

Informatik 1 für Regenerative Energien

Klausur vom 7. Juli 2014

Jörn Loviscach

Versionsstand: 7. Juli 2014, 15:09



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 13? Welches für die Zahl -6 ? Wie rechnet man dann $13 + (-6)$ aus? Schreiben Sie das mit Bits hin.
2. Gegeben sind die C-Variablen `unsigned char a = 0xA1;` und `unsigned char b = 0x35;`. 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 = 101;
bool c = true;
if(a % 10 != 3 || ! c && b / 100 <= 1)
{
    // ...
}
```

4. Die folgende C-Funktion soll feststellen, ob die übergebene Zahl eine „Schnapszahl“ ist, also in Zehnersystem lauter gleiche Ziffern hat wie 55 oder 3333. Einstellige Zahlen sind nie Schnapszahlen. Es sind aber drei Fehler in der Funktion. Finden und korrigieren Sie diese.

```
bool istSchnapszahl(unsigned int z)
{
    if(z > 10)
    {
        return false;
    }

    int letzteZiffer = z / 10;
    while(z > 0)
    {
        if(z % 10 == letzteZiffer)
        {
            return false;
        }
        z /= 10;
    }
    return true;
}
```

5. Gegeben ist die Datenstruktur `Punkt` für einen Punkt in der Ebene:

```
struct Punkt
{
    double x;
    double y;
};
typedef struct Punkt Punkt;
```

Schreiben Sie eine C-Funktion, die ein Array solcher Punkte sowie deren Anzahl entgegennimmt und zurückgibt, wie weit die übergebenen Punkte maximal vom Ursprung entfernt sind. Hinweise: Der Punkt $(a_x|a_y)$ hat vom Ursprung den Abstand $\sqrt{a_x \cdot a_x + a_y \cdot a_y}$. Die Quadratwurzelfunktion in C ist `sqrt` aus `<math.h>`.

6. Ein Programm soll in einer Endlosschleife ganze Zahlen vom Benutzer abfragen und nach jeder Eingabe ausgeben, wie viele gleiche Zahlen unmittelbar hintereinander eingegeben worden sind. Beispiel:

```
Eingabe  7  7  7  2  7  7  3  ...
Ausgabe  1  2  3  1  1  2  1  ...
```

Lösen Sie das mit einem Flussdiagramm. Dieses darf Eingabe, Ausgabe, Zuweisung, Vergleich, Addition, Verzweigung benutzen. (Ignorieren Sie etwaige Probleme, die durch Überlauf entstehen könnten.)

Kreative Anwendung

7. Schreiben Sie eine Funktion `bool pruefeEMailAdresse(char s[])`, die eine nullterminierte C-Zeichenkette entgegen nimmt und prüft, ob genau einmal das Zeichen `@` enthalten ist und ob in den Zeichen danach mindestens ein Punkt vorkommt. Beispiel: Mit der Zeichenkette "bla" aufgerufen, soll die Funktion den Wert `false` zurückliefern, ebenso für "bl@" und "bl@xyz" und "bl@xy@z.de". Für "bl@." und "bl@xy.de" und "bl@.de" liefert sie dagegen `true` zurück. Achtung: Die Länge der Zeichenkette wird nicht gesondert übergeben. Die Funktion `strlen` ist erlaubt.
8. Ersetzen Sie die `if` und `else` in diesem Stück C-Code durch eine Konstruktion mit einem einzigen `switch`, die für alle Werte von `x` und `y` das jeweils gleiche Ergebnis liefert wie der Code mit `if` und `else`:

```
int x;
int y;
// ...
if(x != 1)
{
    if(x == 5)
    {
        y = 7;
    }
    else if(x == 9)
    {
        y++;
    }
}
else if(x == 3)
{
    y = 2;
}
```

9. Die folgende C-Funktion wird direkt nach dem Programmstart mit `f(4)` ausgeführt. Welcher Zahlenwert steht danach in der Variablen `a`?

```
unsigned int f(unsigned int x)
{
    static int a = 0;
    a++;

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

10. Welche Werte stehen nach Ausführung dieser Zeilen C-Code in den Variablen a, b, c?

```
int u[3] = {5, 6};
int v = 100;
int *w = &v;
int *x = u;
x++;
*w = 7;
int a = v;
int b = u[2];
int c = x[0];
```

11. Welche Zahlenwerte (dezimal angegeben, mit Vorzeichen) stehen nach Ausführung des folgenden C-Codes auf einem 16-Bit-Rechner in den Variablen y und z? Geben Sie möglichst auch Zwischenschritte an, damit Ihr Gedankengang nachvollziehbar ist.

```
int x = 257;
int y = 256 * x;
int z = 128 * 256 / 30000;
```

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 <= 12)
    {
        x++;
    }
    int y;
    for(y = x; y >= 14; y--)
    {
        x--;
    }
    return y;
}
```