

# Programmieren lernen mit BOB3

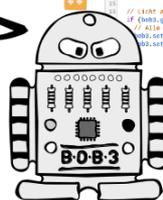
Lehrerbegleitheft - Sekundarstufe I

Einführung in die textuelle Programmierung mit dem Roboter BOB3

## Modul 1

PROBIERE MAL  
EINE ANDERE  
FARBE AUS!

ÄNDERE DIE  
BEDINGUNG  
 $1 == 3$   
WAS ERWARTEST  
DU?



# Modul 1

## Einführung und Grundlagen

Dieses Dokument dient zur Übersicht des Kurses „Programmieren lernen mit BOB3“ für die Sekundarstufe I. Es werden die Lehr- und Lernmaterialien, die benötigten Zeiteinheiten und Vorschläge für konkrete Unterrichtseinheiten gegeben. Im Folgenden ist eine Unterrichtseinheit für 45 min ausgelegt.

### Übersicht Modul 1:

**Modul 1 umfasst insgesamt 6 Unterrichtseinheiten mit je ca. 45 Minuten**



Lehrerbegleitheft mit Konzepten und Ablaufplänen der einzelnen Einheiten

10 Programmier-Tutorial Einheiten:



Angefangen bei ersten Programmierschritten und dem Verstehen des Prinzips der Programmierung von Hardware (Compilieren, Übertragen) lernen die SuS einzelne Aktionen zu programmieren, erzeugen einfache Sequenzen, erlernen das Prinzip der Verzögerung mit der delay()-Funktion, programmieren diverse LED-Blinklicht-Varianten, experimentieren mit verschiedenen Farbkonstanten und lernen Funktionen mit Parametern kennen.



Arbeitsblatt 1 - „**Roboter**“  
+ Lösungen zum AB 1



Arbeitsblatt 2 - „**Benutzeraccount**“  
+ Lösungen zum AB 2



Arbeitsblatt 3 - „**Klammern & Co.**“  
+ Lösungen zum AB 3



Arbeitsblatt 4 - „**Programmieren**“  
+ Lösungen zum AB 4



# 1. Unterrichtseinheit

In der ersten Unterrichtseinheit werden die Schüler zunächst an das Thema Robotik herangeführt. Mittels eines Infotextes erlernen sie, was ein Roboter alles sein kann und wie dieser Begriff definiert wird. Arbeitsblatt 1 beschäftigt sich mit den Fragen „Was ist eigentlich ein Roboter?“, „Welche verschiedenen Roboterarten gibt es?“, „Wozu dienen Roboter?“.



**Zeitbedarf:** ca. 45 min



**Vorkenntnisse:** keine



**Material:** Papier, Stift, Arbeitsblatt 1

## Ablauf

### 1) Bearbeiten von Arbeitsblatt 1: Aufgabe 1 und Aufgabe 2

Die Schüler lesen den Text „Was ist eigentlich ein Roboter?“

Die Schüler kreuzen zutreffende Aussagen über Roboter an

In einer anschließenden Diskussion reflektieren sie ihre Ergebnisse

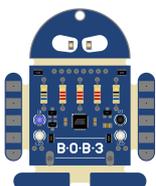


### 2) Bearbeiten von Arbeitsblatt 1: Aufgabe 3

Die Schüler schreiben auf, welche Roboterarten sie selber kennen und wo ihnen diese begegnet sind

**Differenzierung I:** Sie diskutieren, wozu die jeweilige Roboterart dient

**Differenzierung II:** Sie diskutieren die Unterscheidungsmerkmale von Robotern



## 2. Unterrichtseinheit

In der zweiten Unterrichtseinheit richtet sich jeder Schüler einen **Benutzeraccount** auf der Seite [www.ProgBob.org](http://www.ProgBob.org) ein. Mit diesem Account haben die Schüler Zugriff auf alle Lern-, Aufgaben- und Wissensüberprüfungseinheiten des Programmier-Tutorials. Anschließend starten die Schüler das Programmier-Tutorial „**Intro I**“ und lernen die Programmieroberfläche kennen und bedienen.



