

1 Testataufgaben

1.1 Integer Arithmetik

(3 Punkte)

Kompilieren Sie `blatt02_1.c` (\rightarrow StudIP) und starten Sie das Programm mithilfe des beiliegenden `Makefile` (\rightarrow StudIP). Kommentieren Sie jede Ausgabezeile ausführlich mit Hinblick auf die in der Vorlesung vorgestellten Konzepte *Zweierkomplement*, *wrap around* und *implizite Typumwandlung*¹. Erklären Sie die Ausgaben ihrem Tutor.

1.2 Binärdarstellung von Integern

(4 Punkte)

Schreiben Sie ein Programm, das die Binärdarstellung eines vorzeichenlosen 16-Bit Integers (`uint16_t`) in der Konsole ausgibt. Testen Sie insbesondere die Randfälle² des Wertebereichs von `uint16_t`.

1.3 Zahlen-Raute

(3 Punkte)

Schreiben Sie ein Programm, das mithilfe von verschachtelten Schleifen eine Zahlen-Raute beliebiger Größe $0 < N < 10$ in der Konsole ausgibt. Der Parameter N soll im Programm als lokale Variable definiert und initialisiert werden³. Folgende Abbildung ist die Ausgabe für den Fall $N = 4$.

```

    1
  1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
  1 2 3 2 1
    1 2 1
      1
```

¹http://openbook.rheinwerk-verlag.de/c_von_a_bis_z/007_c_typumwandlung_001.htm

²http://openbook.rheinwerk-verlag.de/c_von_a_bis_z/005_c_basisdatentypen_001.htm
(Kapitel 5.1 – 5.7 und 5.16 – 5.17)

³Eine Benutzereingabe (z.B. mit `scanf`) ist nicht gefordert!

2 Präsenzaufgaben

1. Schreiben Sie ein Programm, das eine Umrechnungstabelle Celsius ($^{\circ}\text{C}$) – Kelvin (K) ausgibt. Für die Umrechnung gelte die Regel: $-273^{\circ}\text{C} = 0\text{ K}$ und $2^{\circ}\text{C} - 1^{\circ}\text{C} = 2\text{ K} - 1\text{ K}$. Die Tabelle soll bei 0 K beginnen und bei 550 K enden. Der Abstand zwischen den Tabelleneinträgen soll 5 K betragen.
2. Schreiben Sie ein Programm, das eine Tabelle mit den Funktionswerten der Normalparabel zu den x -Werten aus dem Intervall $[-5, \dots, 5]$ ausgibt. Dafür deklarieren Sie zunächst zwei Variablen x und y vom Typ `float`. Dann angefangen mit $x = -5$ erhöhen Sie x in jedem Schleifendurchlauf um 0.125 und berechnen den zugehörigen y -Wert. Anschließend geben Sie beide Werte auf dem Bildschirm aus. Verwenden Sie für die Abbruchbedingung einen Vergleich mit `<=`, da bei Fließkommazahlen ein Vergleich mit `==` stets gefährlich ist.
3. Schreiben Sie ein Programm, das die Werte von drei `int`-Variablen a, b, c aufsteigend sortiert und auf dem Bildschirm ausgibt. Verwenden Sie nur die `if`-Anweisung als Kontrollstruktur.
4. Schreiben Sie ein Programm, das zu einer vorgegebenen natürlichen Zahl alle Teiler (außer der 1) auf dem Bildschirm ausgibt. Um den Rest einer Integer-Division zu bestimmen darf der Modulo-Operator `%` verwendet werden.
Beispiel: Da $12 \% 6 == 12 \% 4 == 12 \% 3 == 12 \% 2 == 0$ gilt, sind $\{6, 4, 3, 2\}$ Teiler von 12.
5. Schreiben Sie ein Programm, das alle durch 3 teilbaren Zahlen zwischen 1 und 100 ausgibt. Eine Zahl ist durch 3 teilbar, wenn der Rest der Integer-Division gleich Null ist.
Beispiel: $51 \% 3 == 0$ und $53 \% 3 == 2$.
6. Schreiben Sie ein Programm, das zu einer vorgegebenen positiven Integer-Zahl alle positiven Integer-Zahlen bestimmt, die bei Addition mit der Zahl nicht zu einem “wrap-around” führen.
7. Schreiben Sie ein Programm, das unter Verwendung verschachtelter Schleifen folgende Ausgabe produziert:

```
*****
*****
****
***
**
*
```