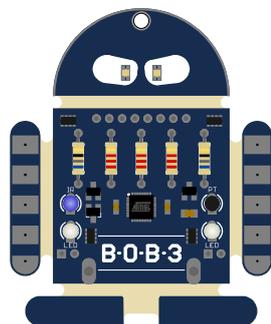


BOB3 Programmierung mit Open Roberta

Lehrerhandreichung für die Grundschule ab Klasse 3



Schalte LED Auge links an Farbe

Arm links oben gedrückt?



Geeignet zur Unterstützung der Kompetenzen im  MEDIENPASS NRW
Dieses Dokument steht unter der Creative Commons Namensnennung 4.0 International (CC BY-SA 4.0) Lizenz. Autor: Katja Bach. Herausgeber: nicai-systems, Stolberg, 2018. DIGITALE BILDUNG: BOB3 Programmierung mit Open Roberta, Lehrerhandreichung

BOB3 Programmierung mit Open Roberta

Lehrerhandreichung für die Grundschule

Version 1.4

Liebe Lehrerinnen und Lehrer,

die folgenden Seiten vermitteln Ihnen eine kompakte Grundlage für die praktische Durchführung des Unterrichts mit BOB3 in der Grundschule. Zusätzlich werden Details zu den Lernzielen, zu den angestrebten Kompetenzen und zum didaktischen Konzept dargestellt.

Da insbesondere im Bereich der Informationstechnologie eine große Heterogenität in den Vorkenntnissen der SuS vorhanden ist, war die Umsetzung der Wissensvermittlung in Form eines individuellen, differenzierten Unterrichts bei der Entwicklung ein zentrales Thema. Die BOB3-Lernkarten wurden nach dem Konzept des Stationenlernens ausgelegt, wodurch sie von den SuS selbstbestimmt und in individuellem Lerntempo bearbeitet werden können. Durch die zugehörigen Werkstattpläne hat die Lehrkraft jederzeit einen Überblick über die einzelnen Lernstände. Insbesondere wurde darauf geachtet, dass schwächere Schüler die Motivation bei der Durchführung nicht verlieren, und dass leistungsstarken Schülern anspruchsvolle Aufgaben im Bereich der Differenzierung geboten werden. Da die Gruppe der SuS von den Lernkarten angeleitet und motiviert wird, ergibt sich für die Lehrkraft die Möglichkeit, den gewonnenen Freiraum zur individuellen Förderung einzelner SuS einzusetzen.

Bildungsziele:

- Die SuS sollen die Bedeutung des Begriffs ‚Programmierung‘ verstehen und die Tätigkeit des ‚Programmierens‘ selbst durchgeführt haben.
- Die SuS sollen erkennen, welche Möglichkeiten sich durch Programmierung ergeben. Sie sollen die Potentiale Informationstechnischer Systeme erkennen und überlegen wo ihre Grenzen sind.
- Die SuS sollen lernen, wie Informationstechnologie entwickelt wird und welchen Einfluss diese auf die Gesellschaft ausübt.
- Die SuS sollen lernen wie man Strategien zur Problemlösung entwickelt und modelliert und sollen diese dann mittels

Algorithmen und passenden digitalen Werkzeugen umsetzen.

- Das Interesse der SuS an Technik, Mathematik & Informatik soll durch das Projekt geweckt und positiv verknüpft werden.

Technische Voraussetzungen:

- iPad Sätze oder
- Android-Tablet Sätze oder
- Computerarbeitsplätze mit Windows, Linux oder MacOS Betriebssystem und Internet-Zugang
- 1 ProgBob/BobDock und 1 BOB3 pro Computerarbeitsplatz