**Zeitbedarf:** ca. 45 min



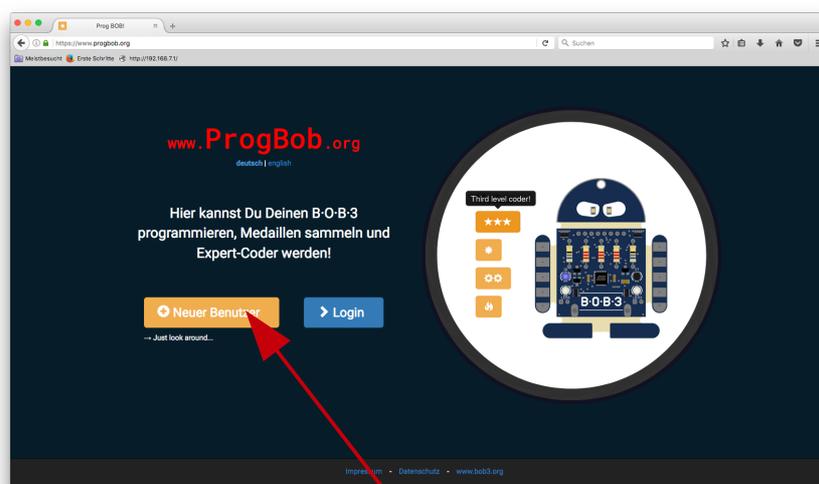
**Vorkenntnisse:** rudimentäre Kenntnisse im Umgang mit einer Computer-Maus, einer Tastatur und einem Web-Browser



**Material:** Papier, Stift, Arbeitsblatt 2, PC oder Laptop, BOB3 mit Helm

## Ablauf

Das Arbeitsblatt 2 wird ausgeteilt, die SuS starten den Webbrowser und gehen auf die Seite <http://www.ProgBob.org>:



[www.bob3.org](http://www.bob3.org)

Per Klick auf den Button „**Neuer Benutzer**“ öffnet sich der Dialog „*Neuen Account anlegen*“.

Die SuS denken sich jeweils einen eigenen **Benutzernamen** und ein **Passwort** aus:

Der **Benutzername** muss mindestens 4 Zeichen enthalten!  
 Das **Passwort** muss mindestens 6 Zeichen enthalten!

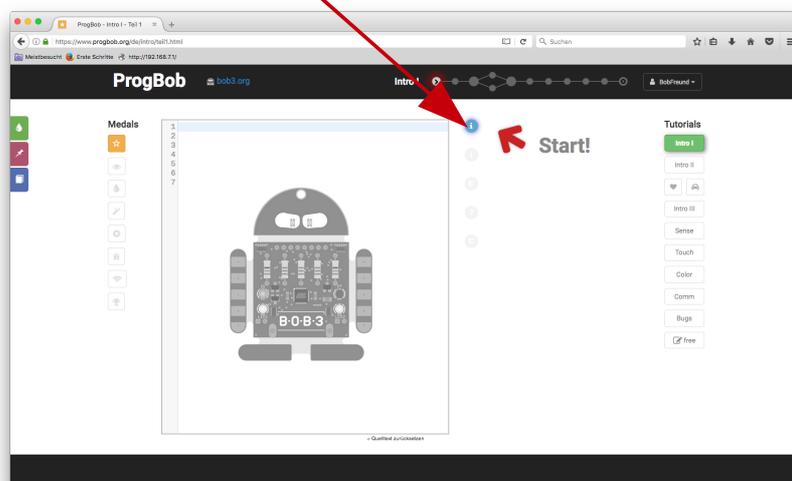
**Email [optional]:**

Die Angabe einer Email-Adresse ist **optional** möglich: Falls man sein Passwort vergessen hat, kann es per Email angefordert werden.

Die Schüler tragen beide Wörter ein und bestätigen den Dialog mit **OK**. Fertig!

