

Lernzirkel: Spielerisch programmieren lernen



medienistik-lab
NR.1 / 2024



VORWORT



Die Einführung ins Programmieren ist oft eine recht langweilige Angelegenheit. Dieser Lernzirkel, gedacht für den **Informatikunterricht in den Klassen 5 und 6**, soll das Thema etwas spannender machen: An zehn verschiedenen Stationen, für die die Schülerinnen und Schüler jeweils 30 Minuten Zeit haben, stehen die **Faszination und der Spaß am Programmieren** im Mittelpunkt.

Darüber hinaus lernen die Schülerinnen und Schüler, wie Computer eigentlich „ticken“ und wie man sie durch Logik und Klarheit dazu bringt, Befehlen zu gehorchen.

Zu jeder Station habe ich ein **kurzes Video** aufgenommen, das die grundlegenden Funktionen der jeweiligen Hardware vermittelt und so den Einstieg in die Auseinandersetzung damit erleichtert.

Die Schülerinnen und Schüler arbeiten dabei **in Gruppen zu dritt**. Bei einer Klassengröße von 30 Kindern lässt sich der Lernzirkel somit gut in vier Doppelstunden durchlaufen.

Die Hardware zum Projekt kann größtenteils beim Kompetenzteam Düsseldorf ausgeliehen werden. Bei Bedarf schreiben Sie mir einfach eine E-Mail an info@medienistik.de.

Viel Spaß beim Lesen!

Tobias Hübner



Lernzirkel: Spielerisch Programmieren lernen

INHALT

- 02 *Vorwort*
- 03 *Inhaltsverzeichnis*
- 04 *Benötigte Hardware*
- 05 *Ablaufplan*
- 10 *Stationsbeschreibungen*
- 21 *Laufzettel für die Schülerinnen und Schüler*



Benötigte Hard- und Software

Die hier vorgestellten Stationen sind Vorschläge für Hard- und Software. Dabei handelt es sich um eine kleine Auswahl der am Markt erhältlichen Produkte, die je nach Bedarf beliebig verändert werden kann. Eine Möglichkeit wäre etwa der Einsatz der an Schulen sehr beliebten LEGO-Mindstorms Roboter. Auch andere Arten von programmierbaren Robotern oder Platinen können eingesetzt werden kann. Eine Übersicht findet sich z. B. [auf der Seite des Open Roberta Projekts](#).

1. Station: Programmieren lernen mit **Human Resource Machine**

Kurzbeschreibung:

Im Spiel Human Resource Machine werden dem Spieler eine Reihe von Rätseln präsentiert, die mit Hilfe einer Art Assembler-Programmiersprache gelöst werden müssen. Für Schulen ist das Spiel kostenlos.

Benötigte Hard- und Software:

Das Spiel Human Resource Machine läuft sowohl unter Windows als auch MacOS und Linux. Auch für viele mobile Geräte wie iPads und iPhones, Android-Tablets und -Smartphones sowie die Nintendo Switch ist das Spiel erhältlich. In der „Hour of Code“-Edition ist das Spiel sogar kostenlos. Mehr Infos dazu gibt es [auf der Website des Entwicklers](#).

2. Station: Programmieren lernen mit **Scottie Go!**

Kurzbeschreibung:

Scottie Go! Ist eine interessante Möglichkeit, Programmierbefehle mit Hilfe kleiner Bausteine aus Pappe zu visualisieren, die auf einem Spielbrett zusammengelegt und ab fotografiert werden können. So lassen sich einfach und auf spielerische Weise kleine Programme zu schreiben.

Benötigte Hard- und Software:

Der Hersteller bietet viele unterschiedliche Versionen von Scottie Go! an, die jedoch alle ähnlich aufgebaut sind. Speziell für Schulen ist zudem eine Education-Version erhältlich.

3. Station: Programmieren lernen mit **Scratch**

Kurzbeschreibung:

Scratch ist die wohl am einfachsten zu erlernende Programmiersprache, mit der sich in wenigen Minuten und ohne Vorkenntnisse ein funktionsfähiges Computerprogramm erstellen lässt. Auch wenn Scratch sehr leicht zu erlernen ist, handelt es sich dennoch um eine „richtige“ Programmiersprache, mit der sich auch komplexere Projekte realisieren lassen. Außerdem lässt sich eine Vielzahl unterschiedlicher Hardware mit Scratch ansteuern, etwa LEGO-Roboter oder Mikrocontroller wie der micro:bit.