Lernkarten

Die Lernkarten sind zur Durchführung eines offenen Unterrichts in Form des Stationenlernens konzipiert. Dabei bearbeiten die Schüler individuell oder in kleinen Gruppen die einzelnen Stationen. Die jeweilige Station ist auf den Lernkarten oben links als Buchstabe aufgeführt, innerhalb einer Station sind die Karten durchnummeriert. Zur Differenzierung sind die Karten oben rechts mit einem bis drei Sternen markiert. Anspruchsvolle Karten haben beispielsweise drei Sterne. Zusätzlich gibt es bei einigen Karten einzelne Aufgaben mit mehreren Sternen. Diese sind ebenfalls zur Differenzierung gedacht. Jede Lernkarte hat zusätzlich zur besseren Übersicht eine Titelzeile als Überschrift, in der das Lernziel der Karte grob beschrieben wird. Als Hilfestellung für die SuS enthält jede Karte am unteren Rand die Web-Adresse der Open Roberta Programmieroberfläche.

Lerneinheit
Station A
Karte 6

★ einfach
★★ mittel
★★★ anspruchsvoll



A6 Grundlagen: **Körper-LEDs** ★

Aufgabe 1: Ändere die Eigenschaft „links“ in „rechts“:



Aufgabe 2: ▶ Starte dein Programm auf dem Roboter - was macht der Bob jetzt?

Aufgabe 3: Verwende noch einen zweiten Block, so dass **beide** Körper LEDs leuchten!



★ Aufgabe 4: Jetzt sollen **zusätzlich** noch **beide** Augen leuchten! Hast du eine Idee, wie das geht? Probiere mal!

lab.open-roberta.org OR Lernkarten GS · BOB3 · Station A · V1.1 · (6/6)

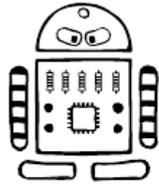


Programmierung mit
Open-Roberta

Robotersystem
BOB3

Werkstattpläne

Passend zu den Lernkarten gibt es zu jeder Station einen Werkstattplan als Laufzettel. Die SuS bearbeiten parallel die Lernkarten und den jeweiligen Werkstattplan und tragen dort ihren Lernfortschritt ein. Zusätzlich beantworten die SuS hier die auf den Lernkarten gestellten Fragen. Die Lehrkraft erhält so jederzeit einen Überblick über die jeweiligen Lernstände.

Name: _____ Klasse: _____		
	Werkstattplan Station A OpenRoberta mit BOB3	 
A1 + A2	Datum: _____ Erledigt: _____	
A3	Aufgabe 2: _____ _____ Aufgabe 4: _____ _____ Datum: _____ Erledigt: _____	
A4	Datum: _____ Erledigt: _____	
A5	Aufgabe 3: <input type="checkbox"/> Ja! <input type="checkbox"/> Nein! Datum: _____ Erledigt: _____	
A6	Aufgabe 3: _____ _____ Datum: _____ Erledigt: _____	
A7	Datum: _____ Erledigt: _____	
 OR Lernkarten GS · BOB3 · Station A · V1.4 · (Laufzettel)		

BOB3 – Technische Übersicht

BOB3 ist ein kleiner Roboter der merkt, ob seine Arme berührt werden und wenn ja, wo! Er kann Freunde erkennen, seine weißen Scheinwerfer einschalten, seine Augen in allen Farben blinken lassen, nah und fern unterscheiden und einiges mehr! Die Schülerinnen und Schüler können ihn frei programmieren, ihm einen eigenen binären Code geben oder ihn mit einer Knopfzelle und dem beiliegenden Lanyard als blinkendes Gadget um den Hals tragen.

Die wichtigsten Elemente von BOB3 sind:



rgb

Augen-LEDs: können in allen Farben leuchten



white

Körper-LEDs: weiße, helle Scheinwerfer



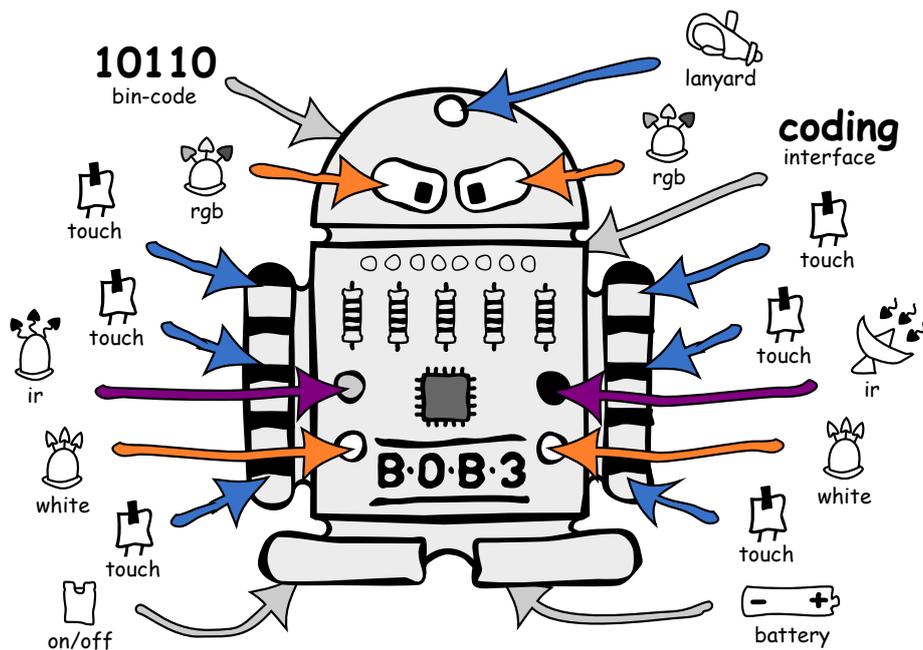
touch

Touch-Sensoren an den Armen, mit denen BOB3 auf Berührungen reagiert



ir

IR-Sensor, mit dem BOB3 nah und fern unterscheiden kann



Weiterhin verfügt BOB3 noch über



Computer-Gehirn: Ein kleiner Mikroprozessor der den Programmcode ausführt



Ein-/Aus-Schalter



Standard-Knopfzelle zum Herumtragen (während BOB3 programmiert wird, bekommt er den Strom über den USB-Anschluss)



Die Anschlussbuchse für den ProgBob-Helm, damit BOB3 programmiert werden kann



persönlicher Binärcode zwischen 0 und 31, damit die SuS die Bobs in der Klasse unterscheiden können



Datenübertragung zwischen zwei BOB3 Einheiten:
Infrarot Sender und Empfänger



Temperatursensor (im Mikroprozessor integriert)



Ein Lanyard mit dem man sich BOB3 als blinkendes Gadget umhängen kann...

Open Roberta

Open Roberta ist eine frei verfügbare, cloudbasierte grafische Programmierumgebung, mit der auch Neulinge ganz einfach und intuitiv programmieren lernen können. Die Programmierumgebung ist ein Projekt innerhalb der Roberta Initiative – Lernen mit Robotern vom Fraunhofer IAIS, Sankt Augustin. Die Initiative hat das Ziel, den Zugang zur Programmierung prinzipiell zu vereinfachen, indem technische und fachliche Barrieren abgebaut werden.