Hinweis: Damit die Schüler die Benutzerdaten nicht vergessen können, tragen sie am besten beide Wörter auf dem **Arbeitsblatt** ein.

Die Schüler starten das Tutorial per Klick auf die erste „**Info-Einheit**“:



Das Tutorial ist **interaktiv** aufgebaut und **selbsterklärend**, so dass die SuS nun selbstständig und im jeweils eigenen Tempo arbeiten können.

Prinzipieller Aufbau des Tutorials:

 Die blauen „Info-Einheiten“ bieten Lerninhalte, z.B.:

**\_\_info\_\_**

Jetzt lernen wir eine **neue** Art von **Bedingungen** kennen:

Im vorigen Beispiel haben wir zwei Zahlen auf **Gleichheit** überprüft. Jetzt wollen wir auf **größer als** und auf **kleiner als** prüfen:

`3 > 1`

> bedeutet "größer als"

`3 > 1` ist **wahr**

`1 > 4` ist **falsch**

 Die grauen „Aufgaben-Einheiten“ bieten Aufgaben, die von den SuS bewältigt werden müssen, z.B.:

**\_\_aufgabe\_\_**

Ändere die Bedingung:

`100 > 1000`

 Compiliere das neue Programm und teste es auf deinem BOB3.

Was passiert jetzt?

 Die roten „Quiz-Einheiten“ dienen zur Wissensüberprüfung und müssen beantwortet werden, z.B.:

Per Klick auf den Button „Quiz auswerten!“ bekommen die SuS direktes Feedback zu ihrer Lösung.

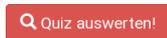
**\_\_quiz\_\_**

1: Wahr oder falsch: `5 < 4`  
 wahr  falsch

2: Wahr oder falsch: `5 != 4`  
 wahr  falsch

3: Wahr oder falsch: `5 != 5`  
 wahr  falsch

4: Wahr oder falsch: `1000 > -1000`  
 wahr  falsch

 Quiz auswerten!



Die Schüler starten mit dem Programmier-Tutorial „Intro I“:

Zunächst wird erklärt, wie am Roboter eine Augen-Led eingeschaltet werden kann. Dabei lernen die Schüler gleichzeitig die `setup()`-Funktion kennen. Dann sollen die Schüler das Auge konkret einschalten und lernen, den dafür benötigten Quellcode einzugeben, zu compilieren und auf den Roboter zu übertragen.

1

\_\_info\_\_

**Jetzt werden wir erst einmal eine LED einschalten!**

Das machen wir in der `setup()` Funktion...

Der Text zwischen den geschweiften Klammern `{...}` wird direkt nach dem Einschalten automatisch **einmal** ausgeführt.

2

\_\_info\_\_

Unsere `setup()` Funktion enthält genau eine Anweisung:

```
bob3.setLed(EYE_1, WHITE);
```

Mit dieser Anweisung wird das Auge 1 `EYE_1` vom `bob3` mit der Farbe Weiß `WHITE` eingeschaltet.

3

\_\_aufgabe\_\_

Compiliere dein Programm, indem du links unten in der Leiste auf den Button `▶ Compile` drückst!

Der Compiler erzeugt aus dem Quellcode eine `.bob3` Datei, die du mit BobDude auf den BOB3 übertragen kannst.

Die Datei findest du nach dem compilieren neben dem Compile-Button.

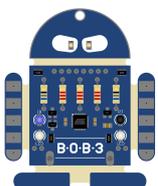
4

\_\_aufgabe\_\_

Wenn du jetzt unten die Datei `teil2.bob3` anklickst, öffnet sie sich automatisch im **BobDude** Übertragungsprogramm.

📌 **Klicke dort auf den "Program!" Button.**

Wenn alles geklappt hat, sollte am BOB3 jetzt das **Auge 1 weiß** leuchten.



Nach der erfolgreichen ersten Programmierung des Roboters müssen die Schüler noch eine **Wissensabfrageeinheit** lösen, um zum nächsten Teil des Tutorials zu gelangen:

**5**

\_\_quiz\_\_

1: Wann wird die `setup()` Funktion aufgerufen?