Benötigte Hard- und Software:

Scratch ist kostenlos und läuft auf nahezu jeder verfügbaren Hardware, selbst auf Mini-Computern wie dem Raspberry Pi. Am einfachsten ist die Verwendung von Scratch direkt im Browser. Dazu muss man lediglich die Seite <https://scratch.mit.edu/> aufrufen und kann direkt loslegen. Wenn man Projekte speichern möchte, kann man ein kostenloses Konto anlegen. Aber auch ohne Registrierung kann man direkt loslegen. Zusätzlich lassen sich an der Station [Lernkarten](#) auslegen.

4. Station: Computer verstehen mit **Turing Tumble**

Kurzbeschreibung:

Turing Tumble ist ein Bausatz für einen mechanischen Computer mit Murmelantrieb. Das Spiel ist eine Weiterentwicklung eines ganz ähnlichen Spielzeugs namens Digi Comp aus den 60er-Jahren. Das beiliegenden Comic erzählt eine spannende Geschichte und enthält zahlreiche Aufgaben mit steigendem Schwierigkeitsgrad. Kinder und Jugendliche lernen so spielerisch die Funktionsweise von Computern ganz ohne Computer oder Bildschirm.

Benötigte Hard- und Software:

Das Spiel Turing Tumble ist u.a. hier erhältlich. Beim Einsatz des Spiels in der Schule lohnt sich der Kauf des „Kugelsammlers“. Dabei handelt es sich um ein Gerät zum Aufsammeln der Kugeln, das außerdem noch viele zusätzliche Kugeln erhält. Diese gehen nämlich erfahrungsgemäß schnell verloren. Auch alle anderen Teile des Spiels lassen sich übrigens einzeln nachkaufen, wenn etwas kaputt oder verloren gehen sollte.



5. Station: Programme steuern mit **Makey Makey**

Kurzbeschreibung:

Der Makey Makey ist keine klassische „Programmier-Hardware“, besitzt aber einen großen Fun-Faktor und ist allein schon deshalb eine Bereicherung für das Unterrichtsvorhaben. An die kleine Platine lassen sich mithilfe mitgelieferter Krokodilklemmen leitende Gegenstände wie Obst, Alufolie oder Knete verbinden, die dann bei Berührung die Funktion einer Taste übernehmen können. Auch ungeübte Kinder können haben beim Makey Makey schnell Erfolgserlebnisse und können sich dann an der Station damit beschäftigen, einen kreativen Game-Controller zu bauen.

Benötigte Hard- und Software:

Der Makey Makey ist bei vielen Händlern erhältlich. Es ist auch ein Klassenset erhältlich, das jedoch unverhältnismäßig teuer ist. Neben den beigefügten Kabeln sollte an der Station noch Knete (Playdoh) und Alupapier ausgelegt werden. Zusätzlich wird auch ein Computer mit USB-Anschluss benötigt, auf dem idealerweise ein Spiel läuft, das sich mit den Pfeiltasten sowie der Leertaste steuern lässt, etwa ein Tetris-Klon oder ein Jump-and-Run-Spiel. Ich selbst verwende gerne das Spiel [VVVVV](#).

6. Station: Roboter programmieren mit **mTiny**

Kurzbeschreibung:

Der Makeblock Tiny ist dafür konzipiert, schon in der Grundschule eingesetzt zu werden. Er wird mit großen Pappkarten zum Bau einer kleinen Stadt geliefert und bietet viele verschiedene Spielmodi, die Grundkenntnisse des Programmierens vermitteln.

Benötigte Hard- und Software:

Wenn man den mTiny kauft, wird jedes für die Station benötigte Zubehör mitgeliefert. Da der Roboter mit dem mitgelieferten Stift sowie den ebenfalls mitgelieferten Karten programmiert wird, ist auch kein Computer oder Smartphone notwendig.

7. Station: Programmieren lernen mit **Mouse Mania**

Kurzbeschreibung:

Bei *Mouse Mania* handelt es sich um ein klassisches Brettspiel, das ohne elektronische Komponenten auskommt. Im Spiel steuern bis zu vier Spielerinnen und Spieler eine Maus, die sich über ein Spielfeld mit Schachbrettmuster bewegen und deren Bewegungen mit Hilfe von Spielkarten „programmiert“ werden. Ziel des Spiels ist es, die Mäuse um Hindernisse herum zu den ausliegenden Käseecken zu bewegen

Benötigte Hard- und Software:

Außer dem Brettspiel selbst ist keine weitere Hard- oder Software notwendig.

