4.8 Vermischte Übungen - Lösungen

Aufgabe 9: *Variablen*

a) Nenne Vorteile, die für die Verwendung von Variablen sprechen.
b) Nenne drei Datentypen, die in Variablen abgespeichert werden können.

Lösung:
Vorteile:

- Das Programm wird lesbarer und besser verständlich.
- Wenn der Wert einer Variablen geändert werden muss, kann dies an einer zentralen Stelle geschehen statt an vielen Stellen.
- Mit Variablen lässt sich rechnen - dadurch lassen sich effiziente Programmierstrukturen wie Schleifen nutzen.

Wir unterscheiden die Datentypen *Zahl*, *Zeichen* und *Wahrheitswert*.

Aufgabe 10: *Schleifen*
An allen Digitalpins des Arduino werden LEDs mit geeigneten Vorwiderständen angeschlossen. Dann wird das rechts abgebildete Programm ausgeführt.

a) Erstelle eine Trace-Tabelle für einen Durchlauf der `Wiederhole fortlaufend`- Schleife.
b) Nenne die Pin-Nummern der LEDs, die nach Durchlaufen dieses Programms einmal geleuchtet haben.
c) Stelle das Programm als Struktogramm dar.

Lösung:
Die Zeilenzählung soll bei „wiederhole fortlaufend“ mit 1 beginnen.

Zeile	pin
2	2
3	2
4	2
5	2
6	2
7	4

Zeile	pin
3	4
4	4
5	4
6	4
7	6
3	6

Zeile	pin
4	6
5	6
6	6
7	8
3	8
4	8

Zeile	pin
5	8
6	8
7	10
3	10
4	10
5	10

Zeile	pin
6	10
7	12
3	12
2	2
...	...

Aus der Trace-Tabelle geht hervor, dass die Pins mit den Nummern 2, 4, 6, 8 und 10 nach einem Durchlauf der `wiederhole fortlaufend`-Schleife einmal geleuchtet haben.
Das Struktogramm sieht im wesentlichen genauso aus wie in der Block-Darstellung von mBlock.

Aufgabe 11: *Bitübertragung*
Ein Programm ist 30 kB groß. Berechne, wie lange es bei einer Bitrate von 115200 Bit pro Sekunde dauert, das Programm auf den Arduino zu übertragen.

Lösung:

$$30\,\text{kB} = 30000\,\text{B} = 240000\,\text{bit}$$
$$240000\,\text{bit}/115200\,\frac{\text{bit}}{\text{s}} \approx 2\,\text{s}$$

Die Übertragung dauert ca. 2 Sekunden.

Aufgabe 12: *Das Binärsystem und das Dezimalsystem*

a) Übersetze vom Binär- ins Dezimalsystem.

(1) 1001_2 (2) 1010_2 (3) 1111_2

b) Übersetze vom Dezimal- ins Binärsystem.

(1) 11 (2) 7 (3) 14

Lösung:

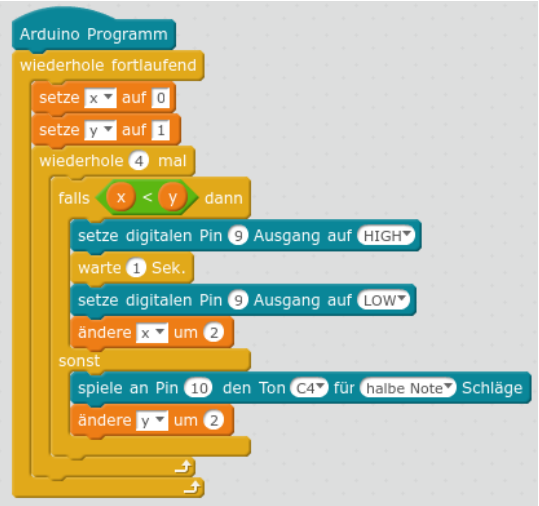
a) (1) $1{\cdot}8{+}0{\cdot}4{+}0{\cdot}2{+}1{\cdot}1 = 9$ (2) $1{\cdot}8{+}0{\cdot}4{+}1{\cdot}2{+}0{\cdot}1 = 10$ (3) $1{\cdot}8{+}1{\cdot}4{+}1{\cdot}2{+}1{\cdot}1 = 15$

b) (1) $11 = 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 = 1011_2$
(2) $7 = 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 = 111_2$
(3) $14 = 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 0 \cdot 1 = 1110_2$

Aufgabe 13: *Programme verstehen*

Am Arduino wird an Pin 9 eine LED mit Vorwiderstand und an Pin 10 ein Piezo-Summer angebracht.

a) Stelle das Programm als Struktogramm dar.
b) Beschreibe die Wirkung des rechts abgebildeten Programms.
c) Erkläre, wie man das Programm ändern müsste, damit die LED zwei Mal blinkt, bevor wieder der Piezo-Summer piept.



Lösung:

Das Struktogramm kann selbst erstellt werden.

Das Programm bewirkt, dass abwechselnd die LED blinkt und der Piezo-Summer piept.

Man kann auf mehrere Arten erreichen, dass die LED zwei Mal blinkt, bevor der Piezo-Summer wieder piept:

- Das Blinken der LED wird in eine `wiederhole 2 mal`-Schleife gepackt.
- x wird immer um 1 geändert statt um 2. Dafür muss die Bedingung $x \leq y$ heißen (im Programm geht das über `x < y` `oder` `x=y`). Die Anzahl der Wiederholungen sollte dann nicht 4, sondern 3 oder 6 oder ein anderes Vielfaches von 3 sein.
- ...

Aufgabe 14: *Programm entwickeln*

An allen Digitalpins des Arduino sind LEDs mit Vorwiderstand angeschlossen. Für das folgende Programm ist bereits eine Variable namens `p` angelegt.

Entwickle ein Programm, das die LEDs an Pin 1 bis 5 [2,4,6,8] nacheinander zum Leuchten bringt und nach einer Sekunde wieder ausschaltet. Es leuchtet also immer nur eine LED zur selben Zeit.

Anforderungen:

- Das Programm soll als Struktogramm dargestellt werden.
- Es sollen so wenig Code-Wiederholungen wie möglich vorkommen (*effizientes Programmieren*).

Befehle:

Lösung:

Für die LEDs an Pin 1 bis 5:

- wiederhole fortlaufend
  - setze p auf 1
  - wiederhole 5 mal
    - setze Pin p auf HIGH
    - warte 1 Sekunde
    - setze Pin p auf LOW
    - ändere p um 1

Alternativ:

- wiederhole fortlaufend
  - setze p auf 1
  - wiederhole bis p = 6
    - setze Pin p auf HIGH
    - warte 1 Sekunde
    - setze Pin p auf LOW
    - ändere p um 1

Für die LEDs 2, 4, 6, 8:

- wiederhole fortlaufend
  - setze p auf 2
  - wiederhole 4 mal
    - setze Pin p auf HIGH
    - warte 1 Sekunde
    - setze Pin p auf LOW
    - ändere p um 2