- kurz vor dem Abschalten
- immer wenn der BOB3 auf dem Tisch sitzt
- direkt nach dem Einschalten

2: Wie oft wird die `setup()` Funktion aufgerufen?

- keinmal
- einmal
- immer wieder

🔍 Quiz auswerten!

➔

\_\_quiz\_\_

1: Wann wird die `setup()` Funktion aufgerufen? 😊

- kurz vor dem Abschalten
- immer wenn der BOB3 auf dem Tisch sitzt
- direkt nach dem Einschalten

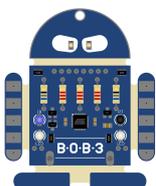
2: Wie oft wird die `setup()` Funktion aufgerufen? 😊

- keinmal
- einmal
- immer wieder

🔍 Quiz auswerten!

👏 Prima, alles richtig!

Das Quiz wird sofort online ausgewertet und bietet den Schülern so direktes Feedback über ihr erworbenes Wissen.



## 3. + 4. Unterrichtseinheit

In der dritten und vierten Unterrichtseinheit bearbeiten die Schüler in eigenem Tempo das „Intro I“ Tutorial. Das Tutorial vermittelt den Schülern zunächst den grundlegenden Umgang und die Bedienung der Lernerfläche. In den folgenden Kapiteln bekommen die Schüler die Aufgabe, kleine Änderungen an den Programmtexten durchzuführen. Die Schüler erlernen die informatischen Grundkonzepte „**Sequenz**“, „**Aktion**“ und „**Verzögerung**“.



**Zeitbedarf:** ca. 90 min



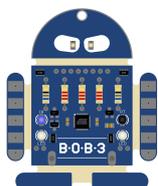
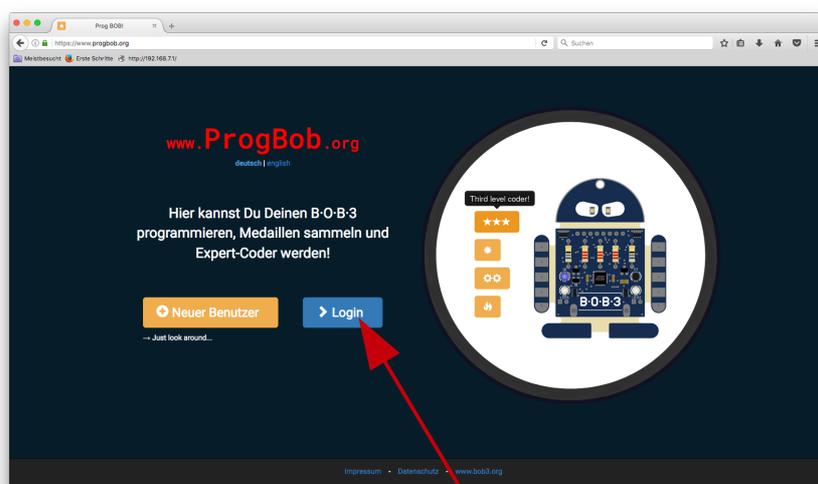
**Vorkenntnisse:** rudimentäre Kenntnisse im Umgang mit einer Computer-Maus, einer Tastatur und einem Web-Browser



**Material:** Papier, Stift, Arbeitsblatt 3, PC oder Laptop, BOB3 mit Helm

## Ablauf

Die Schüler loggen sich mit ihren aus der vorherigen Unterrichtseinheit erstellten Benutzerdaten auf der Seite <http://www.ProgBob.org> ein:



[www.bob3.org](http://www.bob3.org)