8. Station: Roboter programmieren mit **mBot**

Kurzbeschreibung:

Der *mBot* ist ein Roboter, der mit Hilfe eines mobilen Endgeräts und einer einfachen, blockbasierten Sprache programmiert werden kann. Wie auch beim *mTiny* ist der Einstieg sehr einfach, der *mBot* bietet jedoch vielfältigere Möglichkeiten, da er einen Ultraschall- sowie einen Line-Follower-Sensor besitzt. Durch die modulare Bauweise können zudem weitere Komponenten angeschlossen werden. Mittlerweile ist auch der *mBot 2* erhältlich, der zusätzlich einen Farbbildschirm und weitere Sensoren enthält.

Benötigte Hard- und Software:

Neben dem Roboter selbst wird ein Smartphone oder Tablet mit der kostenlosen App des Herstellers zur Programmierung benötigt.

9. Station: Roboter programmieren mit **ozobot**

Kurzbeschreibung:

Der ozobot ist ein winziger Roboter in der Größe einer Mozartkugel, der selbständig einer Linie auf dem Boden folgen kann. Programmiert wird er ganz analog durch den Einsatz von Farbcodes, die den Roboter beispielsweise zum Umkehren bewegen oder die integrierten LEDs aufleuchten lassen.

Benötigte Hard- und Software:

An der Station benötigt man neben dem ozobot selbst nur einige dicke Filzstifte in den Farben schwarz, blau, rot und grün. Zur Erklärung ist es zudem hilfreich, einige der [vom Hersteller zur Verfügung gestellten Arbeitsblätter](#) auszudrucken, die die Funktionsweise des Roboters erklären.

10. Station: Programmieren mit **Calliope mini**

Kurzbeschreibung:

Der Calliope Mini kann mit Hilfe eines iPads oder Computers programmiert werden. Die Platine bietet neben Knöpfen auch ein kleines 5x5 Pixel großes Display. Auch die Ansteuerung von externen Motoren und Sensoren ist möglich.

Benötigte Hard- und Software:

Neben dem Calliope mini wird an der Station ein Computer zur Programmierung benötigt. Das kann entweder ein Smartphone oder Tablet oder auch ein Laptop sein. Ich habe zusätzlich ein Calliope Fischertechnik-Set ausgelegt, das ist aber optional.

Bonus-Station: Programmieren mit **Rocky's Boots (1982)**

Kurzbeschreibung:

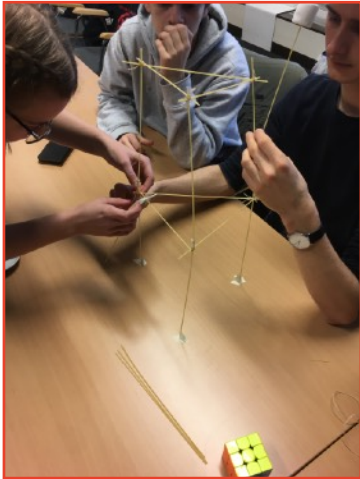
Rocky's Boots ist eines der ersten Spiele, das bereits 1982 die Prinzipien des Programmierens auf unterhaltsame Art vermittelt hat. Die darin gelehrtten Inhalte sind noch immer aktuell und für die Kinder ist es immer ein großer Spaß, Retro-Games aus einer Zeit lange vor ihrer Geburt auszuprobieren.

Benötigte Hard- und Software:

Das Spiel kann [auf der Seite archive.org direkt im Browser](#) gespielt werden oder heruntergeladen und mit Hilfe eines Emulators gespielt werden.

Einsteig

Zu Beginn des Lernzirkels muss die Einteilung in 3-er Gruppen erfolgen. Ich verwende hierzu gerne die kostenlos herunterladbaren [Karten der „Grundschtante“](#).



Um sich als Team einzufinden, habe ich die erste Doppelstunde damit verbracht, eine **Marshmallow-Challenge** zu veranstalten. Die Regeln der Marshmallow Challenge sind einfach: die Dreier-Teams haben 18 Minuten Zeit, um aus einem Meter Klebeband, 1 Meter Schnur und 20 ungekochten Spaghetti-Nudeln einen Turm zu bauen, auf dessen Spitze ein Marshmallow befestigt ist.