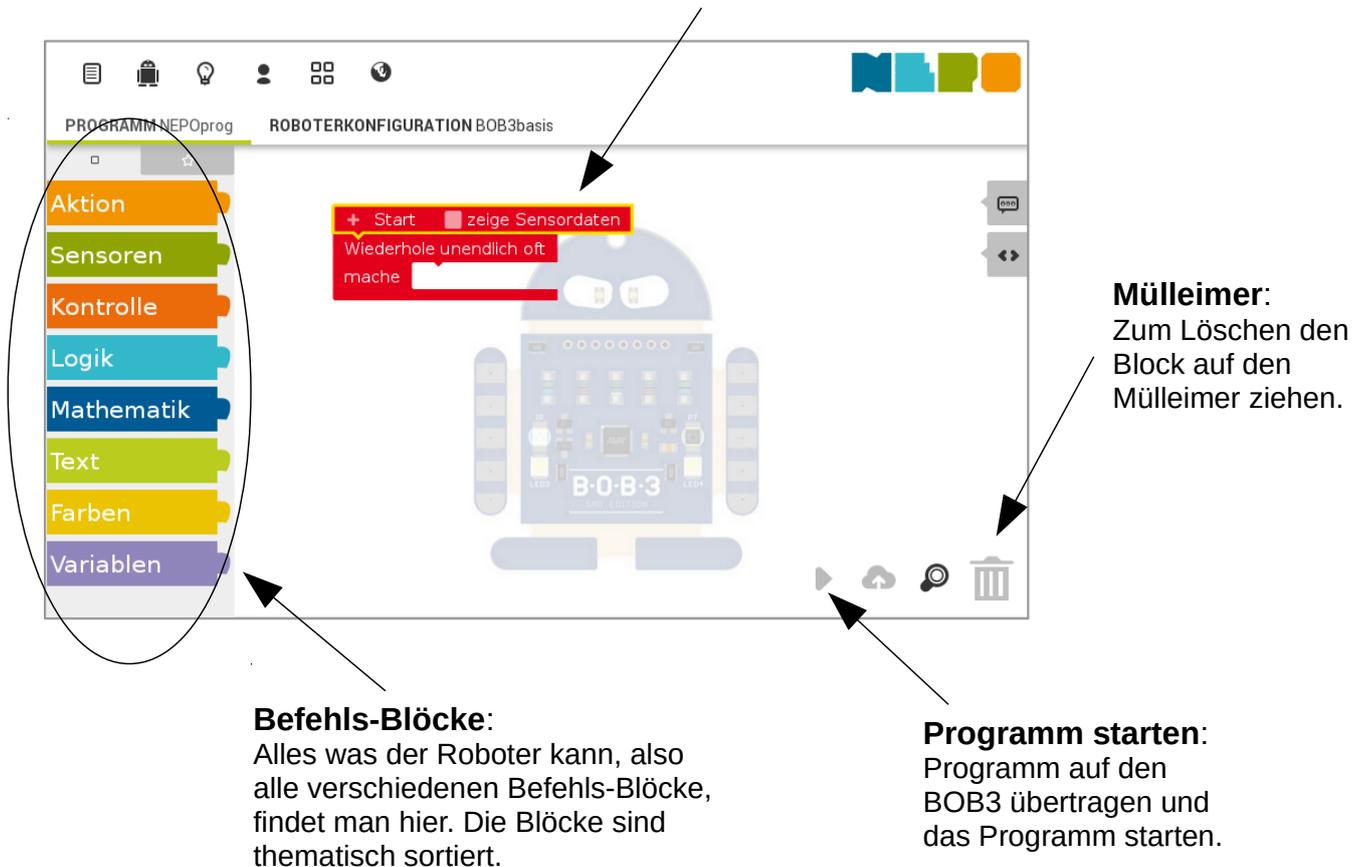
Um den BOB3 mit Open Roberta zu programmieren, geht man wie folgt vor:

- **iPad / Android-Tablet** → BOB3 **App** starten und **Grundschule** auswählen
- **Laptop / PC** → Starten des **Webrowsers** und Öffnen der Seite **rob.bob3.org**

Die Oberfläche ist übersichtlich gestaltet und intuitiv bedienbar, die wichtigsten Bedienelemente sind hier kurz dargestellt:

Programm-Block:

Alles was der Roboter machen soll, wird hier eingefügt.



Mülleimer:
Zum Löschen den Block auf den Mülleimer ziehen.

Befehls-Blöcke:
Alles was der Roboter kann, also alle verschiedenen Befehls-Blöcke, findet man hier. Die Blöcke sind thematisch sortiert.

Programm starten:
Programm auf den BOB3 übertragen und das Programm starten.

Vorbereitungen



Informationen für den System-Administrator

Um BOB3 mit Open Roberta zu programmieren, muss für **iPads** und **Tablets** die **BOB3 App**, für **Laptops** und **PCs** ein kleines Programm installiert werden.

