

Informatik 1 für Regenerative Energien

Probeklausur 1

Jörn Loviscach

Versionsstand: 8. März 2011, 17:24



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, falls nicht in Mailingliste

Fingerübungen

1. Welches Bitmuster (acht Bit mit Zweierkomplement) steht für die Zahl 3? Welches für die Zahl -5 ? Wie rechnet man dann $3 + (-5)$ aus? Schreiben Sie das mit Bits hin.^{c1}
2. Gegeben sind die C-Variablen `unsigned char a = 0x12;` und `unsigned char b = 0x2D;`. 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 = 42;
bool b = true;
if(a + 3 * 4 > 13 && ! b || a < 60 && a != 30)
{
    // ...
}
```

4. Die folgende C-Funktion soll für eine eingegebene Zahl $n \geq 0$ die Fakultät $n! = n \cdot (n-1) \cdot \dots \cdot 3 \cdot 2 \cdot 1$ bzw. $0! = 1$ berechnen. Es sind aber drei Fehler darin. Finden und korrigieren Sie diese.

^{c1}jl: Aufgabe von -4 auf -5 geändert.

```
int factorial(int n) // fehlerhaft
{
    int result = 0;
    for(int i = 2; i < n; i++)
    {
        result *= i;
    }
}
```

5. Gegeben sei die Datenstruktur `Person`:

```
struct Person
{
    char name[8]; // Nachname
    int age; // Alter
};
typedef struct Person Person;
```

Schreiben Sie eine C-Funktion

`void printAdults(int number, Person persons[])`, die mit Hilfe von `puts` die Nachnamen aller Personen mit einem Alter von 18 oder mehr ausgibt. In `number` wird der Funktion die Anzahl der Personen in der Liste `persons` übergeben.

6. Folgende Initialisierung eines C-Arrays des Datentyps `Person` aus der vorigen Aufgabe hat drei Fehler. Finden und korrigieren Sie diese.

```
// fehlerhaft
Person myFriends =
{
    {"Berta", 42};
    {"Karlheinz", 43}
};
```

Kreative Anwendung

7. Ein Programm soll zu einer eingegebenen Zahl $n = 0, 1, 2, \dots$ die Potenz 3^n berechnen und ausgeben. Lösen Sie das mit einem Flussdiagramm, das Eingabe, Ausgabe, Zuweisung, Addition, Multiplikation und Vergleich/Verzweigung benutzt.
8. Die Initialisierung aus Aufgabe 6 sei korrigiert. Geben Sie nun drei Möglichkeiten an, die Fragezeichen in dem folgenden C-Fragment so zu ersetzen, dass das Programm in den Rumpf des `if` springt. (2+2 statt 4 zu schreiben, gilt dabei nicht als neue Möglichkeit.)

```
if(myFriends??? == 'a')
{
    // ...
}
```

9. Folgende C-Funktion liefert auf einem 16-Bit-Rechner bei dem Aufruf `product(130, 1000)` das Ergebnis `-1072`. Was passiert hier? Wie könnte man die Funktion ändern, um das zu verhindern?

```
int product(int a, int b)
{
    return a * b;
}
```

10. Das folgende C-Programmfragment soll die Datenstruktur „Stapel“ für `int` implementieren. Schreiben Sie `hasNext` und `pop`.

```
#define maximumNumberOfElements 8
int stack[maximumNumberOfElements];
int count = 0;

void push(int x)
{
    if(count < maximumNumberOfElements)
    {
        stack[count] = x;
        count++;
    }
}

bool hasNext(void)
{ ??? }

int pop(void)
{ ??? }
```

11. Ein Algorithmus habe bei einer Eingabegröße von n die Laufzeit $13+3n+\frac{1}{n^2+2}$. Ist das asymptotisch $O(n)$ und/oder $O(n^2)$ und/oder $O(n^3)$?
12. Schreiben Sie diese C-Funktion so um, dass man sofort sehen kann, was sie tut.

```
int f(int x)
{
    int y = 42;
    for(int i = -1; i < 2; i += 2)
    {
        y = i*x;
        if(y >= 0)
        {
            break;
        }
    }
    return y;
}
```