Anschließend wird gemessen, in welcher Höhe sich der Marshmallow befindet. Das Team, dessen Marshmallow am weitesten von der Tischoberfläche entfernt ist, hat gewonnen.

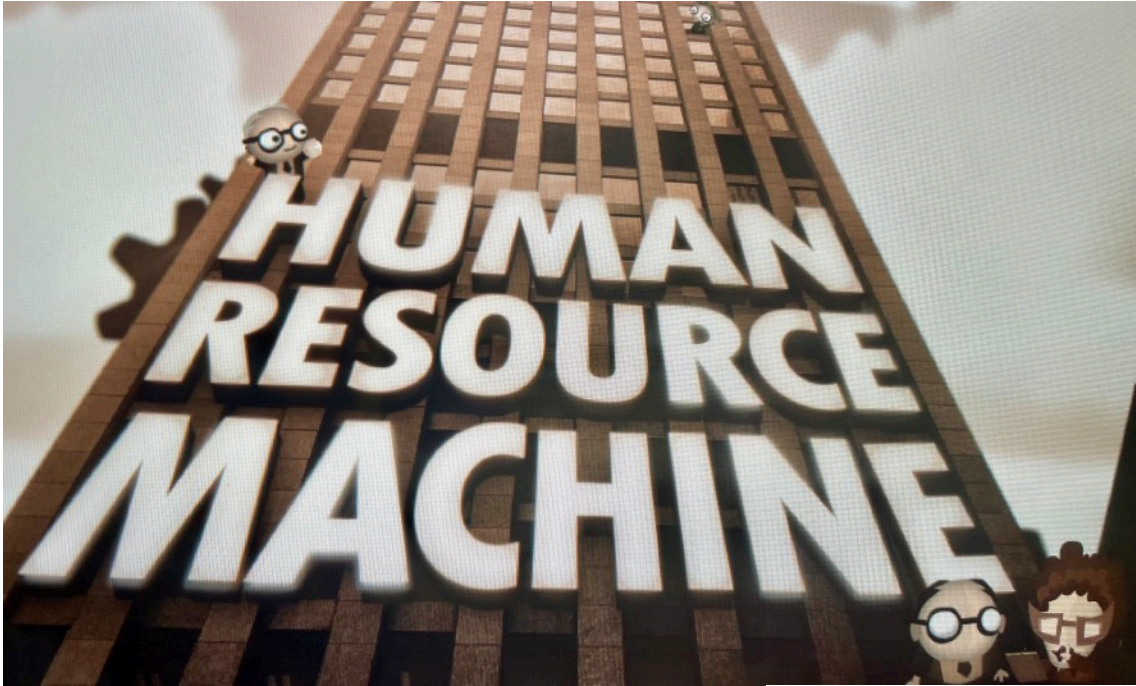
In einem TED-Talk erklärt Tom Wujec, wie er über 70 unterschiedlichsten Teams diese Aufgabe gestellt hat und welche Lehren er daraus gezogen hat. [Das Video](#) habe ich im Anschluss an die Challenge mit meinen Schülerinnen und Schülern angeschaut und besprochen.

Beispielsweise schnitten Kindergarten-Kinder mit am besten ab, da sie weniger darum kämpfen, wer die Leitung in der Gruppe hat und verschiedenste Prototypen bauen und testen, statt die eine perfekte Lösung zu finden. Eine weitere interessante Erkenntnis ist, dass eine hohe Belohnung für den Gewinner nicht dazu führt, dass die einzelnen Teams besser abschneiden - im Gegenteil! Als ein Preis in Höhe von \$10.000 USD ausgesetzt wurde, schaffte es keines der zehn teilnehmenden Teams, einen tragfähigen Turm zu bauen.

Die Marshmallow Challenge wurde abschließend mit einigen **Fragen** evaluiert, z. B.:

- Gab es jemanden, der im Team die Führung übernommen hat? Wenn ja, wie wurde diese Person ausgewählt?
- Haben sch alle im Team zu gleichen Teilen an der Lösung des Problems beteiligt?
- Was hätte man rückblickend besser machen können?
- War der Zeitdruck hilfreich oder hinderlich?