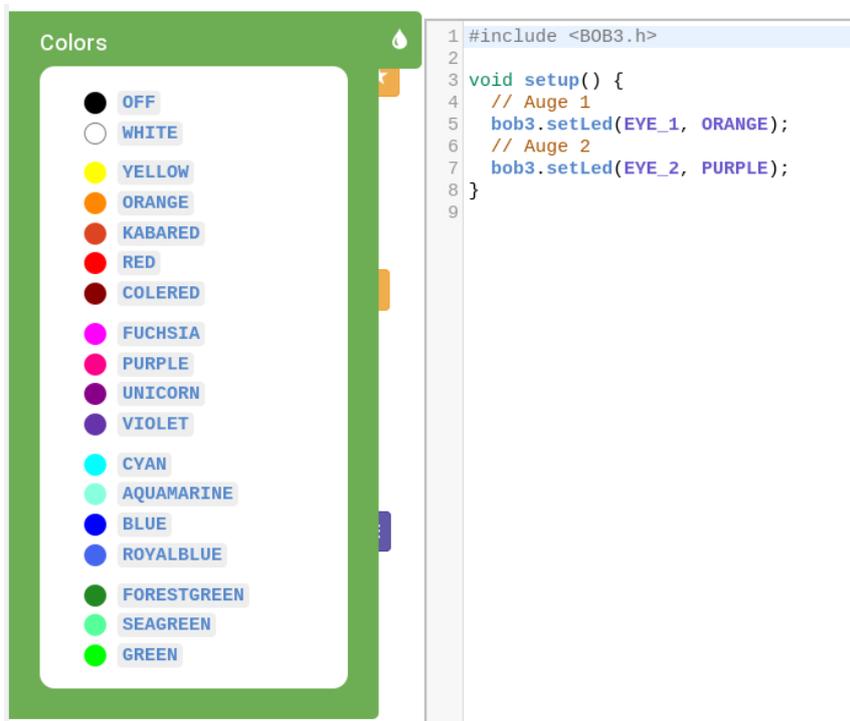
!! Wichtig: erneutes Einloggen mit bereits vorhandenen Daten über „**Login**“ !!

Der **Lernstand** des einzelnen Schülers ist im jeweiligen Benutzeraccount gespeichert, so dass jeder an der jeweils individuellen Stelle weiterarbeiten kann.

Lerninhalte des „Intro I“-Tutorials:

- Die SuS sollen die Web-Oberfläche der ProgBob Tutorials kennen lernen und bedienen können
- Die SuS sollen Programm-Quelltext selbstständig compilieren können und auf den Mikrocontroller übertragen
- Die SuS sollen zunächst vorgegebene kleine Änderungen am Quelltext vornehmen
- Die SuS sollen den Funktionsaufruf zur Veränderung der Farbe der LEDs kennen lernen und verstehen
- Die SuS sollen die delay()-Funktion kennen lernen um Blinkeffekte mit den LEDs zu realisieren
- Die SuS sollen jetzt selbstständig Änderungen am Quelltext vornehmen (Farbe der LEDs frei verändern, Parameter der delay()-Funktion, eigene Blinkroutinen)

Die Schüler lernen unter anderem Farben als Parameter kennen und wenden dieses Wissen in zunächst kleinen und überschaubaren Programmen an:



The image shows two side-by-side windows from the BOB3 web interface. The left window, titled 'Colors', displays a list of color options for LED control, each with a corresponding colored circle and a text label: OFF, WHITE, YELLOW, ORANGE, KABARED, RED, COLERED, FUCHSIA, PURPLE, UNICORN, VIOLET, CYAN, AQUAMARINE, BLUE, ROYALBLUE, FORESTGREEN, SEAGREEN, and GREEN. The right window shows a code editor with the following C++ code:

```

1 #include <BOB3.h>
2
3 void setup() {
4     // Auge 1
5     bob3.setLed(EYE_1, ORANGE);
6     // Auge 2
7     bob3.setLed(EYE_2, PURPLE);
8 }
9

```



Die Schüler bearbeiten das **Arbeitsblatt 3**:



„Sind Klammern, Semikolons und andere Zeichen wichtig beim Programmieren?“

Die SuS erlernen anhand eines einfachen Programm-Beispiels, dass **Klammern**, **Semikolons** und andere Zeichen wichtig beim Programmieren sind. Außerdem muss auf **Groß-** und **Kleinschreibung** und **korrekte Schreibweisen** geachtet werden.

```

1 #include <BOB3.h>
2
3 void setup() {
4     // Auge 1
5     bob3.setLed(EYE_1, ORANGE);
6     // Auge 2
7     bob3.setLed(EYE_2, PURPLE);
8 }
9

```

Die Schüler erlernen den Aufbau der `setup()`-Funktion., das Einfügen von Kommentaren in Quelltext und die Anweisung `bob3.setLed(EYE_1, ORANGE);` kennen.

```

1 #include <BOB3.h>
2
3 void setup() {
4     // Auge 1
5     bob3.setLed(EYE_1, ORANGE);
6     // Auge 2
7     bob3.setLed(EYE_2, PURPLE);
8 }
9

```

Die Schüler bearbeiten die Aufgaben 1 bis 5 vom **Arbeitsblatt 3**. Es sind jeweils verschiedene Fehler in der Semantik des Programmcodes eingebaut, die von den Schülern gefunden und berichtigt werden müssen.



## 5. + 6. Unterrichtseinheit

In der fünften und sechsten Unterrichtseinheit beenden die Schüler das „Intro I“ Tutorial. Die Schüler beherrschen nun die informatischen Grundkonzepte „**Sequenz**“, „**Aktion**“ und „**Verzögerung**“. Zusätzlich bearbeiten die SuS das **Arbeitsblatt 4 „Programmieren“**. Das Arbeitsblatt behandelt den Ablauf und die Bedeutung des eigentlichen Programmiervorgangs eines Embedded-Systems.



**Zeitbedarf:** ca. 90 min



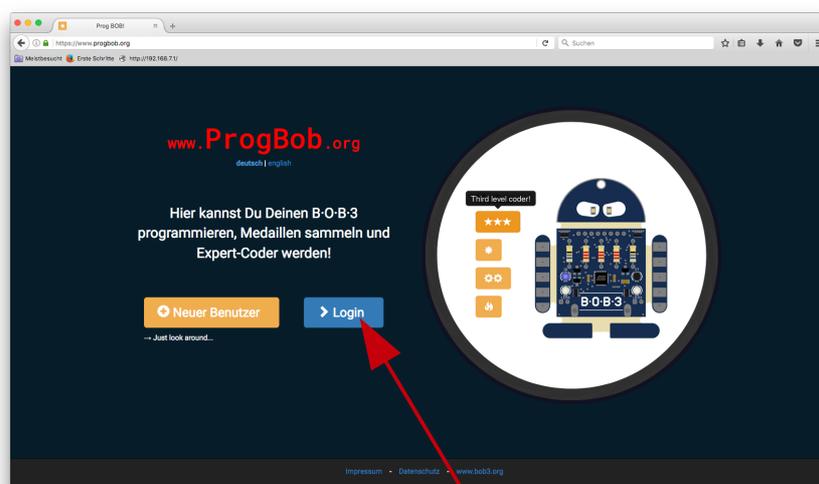
**Vorkenntnisse:** rudimentäre Kenntnisse im Umgang mit einer Computer-Maus, einer Tastatur und einem Web-Browser



**Material:** Papier, Stift, Arbeitsblatt 4, PC oder Laptop, BOB3 mit Helm

## Ablauf

Die Schüler loggen sich mit ihren bereits erstellten Benutzerdaten im Programmier-Tutorial auf der Seite <http://www.ProgBob.org> ein:



[www.bob3.org](http://www.bob3.org)

!! Wichtig: erneutes Einloggen mit bereits vorhandenen Daten über „**Login**“ !!

