + 12$  aus? Schreiben Sie das mit Bits hin.
2. Gegeben sind die C-Variablen `unsigned char a = 0xA1;` und `unsigned char b = 0x7B;`. Geben Sie in Hexadezimal an, welche Werte die drei C-Ausdrücke `~a` 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?

```
int a = 13;
int b = 14.0;
bool c = false;
if(a % 5 > 2.0 * b || a * 2 == 26 && ! c)
{
    // ...
}
```

4. Die folgende C-Funktion soll zählen, durch wie viele Zahlen *über* 1 die übergebene Zahl teilbar ist. Als Eingabe sind nur Zahlen ab 1 aufwärts erlaubt. Beispiele: Für 5 und für 13 soll sie 1 zurückliefern, weil diese Zahlen nur durch sich selbst teilbar sind; für 10 soll sie 3 zurückliefern (Teiler 2, 5, 10). Es sind aber drei Fehler in der Funktion. Finden und korrigieren Sie diese.

```
int countFactors(unsigned int x) // hat drei Fehler
{
    int c = 0;
    for(unsigned int t = 2; t < x; t++)
    {
        if(x / t == 0)
        {
            c++;
        }
    }
    return t;
}
```

5. Gegeben sei die Datenstruktur Buch:

```
struct Buch
{
    char Titel[64];
    char Autor[32];
    int AnzahlExemplare;
    char Standort[16];
};
typedef struct Buch Buch;
```

Schreiben Sie eine C-Funktion `int zaehleBuecher(Buch buecher[], int anzahl, char autor[])`, die als Ergebnis eine ganze Zahl zurückgibt – nämlich, wie viele Exemplare (Exemplare!) der Bücher im Array `buecher` von diesem Autor stammen. In `anzahl` wird der Funktion die Anzahl der Einträge des Arrays `buecher` 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 `if` statt mit `switch`:

```
int x;
int y;
// ...
switch(y)
{
case 3:
case 4:
    x = y;
case 11:
    x++;
    break;
case 12:
    x += y;
default:
    x += 7;
}
```

### Kreative Anwendung

7. Ein Programm soll so lange (positive) Zahlen vom Benutzer abfragen, bis der die Zahl 0 eingibt. Dann soll das Programm ausgeben, ob die Zahlen in gleichmäßigen Abständen waren. Beispiele: Bei der Eingabe 3, 13, 7, 3 soll es „Nein!“ ausgeben; bei der Eingabe 5, 7, 9, 11 ebenso wie bei der Eingabe 4, 8, 12 soll es „Ja!“ ausgeben. Lösen Sie das mit einem Struktogramm, das Eingabe, Ausgabe, Zuweisung, Rechenoperationen, Vergleich und Schleife benutzt. Gehen Sie davon aus, dass der Benutzer mindestens drei andere Zahlen eingibt, bevor er eine 0 eingibt.
8. Schreiben Sie eine C-Funktion `int numberFromBinary(char b[])`, die einen nullterminierten C-String aus Symbolen 0 und 1 entgegen nimmt, als Binärzahl auffasst und den Wert als Zahl zurück liefert. Beispiel: Mit der Zeichenkette "10010" aufgerufen, soll die Funktion den Wert 18 zurückliefern. Achtung: Die Länge der Zeichenkette wird nicht gesondert übergeben. Die Funktion `strlen` ist erlaubt.
9. Ein Algorithmus habe bei einer Eingabegröße von  $n$  die Laufzeit  $\frac{n^3 + \sqrt{n}}{13 + n + 2 \sin(n)}$ . Ist das asymptotisch  $O(n)$  und/oder  $O(n^2)$  und/oder  $O(n^3)$ ?

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

```
int x[] = {3, 4, 5};
int y = 6;
int *z = &y;
int u = z[0];
y = 7;
int v = *z;
z = x;
z++;
int w = z[1];
```

11. Welcher Wert steht nach Ausführung des folgenden C-Code auf einem 16-Bit-Rechner in der Variablen z? (Geben Sie möglichst auch Zwischenschritte an, damit Ihr Gedankengang nachvollziehbar ist.)

```
unsigned int x = 16 * 256 + 1;
unsigned int z = x * 16;
```

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.)

```
int f(int x)
{
    int a = x - 2;
    if(x < 3)
    {
        a--;
        return a;
    }
    int b = 0;
    for(int i = 3; i < x; i++)
    {
        b++;
    }
    return b;
}
```