Programmieren lernen mit Human Resource Machine



Materialien:

- iPad

Anleitung:

*Spielt das Spiel so weit
ihr kommt und bewerte
anschließend den Spaßfaktor, Lernfaktor
und den Schwierigkeitsgrad!*



[https://youtu.be/
XkOn7XraTms](https://youtu.be/XkOn7XraTms)

Programmieren lernen mit Scottie Go!



Materialien:

- *Scottie Go! Klassenset*
- *iPhone mit Scottie Go App*

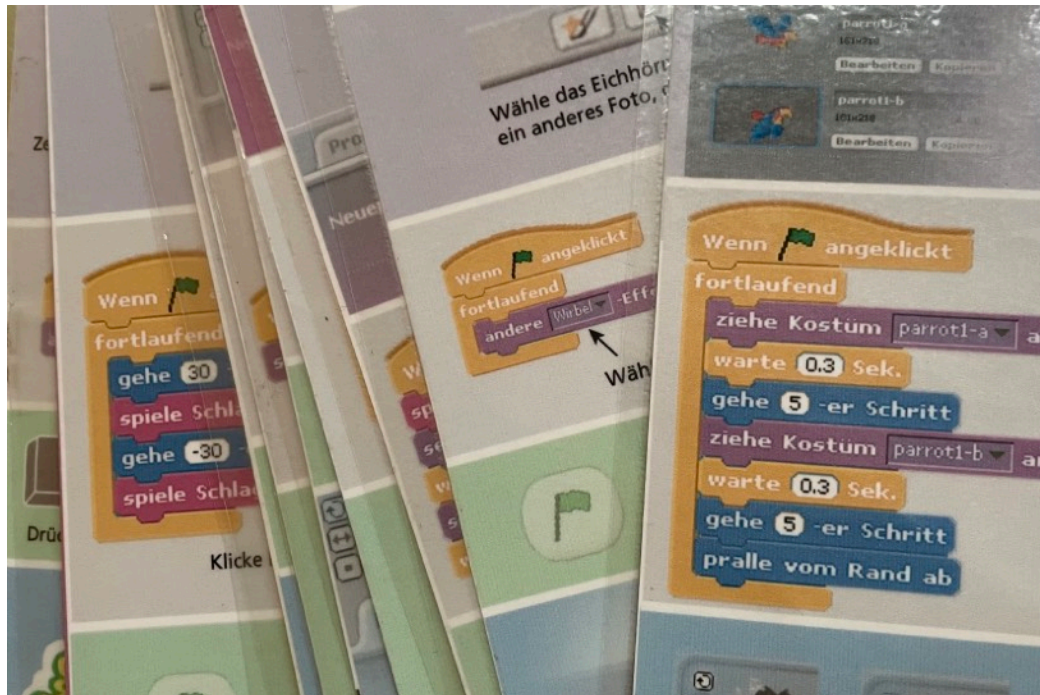


Anleitung:

<https://youtu.be/UWdt-LZM-PQ>

*Folge dem App-Tutorial
und programmiere deinen Roboter.
Bewerte anschließend den Spaßfaktor,
Lernfaktor und den Schwierigkeitsgrad!*

Programmieren lernen mit Scratch



Materialien:

- *PiTop mit Raspberry Pi*
- *Scratch-Lernkarten*



Anleitung:

*Schreibe ein oder mehrere Programme mit Scratch.
Bewerte anschließend den Spaßfaktor,
Lernfaktor und den Schwierigkeitsgrad!*

<https://youtu.be/JFSeY61xymg>

Computer verstehen mit Turing Tumble



Materialien:

- *Spiel Turing Tumble*



Anleitung:

Spielen Sie das Spiel Turing Tumbler. Bewerte anschließend den Spaßfaktor, Lernfaktor und den Schwierigkeitsgrad!

<https://youtu.be/qDHB5I3h5eU>

Programme steuern mit Makey Makey



Materialien:

- *Makey Makey Set*
- *Laptop mit dem Spiel
VVVVV*



Anleitung:

[https://youtu.be/
BKSQgDytpw4](https://youtu.be/BKSQgDytpw4)

*Baue eine originelle
Steuerung mit dem Makey Makey.
Bewerte anschließend den Spaßfaktor,
Lernfaktor und den Schwierigkeitsgrad!*

Roboter programmieren mit **mTiny**



Materialien:

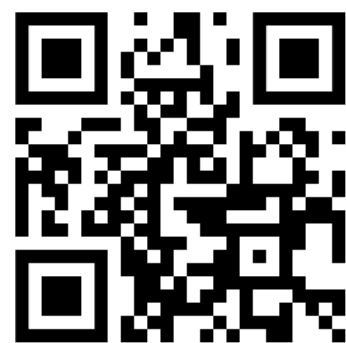
- *mTiny Set*

Anleitung:

Spiele mit dem mTiny.