Es sind folgende Alternativen wählbar:

iPad mit IOS:

Benötigt wird:

- BOB3-App aus dem App Store laden:
<https://itunes.apple.com/us/app/bob3-programmieren-lernen/id1449188606?mt=8>



Tablet mit Android:

Benötigt wird:

- BOB3-App aus dem Google Play Store laden:
<https://play.google.com/store/apps/details?id=org.bob3.bobdude>



Windows:

Benötigt wird:

- BobDude: <http://dude.bob3.org>
 1. **BobDude** Programm als Administrator installieren
 2. BOB3-ProgBob-Helm an den USB Port anschließen, die LEDs am Helm sollten nach kurzer Zeit (automatische Treiberinstallation) aufleuchten



MacOS:

Benötigt wird:

- Open Roberta Create Agent: MacOS_ORCA_x.x.x.pkg
<https://github.com/OpenRoberta/arduino-create-agent/releases>
1. BOB3-ProgBob-Helm an den USB Port anschließen, die LEDs am Helm sollten nach kurzer Zeit (automatische Treiberinstallation) aufleuchten
 2. Open Roberta Create Agent installieren (Doppelklick auf das Installationspaket)

Weitere Informationen zur Installation des Open Roberta Create Agent:
<https://jira.iais.fraunhofer.de/wiki/display/ORInfo/Vorbereitung+bob3>



Linux:

Benötigt wird:

- Open Roberta Create Agent: Linux64_ORCA_x.x.x.tar.gz
<https://github.com/OpenRoberta/arduino-create-agent/releases>
1. BOB3-ProgBob-Helm an den USB Port anschließen, die LEDs am Helm sollten nach kurzer Zeit (automatische Treiberinstallation) aufleuchten
 2. Open Roberta Create Agent als root installieren
tar -xzf Linux64_ORCA_0.0.1.tar.gz
cd Linux/
sh install.sh



Station A: Einführung - Grundlagen

Die Bearbeitung der **Station A** ist die Voraussetzung für alle anderen Stationen und sollte somit von allen Schülerinnen und Schülern **als Erstes** bearbeitet werden!

In dieser Station lernen die SuS zunächst die grundlegende Bedienung der Open Roberta Oberfläche und die Übertragung der Programme auf den BOB3. Später lernen sie erste Befehlsblöcke zur Veränderung der Farbe der Augen kennen und verwenden diese.



Schwierigkeitsgrad: Einfach ★

Zeitbedarf: ca. 20-40 Minuten

Voraussetzungen:

- Die SuS sollten über altersgerechte Lese-Kompetenzen verfügen
- SuS sollten vorab schon mal mit dem Internet in Kontakt gekommen sein
- SuS sollten wissen, wie man einen Web-Browser startet und wie man dort die Web-Adresse mit der Tastatur eingibt

Lernziele:

- SuS sollen die Web-Oberfläche von Open Roberta kennen lernen und bedienen können
- SuS sollen einen Befehlsblock in das Programm einbauen können
- SuS sollen das fertige Programm auf den BOB3 übertragen können
- SuS sollen zunächst vorgegebene kleine Änderungen an den Blöcken durchführen
- SuS sollen lernen, wie man die Farbe der LEDs verändern kann

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programm durchführen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden.

Soziale Kompetenzen: Gegenseitige Hilfestellung bei der Bedienung des Computers; Diskutieren über technische Sachverhalte.



Station B: Polizei-Blinklicht

An dieser Station programmieren die SuS zunächst ein einfaches Blinkprogramm mit dem BOB3: Das linke Auge wird rot eingeschaltet, es wird kurz gewartet und wieder ausgeschaltet. Die SuS erweitern das Programm dann so, dass beide Augen abwechselnd blinken und können dabei mit verschiedenen Farben und Geschwindigkeiten experimentieren. Die SuS lernen, wie man mit Befehls-Blöcken zur Änderung der Farbe und mit Befehls-Blöcken zur kurzen Verzögerung Blinkmuster erzeugen kann, indem sie diese zu Sequenzen verbinden. Zur Differenzierung können die SuS zusätzlich ihr Programm zu einem Polizeiblinklicht mit einer komplexeren Sequenz erweitern.

