

# Lösungen

## Arbeitsblatt 9 - Variablen

**Aufgabe 1:** Beschreibe, was eine Variable ist und welche Eigenschaften sie hat.

Eine Variable ist ein Speicherort für Zahlen, genauer gesagt für Zahlen, Zeichen oder sonstige Daten.

Eine Variable hat einen Namen und einen Datentyp.

**Aufgabe 2:** Beschreibe, was das Programm von Blatt 1 macht.

In Zeile 5 wird eine Integer Variable namens temperatur deklariert und mit dem Wert 22 initialisiert. Je nach Wert der Variablen werden beim

Bob die Augen in unterschiedlichen Farben eingeschaltet.

Mit dem aktuellen Wert 22 werden beide Augen in orange eingeschaltet. Bei einem Wert der Variablen von z.B. 30 würden die Augen rot leuchten, bei dem Wert 3 würden sie blau leuchten.

**Aufgabe 3:** Warum heisst eine Variable ‚Variable‘? Erläutere deine Antwort!

Eine Variable heisst Variable, da man ihren Wert verändern kann, dieser also variabel ist.

Zu beachten ist, dass der Datentyp immer gleich bleiben muss!

**Aufgabe 4:** Betrachte das folgende Beispiel. Welchen Wert hat die Variable **temp** nach der Zeile 7?

```

5   int temp = 5;
6   temp = 8;
7   temp = temp + 4;
    
```

**Aufgabe 5:** Welche Programm-Codes sind korrekt und würden compilieren? Kreuze die richtigen Antworten an, es sind mehrere Antworten möglich:

- `int temperatur == 22;`
- `int Temperatur = 22;`
- `int temperatur = 22;`
- `int temp = 22,`
- `int temp = 22;`
- `int temp22 = 22;`
- `integer temperatur = 22;`
- `Integer temperatur = 22;`
- `int temperatur = 11;`
- `int temp = 11;`

**Aufgabe 6:** Welche Aussagen sind richtig? Kreuze die richtigen Aussagen an:

- Eine Variable kann gleichzeitig mehrere Zahlen abspeichern
- Eine Variable kann immer nur genau eine Zahl abspeichern
- Eine Variable kann auch Zeichen abspeichern
- Der Wert einer Variablen kann verändert werden
- Variablen vom Typ Integer speichern ganze Zahlen
- Variablen vom Typ Integer speichern Kommazahlen
- Variablen vom Typ Integer speichern Zeichen
- Eine Variable behält immer ihren Initialwert
- Eine Variable hat immer einen Namen und immer einen Datentyp
- Eine Variable hat immer einen Namen und manchmal einen Datentyp
- Der Datentyp bestimmt die Art der zu speichernden Daten

# Lösungen

## Arbeitsblatt 10 - Touch-Sensoren

**Aufgabe 1:** Beschreibe, wie das Zeit-Multiplex-Verfahren funktioniert.

Beim Zeit-Multiplex-Verfahren werden mehrere Signale zusammengefasst, so dass alle sechs Signale der Armsensoren von BOB3 mit nur zwei Mikrocontroller-Eingängen ausgewertet werden können.

Der Mikrocontroller misst die Messfelder zusammen mit jeweils einem Aktivierungsfeld und bekommt so die Information, ob und wo welcher Arm gerade berührt wird.

**Aufgabe 2:** Beschreibe, aus welchen Teilen die Armsensoren von BOB3 bestehen und wie sie funktionieren.

Ein Armsensor von BOB3 besteht aus drei Aktivierungsfeldern und zwei Messfeldern. Falls ein Aktivierungsfeld gleichzeitig mit einem Messfeld berührt wird, bekommt der Controller von BOB3 ein Signal, welches Aktivierungsfeld berührt wurde. So kann Bob zwischen ‚Arm oben berührt‘, ‚Arm mittig berührt‘ und ‚Arm unten berührt‘ unterscheiden.

**Aufgabe 3:** Welche Programm-Codes sind korrekt und würden compilieren? Kreuze die richtigen Antworten an, es sind mehrere Antworten möglich:

- `int sensorWert = bob3.getArm();`
- `int sensorWert == bob3.getArm(2);`
- `int sensorWert = bob3.getArm(1,2);`
- `int sensorWert = bob3.getArm(2,1);`
- `int wert = bob3.getArm(2,1);`
- `int WERT = bob3.getArm(2);`
- `int wert = getArm(1);`
- `if (2 == bob3.getArm(1))`
- `if (bob3.getArm(2) == 1)`
- `int sensorWert = bob3.getArm(1);`
- `int sensorWert = bob3.getArm(2);`

**Aufgabe 4:** Betrachte das folgende Programm und beschreibe genau, was bei welcher Aktion am Bob passiert!

```

1 #include <BOB3.h>
2
3 void loop() {
4   int wert1 = bob3.getArm(1);
5   int wert2 = bob3.getArm(2);
6
7   if (wert1 == 1) {
8     bob3.setEyes(WHITE, WHITE);
9     delay(200);
10    bob3.setEyes(OFF, OFF);
11    delay(200);
12  }
13
14  if (wert1 == 3) {
15    bob3.setWhiteLeds(ON, ON);
16    delay(200);
17    bob3.setWhiteLeds(OFF, OFF);
18    delay(200);
19  }
20
21  if ((wert1 == 2) && (wert2 == 2)) {
22    bob3.setEyes(ORANGE, WHITE);
23    bob3.setWhiteLeds(ON, OFF);
24    delay(200);
25    bob3.setEyes(WHITE, ORANGE);
26    bob3.setWhiteLeds(OFF, ON);
27    delay(200);
28  }
29
30 }
31

```

---

Das Programm reagiert mit verschiedenen Aktionen auf Aktivierungen der Armsensoren:

---

Falls Arm 1 oben berührt wird, blinken am Bob beide Augen gleichzeitig weiß.