Bewerte anschließend

den Spaßfaktor, Lernfaktor und den Schwierigkeitsgrad!



<https://youtu.be/ghlxcjsEi-c>

Programmieren lernen mit Mouse Mania



Materialien:

- *Mouse Mania Brettspiel*



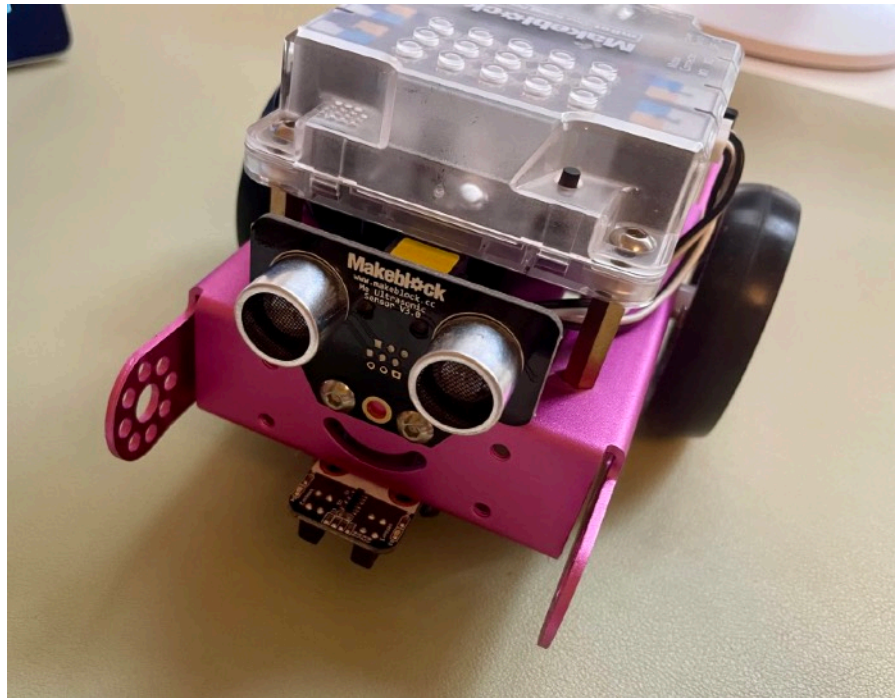
Anleitung:

*Programmiere und teste
den Edison-Roboter.*

[https://youtu.be/
M2fwnRDmnu8](https://youtu.be/M2fwnRDmnu8)

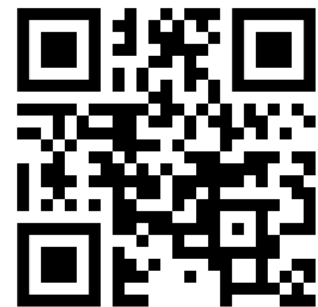
*Bewerte anschließend den Spaßfaktor,
Lernfaktor und den Schwierigkeitsgrad!*

Roboter programmieren mit mBot



Materialien:

- mBot
- iPad

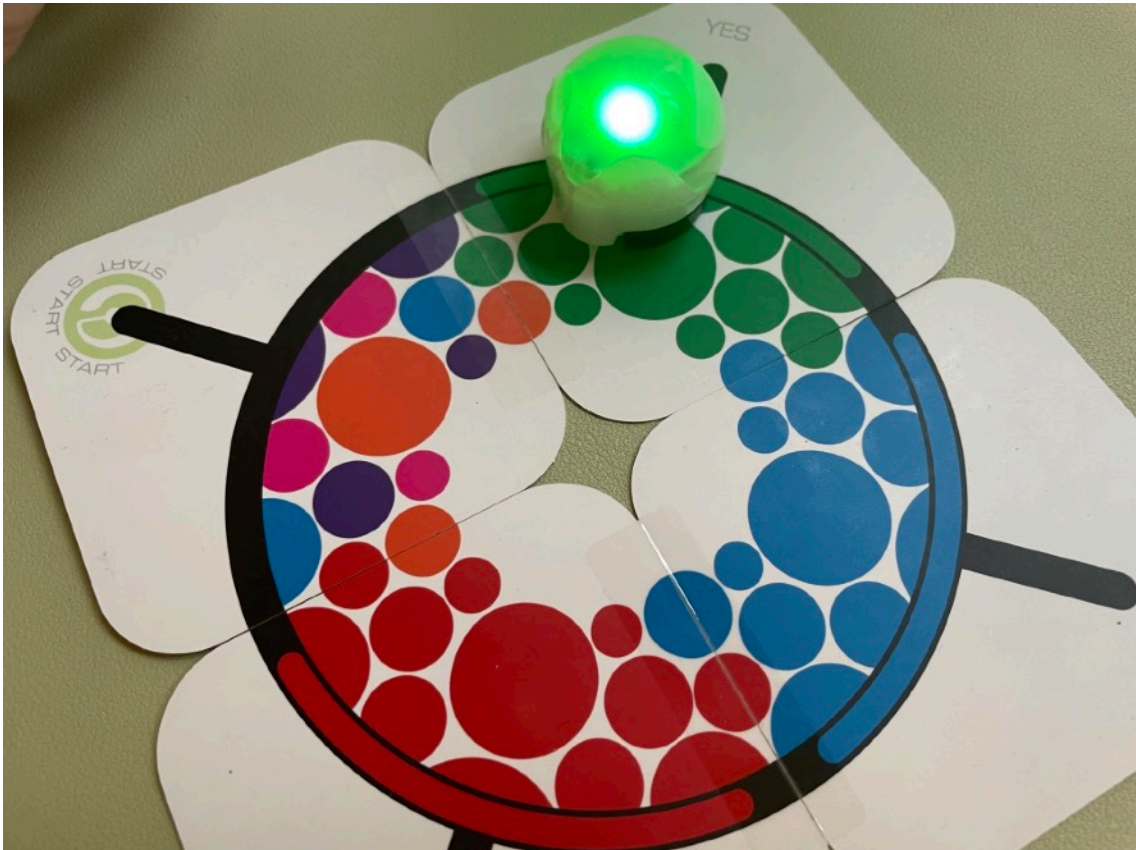


Anleitung:

Programmiere und teste den mBot. Bewerte anschließend den Spaßfaktor, Lernfaktor und den Schwierigkeitsgrad!

<https://youtu.be/CLznAWKyr28>

Roboter programmieren mit ozobot



Materialien:

- ozobot
- Stifte, Papier, Anleitungen



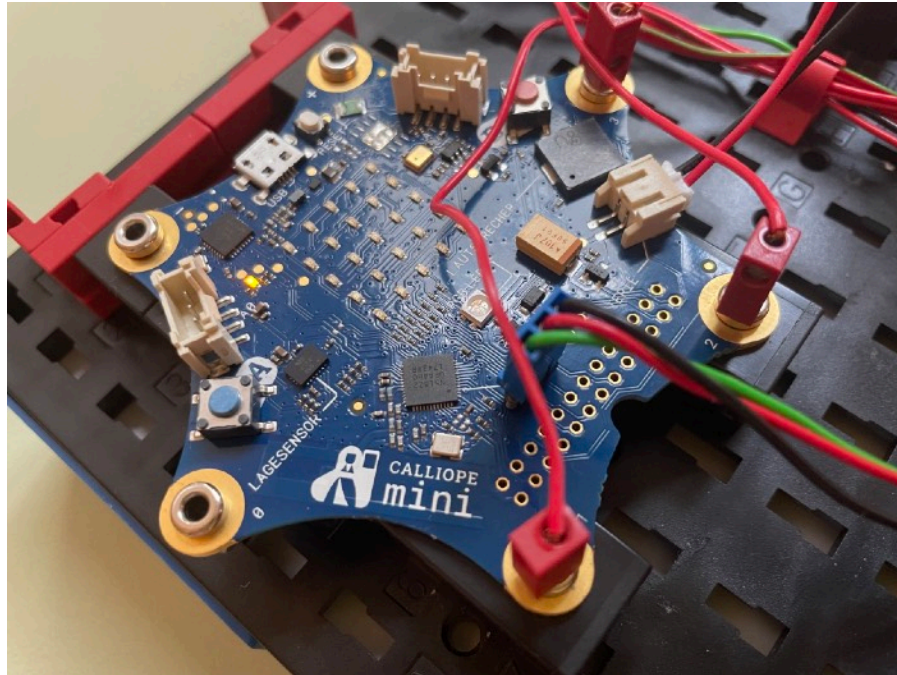
Anleitung:

Programmiere den Ozobot.