Schwierigkeitsgrad: Einfach ★ bis Mittel ★★

Zeitbedarf: ca. 15-30 Minuten



Voraussetzungen:

- SuS sollten die Station A erfolgreich bearbeitet haben

Lernziele:

- SuS sollen die Befehls-Blöcke zur Veränderung der Farbe der LEDs kennen lernen und verstehen
- SuS sollen die Befehls-Blöcke zur Verzögerung (Warte ms) kennen lernen
- SuS lernen wie man Befehls-Blöcke zu Sequenzen kombiniert, um damit Blinkeffekte mit den LEDs zu realisieren
- SuS sollen ihre Kenntnisse vertiefen, wie man die Farbe der LEDs verändern kann

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm.



Station C: Taschenlampe

An dieser Station soll der Bob als Taschenlampe programmiert werden. Bei Berührung des linken Arms sollen die Augen-LEDs und später auch die Körper-LEDs weiß eingeschaltet werden. Mit Berührung des rechten Arms werden dann alle LEDs wieder ausgeschaltet. Dazu lernen die SuS das Konzept von Verzweigungen und Bedingungen kennen. Sie verwenden einen „Wenn-Mache“-Block und kombinieren diesen mit einem Block für die Eingabewerte der Touch-Sensoren, um auf Benutzer-Eingaben reagieren zu können. Zur Differenzierung muss die Lösung bei Karte 5 von den SuS selbstständig aus den Hinweisen erarbeitet werden.

Schwierigkeitsgrad: Einfach ★ bis Mittel ★★

Zeitbedarf: ca. 15-30 Minuten

Voraussetzungen:

- SuS sollten die Station A erfolgreich bearbeitet haben

Lernziele:

- SuS sollen die Befehls-Blöcke zum Ein und Ausschalten der LEDs kennen lernen und verstehen
- SuS sollen den Befehls-Block zur bedingten Ausführung (Wenn-Mache-Block) kennen lernen.
- SuS sollen den Touch-Sensor-Block kennen lernen, mit dem sie die Sensoren an den Armen verwenden können.

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm.



Station D: Regenbogen

An dieser Station experimentieren die SuS mit den sechs Touch-Sensoren von BOB3: Je nachdem welcher Touch-Sensor ausgelöst wird, sollen die Augen-LEDs in einer bestimmten Farbe leuchten. Wird danach ein anderer Touch-Sensor ausgelöst, soll die Farbe sich verändern. Die SuS können dazu ihre sechs Lieblingsfarben verwenden. Sie erwerben vertiefte Kenntnisse in der Benutzung des „Wenn-Mache“-Blocks. Sie lernen, wie dieser Block zum „Wenn-Mache-Sonst-Wenn-Mache“ Block erweitert werden kann, und wie man dadurch auf verschiedene Fälle unterschiedlich reagieren kann.

Schwierigkeitsgrad: Mittel ★★

Zeitbedarf: ca. 15-30 Minuten



Voraussetzungen:

- SuS sollten die Station A erfolgreich bearbeitet haben. Erworbene Kenntnisse aus Station C sind von Vorteil, jedoch nicht grundsätzlich erforderlich.

Lernziele:

- SuS sollen den Befehls-Block zur bedingten Ausführung (Wenn-Mache-Block) kennen lernen und diesen zu Fallunterscheidungen erweitern können.
- SuS sollen ihre Kenntnisse vertiefen, wie man die Farbe der LEDs verändern kann

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm.