Je nach Lernfortschritt bearbeiten die SuS die jeweilige Tutorial-Einheit des „Intro I“ Tutorials. Die einzelnen Tutorial Einheiten bauen inhaltlich direkt aufeinander auf, wobei der Schwierigkeitsgrad der einzelnen Einheiten sich im Laufe des Tutorials stetig steigert.

```

1 #include <BOB3.h>
2
3 void setup() {
4
5 }
6
7 void loop() {
8   bob3.setEyes(CYAN, KABARED);
9   delay(500);
10  bob3.setEyes(AQUAMARINE, ORANGE);
11  delay(500);
12  bob3.setEyes(BLUE, PURPLE);
13  delay(500);
14  bob3.setEyes(ROYALBLUE, VIOLET);
15  delay(500);
16  bob3.setEyes(GREEN, GREEN);
17  delay(2000);
18  bob3.setEyes(OFF, OFF);
19  delay(500);
20 }
21

```

In der neunten Lerneinheit bekommen die SuS die Aufgabe, ein bereits bestehendes Programm zu verstehen und selbstständig kleinere Änderungen vorzunehmen. Die SuS sollen hierbei das Konzept und die Unterschiede der Funktionen setup() und loop() verstanden haben.

    aufgabe    

Ändere den Programmcode so, dass Folgendes passiert:

**Auge 1** soll - wie vorher - in vier verschiedenen **Blautönen** blinken!

**Auge 2** soll - diesmal - in vier verschiedenen **Rottönen** blinken!

Compiliere dein Programm und teste es auf dem BOB3.

    aufgabe    

Nun ändere den Programmcode so, dass Folgendes passiert:

**Auge 1** blinkt - wie vorher - 4× in verschiedenen **Blautönen**.

**Auge 2** blinkt - wie vorher - 4× in verschiedenen **Rottönen**.

Dannach soll:

- eine kurze Pause folgen
- beide Augen für **2 Sekunden** (2000 Millisekunden) in **grellem Grün** leuchten
- beide Augen ausgeschaltet werden
- eine kurze Pause folgen

Anschließend geht's automatisch wieder von vorn los!

Compiliere dein Programm und teste es auf dem BOB3.

Anschließend sollen die Schüler das Programm eigenständig um eine kleine **Sequenz erweitern**. Die SuS sollen sich hierbei konkret mit einer **algorithmischen** Aufgabenstellung auseinandersetzen, eine Lösung entwickeln, selbstständig umsetzen und verifizieren.



Die Schüler bearbeiten das **Arbeitsblatt 4**:



„Was bedeutet eigentlich Programmieren?“

Die SuS erlernen, dass mit dem Begriff **Programmieren** die Tätigkeit bezeichnet wird, ein **Computerprogramm** zu erstellen. Mit Computerprogrammen können Autos gesteuert, Waschmaschinen eingeschaltet oder auch Raketen zum Mars geflogen werden. Insgesamt können Computerprogramme also ganz verschiedene Geräte steuern. Der Programmierer, auch Softwareentwickler genannt, hat eine Idee im Kopf, was ein Gerät tun soll. Dann muss er seine Idee in einer **Programmiersprache** (Quellcode) formulieren.

„Was genau wird programmiert?“

Die SuS erlernen, welcher **Bestandteil** der einzelnen Geräte konkret programmiert wird. Es wird natürlich nicht die Plastikhülle programmiert, sondern ein sogenannter **Mikrocontroller**. Beim BOB3 ist der Mikrocontroller gut zu sehen: Der Controller ist das kleine schwarze Kästchen auf dem Bauch. Wenn man ganz genau hinschaut, dann kann man viele Anschlussdrähte entdecken, die den Controller mit der restlichen Elektronik verbinden.

„Was passiert beim Programmieren?“

Die SuS erlernen, dass der Programmierer seinen **Quellcode**, also Befehle, Buchstaben und Zeichen einer bestimmten **Programmiersprache**, in einen **Editor** eingibt. Ein Editor ist einfach ein Eingabefenster am Computer. Damit der Mikrocontroller vom BOB3 versteht, was er tun soll, muss dieser Quellcode zuerst in **Maschinensprache** übersetzt werden. Das macht ein **Compiler**. Man sagt auch, man kompiliert den Quellcode. Dann wird das Programm in Maschinensprache auf den Mikrocontroller übertragen, fertig!!