*Bewerte anschließend den Spaßfaktor,
Lernfaktor und den Schwierigkeitsgrad!*

[https://youtu.be/
QbiRcNtXUA0](https://youtu.be/QbiRcNtXUA0)

Programmieren lernen mit Calliope mini



Materialien:

- *Calliope mini*
- *Fischertechnik-Set mit Calliope*
- *Laptop*



<https://youtu.be/v-M5F6cfFdU>

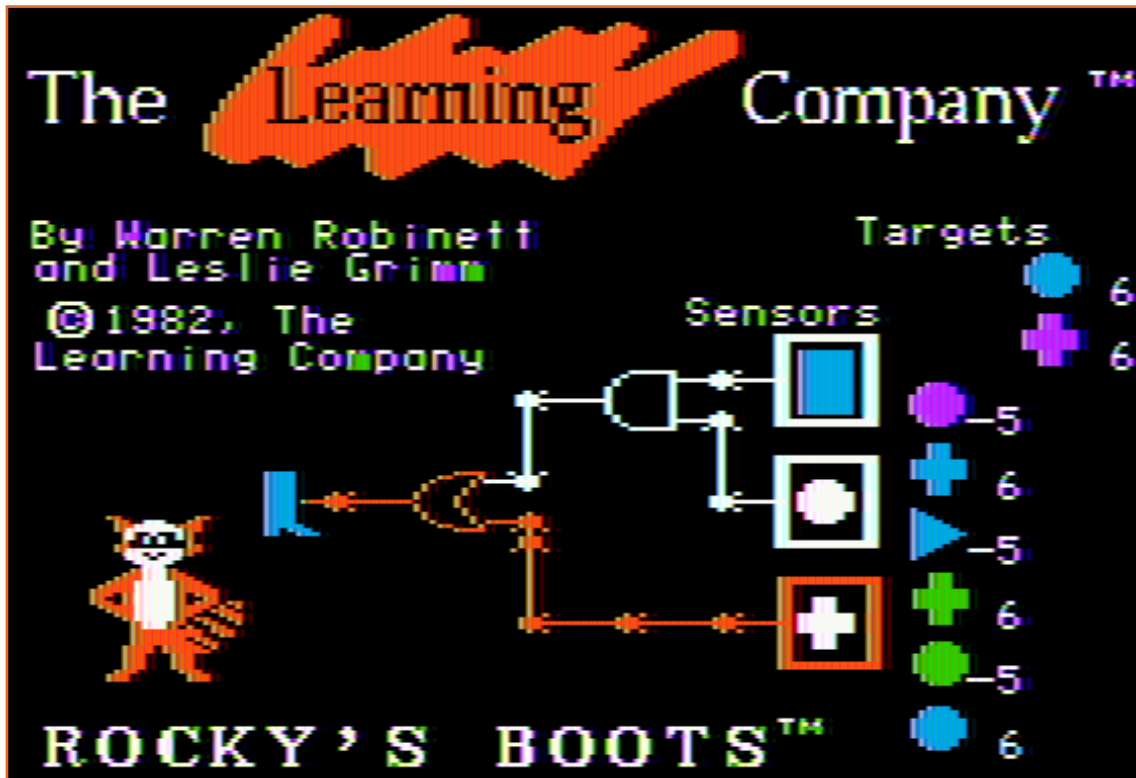
Anleitung:

Programmiere den Calliope Mini.

Bewerte anschließend den Spaßfaktor,

Lernfaktor und den Schwierigkeitsgrad!

Bonus: Programmieren mit Rocky's Boots (1982)



Materialien:

- *Game Boy (mit Raspberry Pi)*
- *Rocky's Boots Spiel*

Anleitung:

Spiel das Spiel und bewerte anschließend den Spaßfaktor, Lernfaktor und den Schwierigkeitsgrad!

Stationen-Lernen Programmieren

Name: _____

Name der Station:	Ich habe die Station besucht am:	Bewertung Spaßfaktor (0-10):	Bewertung Lernfaktor (0-10):	Bewertung Schwierigkeit (0-10):
Human Resource Machine				
Scottie Go!				
Scratch				
Turing Tumble				
Makey Makey				
mTiny				
Mouse Mania				
mBot				
ozobot				
Calliope mini				