Station E: Alarmsensor / Alarmanlage

An dieser Station wird der Bob zunächst als Alarmsensor und dann als Alarmanlage programmiert. Die SuS experimentieren mit dem IR-Sensor und lernen, dass der Bob auch berührungslos Objekte, wie zum Beispiel eine Hand, wahrnehmen kann. Diese Station ist speziell für leistungsstarke SuS konzipiert. Die SuS erwerben Kenntnisse in der Benutzung des „Wenn-Mache-Sonst“-Blocks. Sie lernen Sensorwerte kennen und vergleichen diese mit einem Referenzwert. Später lernen sie auch Endlos-Schleifen und deren Abbruch kennen, um den Alarm-Zustand anzuzeigen und zu beenden.

Schwierigkeitsgrad: Mittel ★★ bis Hoch ★★★

Zeitbedarf: ca. 30-60 Minuten

Voraussetzungen:

- SuS sollten die Stationen B und C erfolgreich bearbeitet haben. Erworbene Kenntnisse aus der Station D sind von Vorteil.

Lernziele:

- SuS sollen den Befehls-Block zur bedingten Ausführung (Wenn-Mache-Sonst-Block) kennen lernen.
- SuS lernen den IR Sensor kennen und verwenden den Wert des Sensors in Kombination mit einem ‚größer als‘ Block.
- SuS lernen den ‚Wiederhole unendlich oft‘ Block kennen und setzen ihn zur Anzeige des Alarms ein.
- SuS sollen die Befehls-Blöcke zur Verzögerung (Warte ms) kennen lernen um damit Blinkeffekte mit den LEDs zu realisieren
- SuS sollen ihre Kenntnisse vertiefen, wie man die Farbe der LEDs verändern kann
- SuS lernen den Block ‚Schleife abbrechen‘ kennen, um damit eine Rücksetzfunktion für den Alarm zu implementieren.
- SuS erhalten einen kleinen Einblick in die Funktionsweise einer logischen UND und einer logischen ODER Verknüpfung.

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen; Selbstständige Kombination vorgegebener Funktionsblöcke.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden; Sensorwerte aus der realen Welt mit digitalen Mitteln verarbeiten.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm; gemeinsames Überlegen und Diskutieren über technische Problemstellungen.



Station F: Zufall

An dieser Station programmieren die SuS Bob als Zufallsgenerator für Ja/Nein Entscheidungen (Zufallsexperiment Münzwurf). Nach Berührung des rechten Arms blinken die LEDs zunächst in Gelb. Je nach Ergebnis des Zufallsexperiments leuchten die LEDs anschließend in Rot oder in Grün auf. Die SuS lernen dabei eine Methode zur wiederholten Ausführung von Programmblöcken kennen, und erwerben erste Kenntnisse über die Funktion zur Generierung von Zufallszahlen. Zum Abschluss erstellen die SuS eine Statistik über die Zufallsergebnisse.

Schwierigkeitsgrad: Mittel ★★

Zeitbedarf: ca. 30-60 Minuten



Voraussetzungen:

- SuS sollten die Stationen B und C erfolgreich bearbeitet haben.

Lernziele:

- SuS sollen den Befehls-Block zur Wiederholung (Wiederhole-N-mal-Block) kennen lernen und diesen zur wiederholten Ausführung von Programmblöcken nutzen
- SuS lernen den Block zur Generierung von Zufallszahlen kennen.
- SuS sollen ihre Kenntnisse darin vertiefen, für welche Zwecke Wenn-Mache-Blöcke verwendet werden können.
- SuS erleben die Unvorhersagbarkeit von Zufallsereignissen.
- SuS verwenden eine Strichliste zur Auswertung eines Zufallsexperiments.

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen; Daten, Häufigkeiten und Wahrscheinlichkeiten.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden; Statistiken zur Auswertung von Zufallsereignissen verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm; gemeinsames Überlegen und Diskutieren über technische und mathematische Problemstellungen; Organisation der Aufgabenteilung beim gemeinsamen Experimentieren.