Falls Arm 1 unten berührt wird, dann blinken am Bob beide Bauch-LEDs gleichzeitig weiß.

Falls Arm 1 und Arm 2 **gleichzeitig** mittig berührt werden, dann blinken die Bauch-LEDs abwechselnd weiß und die Augen blinken abwechselnd weiß und orange!

---

**Aufgabe 5:** Nenne alle möglichen Rückgabewerte der Methode `bob3.getArm()`.

**0, 1, 2, 3**

---

# Lösungen

## Arbeitsblatt 11 - switch-case

**Aufgabe 1:** Beschreibe, was die switch-case-Kontrollstruktur ist und wann man sie anwendet.

switch-case ist eine Kontrollstruktur die dazu dient, eine Verzweigung in viele verschiedene Fälle, abhängig vom Wert einer Variablen, zu ermöglichen. Falls man wenige Fälle unterscheiden möchte, kann man gut mit if-Abfragen arbeiten, bei vielen Fällen bietet sich die switch-case-Kontrollstruktur an!

**Aufgabe 2:** Werden bei der switch-case-Verzweigung des Beispiels von Blatt 2 **alle** Fälle ausgeführt? Begründung!

Nein! Es wird nur der Fall **case 3:** ausgeführt, da die Variable `anzahlLampen` den Wert 3 hat.

**Aufgabe 3:** Was passiert am BOB3, wenn im Beispiel von Blatt 2 in Zeile 5 die **drei** in eine **eins** geändert wird: `int anzahlLampen = 1; ?`  
Welcher Zweig wird ausgeführt?

Es wird der Zweig mit dem Fall **case 1:** ausgeführt, am Bob wird eine Led (Auge 1) eingeschaltet.

**Aufgabe 4:** Was passiert am BOB3, wenn im Beispiel von Blatt 2 in Zeile 5 die **drei** in eine **fünf** geändert wird: `int anzahlLampen = 5; ?`  
Welcher Zweig wird ausgeführt?

Es wird der **default:** Zweig ausgeführt, am Bob werden alle Leds ausgeschaltet.

**Aufgabe 5:** Beschreibe ausführlich die Funktion des folgenden Programms:

```

8 void loop() {
9   int wert1 = bob3.getArm(1);
10
11   switch (wert1) {
12     case 0:
13       bob3.setLed(EYE_1, OFF);
14       bob3.setLed(EYE_2, OFF);
15       break;
16
17     case 1:
18       bob3.setLed(EYE_1, RED);
19       bob3.setLed(EYE_2, RED);
20       break;
21
22     case 2:
23       bob3.setLed(EYE_1, ORANGE);
24       bob3.setLed(EYE_2, ORANGE);
25       break;
26
27     case 3:
28       bob3.setLed(EYE_1, GREEN);
29       bob3.setLed(EYE_2, GREEN);
30       break;
31   }
32
33   delay(50);
34 }
35

```

BOB3 arbeitet als Ampelmännchen:

Wenn Arm 1 oben gedrückt wird, dann leuchten beide Augen rot, wird der Arm in der Mitte gedrückt, leuchten beide Augen orange, wird der Arm unten gedrückt, leuchten beide Augen grün. Ohne Berührung sind die Augen aus!

**Aufgabe 6:** Betrachte das Programm von Aufgabe 5. Was passiert, wenn Bob's Arm 2 mittig berührt wird? Lies die Aufgabenstellung genau!

- |  |   |
|--|---|
| <input type="checkbox"/> Beide Augen leuchten orange           | <input type="checkbox"/> Beide Augen leuchten weiss |
| <input checked="" type="checkbox"/> Beide Augen leuchten nicht | <input type="checkbox"/> Beide Augen leuchten grün  |
| <input type="checkbox"/> Beide Augen leuchten rot              | <input type="checkbox"/> Beide Bauch-LEDs leuchten  |

**Aufgabe 7:** Betrachte das Programm von Aufgabe 5. Welche Zeile wird **nach** der break-Anweisung in Zeile 25 ausgeführt?

- |                                   |  |
|-----------------------------------|--|
| <input type="checkbox"/> Zeile 9  | <input type="checkbox"/> Zeile 27            |
| <input type="checkbox"/> Zeile 11 | <input checked="" type="checkbox"/> Zeile 33 |
| <input type="checkbox"/> Zeile 22 | <input type="checkbox"/> Zeile 34            |