Station G: Würfel

An dieser Station programmieren die SuS zunächst ein einfaches Würfelprogramm. Sie weisen einer Variablen den Zufallswert zu und verwenden den Wiederhole-N-mal-Block, um mit den LEDs das Ergebnis des Wurfs anzuzeigen. Im zweiten Teil wird das Programm so abgeändert, dass ein Würfergebnis zwischen Eins und Vier simultan mit den LEDs angezeigt wird. Zur Fallunterscheidung verwenden die SuS einen Wenn-mache-sonst-wenn-mache-Block. Zum Abschluss experimentieren die Schüler mit dem Zufallsgenerator und erstellen eine Statistik über die Ergebnisse.

Schwierigkeitsgrad: Hoch ★★★

Zeitbedarf: ca. 45-90 Minuten



Voraussetzungen:

- SuS sollten die Stationen D und F erfolgreich bearbeitet haben.

Lernziele:

- SuS lernen das Konzept einer Variablen kennen.
- SuS vertiefen ihre Kenntnisse über den Block zur Generierung von Zufallszahlen.
- SuS vertiefen ihre Kenntnisse über den Befehls-Block zur Wiederholung.
- SuS verwenden den Wenn-mache-sonst-wenn-mache-Block zur Fallunterscheidung.
- SuS legen eine Strichliste an und machen sich dabei bewusst, dass das nächste Ergebnis nicht vorhersagbar ist.
- SuS vergleichen ihre Ergebnisse und erleben dabei zufällige, statistische Schwankungen.

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen; Daten, Häufigkeiten und Wahrscheinlichkeiten; Simultanerfassung von Mengen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden; Statistiken zur Auswertung von Zufallsereignissen verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm; Organisation der Aufgabenteilung beim gemeinsamen Experimentieren.



Station H: Foto-Bob

An dieser Station wird Bob so programmiert, dass er den Selbstauslöser und das Blitzlicht von einem Fotoapparat simuliert. Am Anfang zeigt Bob die Berührung der Arme an. Bei Auslösung des Sensors wird das Auge auf der jeweiligen Seite gelb eingeschaltet. Anschließend verwenden die SuS den UND-Operator um bei gleichzeitiger Berührung beider Arme den Selbstauslöser zu starten, der Selbstauslöser wird dabei durch eine eigene Funktion realisiert. In dieser Funktion wird ein Wiederhole-N-mal-Block verwendet, um die Augen in der Wartezeit schnell Blinken zu lassen. Anschließend programmieren die SuS eine weitere Funktion um das Blitzlicht zu simulieren: Dazu werden alle LEDs kurz, weiß eingeschaltet und anschließend wieder ausgeschaltet.



Schwierigkeitsgrad: Hoch ★★★

Zeitbedarf: ca. 45-90 Minuten

Voraussetzungen:

- SuS sollten die Station E erfolgreich bearbeitet haben. Erworbene Kenntnisse aus weiteren Stationen sind von Vorteil, jedoch nicht grundsätzlich erforderlich.

Lernziele:

- SuS sollen den Befehls-Block zur bedingten Ausführung zur Reaktion auf Eingaben durch die Arm-Sensoren verwenden.
- SuS reflektieren den Zweck des logischen ‚Und‘-Operators.
- SuS lernen die Deklaration und den Aufruf eigener Funktionen kennen und setzen diese Technik zur Strukturierung eigener Programme ein.
- SuS vertiefen ihre Kenntnisse in der Verwendung des Wiederhole-N-mal-Blocks.

Prozessbezogene Kompetenzen: Bedienung der Oberfläche; Compilieren eines Programms; Übertragung der Software auf den Roboter; Änderungen im Programmtext durchführen.

Konzeptbezogene Kompetenzen: Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen und verstehen und bewusst nutzen; eine strukturierte algorithmische Sequenz zur Lösung eines Problems verwenden.

Soziale Kompetenzen: Gegenseitige Unterstützung bei der Suche nach Fehlern im Programm; gemeinsames Überlegen und Diskutieren über technische und mathematische Problemstellungen.

