

OZOBOT IN DER SCHULE

CODING offline | online

Sandra.Mueller@brk.nrw.schule
Tom.Schardt@brk.nrw.schule
01/2024

Inhalt

- 1 Coding in der Grundschule
- 2 Klassisches Offlinecoding
- 3 Offlinecoding mit dem **Ozobot**
- 4 Onlinecoding mit dem **Ozobot**
- 5 Konkrete Einbindung in den Unterricht

Coding in der Grundschule

Das machen Sie seit Jahren....

- **Algorithmen** findet man überall im Alltag

- Rezepte
- Schriftliche Rechenverfahren
- Bausätze
- Brettspiele
- Zähneputzen

**Bruschetta
italien style**

für 3 - 4 Personen

Zutaten:

- 4-5 Tomaten
- 1 Bund frisches Basilikum
- 8 Eßl. Balsamico (weiß)
- 5 Eßl. Olivenöl
- 1 Knoblauchzehe
- 1 Baguette Pierre von Huth
- Prise grobes Meersalz
- Prise Pfeffer aus der Mühle
- Prise Zucker

😊 einfach ⌚ ca. 20 Minuten

Zubereitung:

1. Die Tomaten waschen und in kleine Würfel schneiden.
2. Den Basilikum und den Knoblauch sehr fein hacken.
3. In der Zwischenzeit Baguette Pierre bei 180 °C im Ofen knusprig backen, oder in ca. 2 cm dicke Scheiben schneiden und tosten.
4. Alle Zutaten in einer kleinen Schüssel mit dem Olivenöl und dem Balsamico gründlich vermengen. Ein paar Minuten durchziehen lassen.
5. Das Baguette Pierre aus dem Ofen holen. Es sollte kross gebacken sein. Das Brot in Scheiben schneiden und mit etwas Olivenöl bestreichen und mit den marinierten Tomatenwürfeln belegen.
6. Das Bruschetta warm servieren.

Guten Appetit!

 werde Fan auf Facebook

 Bäckerei Huth



$$\begin{array}{r} 1645 : 7 = 235 \\ - 14 \\ \hline 24 \\ - 21 \\ \hline 35 \\ 35 \\ \hline 0 \end{array}$$

Coding in der Grundschule

Algorithmus – was ist das?

Einen eindeutigen Handlungsablauf bezeichnet man auch als **Algorithmus**.

Die **EINZIGEN** Bestandteile eines Handlungsablaufs / Algorithmus' sind:

- **Anweisung** - „hole zwei Eier“
- Hintereinanderausführung/**Sequenz** – „hole zwei Eier, schlage sie in einen Topf“
- **Bedingung** - „wenn Topf groß genug ist, mache weiter, sonst nicht“
- **Wiederholung** - „solange es keine gleichmäßige Masse ist, rühre mit Schneebesen“
- **Teilaufgabe** - „mit Butter braten“ (diese Teilaufgabe ist wieder ein Algorithmus)

Coding in der Grundschule

Warum dann auch noch explizit/digital?

- Weil es eh passiert:
Taschenrechner: $1 / 3 = 0,33333$ ABER
 $0,33333 * 3 = 0,9999$
- Weil man experimentieren kann, ohne etwas kaputt zu machen
- Weil die Lösungen kreativ und hoch individuell sind
- Weil es Spaß macht

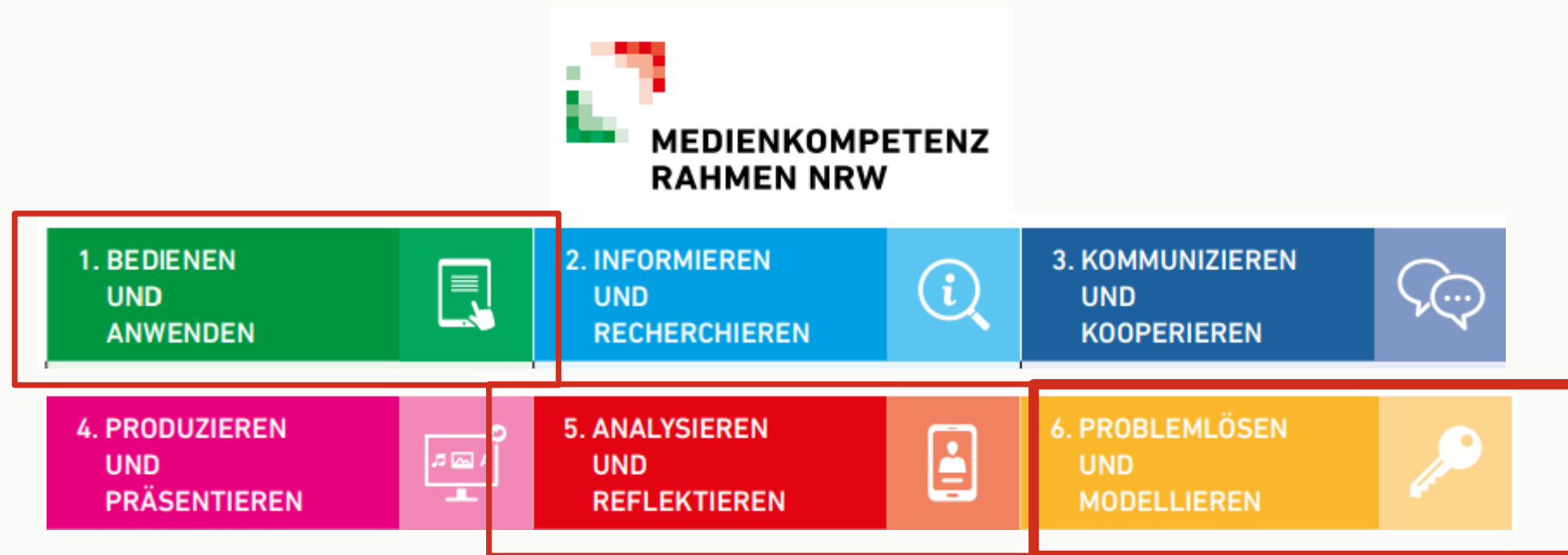
- weil wir müssen ;P



Coding in der Grundschule

Einbettung in den Medienkompetenzrahmen

Coding im Medienkompetenzrahmen NRW



Coding in der Grundschule

Einbettung in den Medienkompetenzrahmen



6. PROBLEMLÖSEN UND MODELLIEREN



6.1 Prinzipien der digitalen Welt

Grundlegende Prinzipien und Funktionsweisen der digitalen Welt identifizieren, kennen, verstehen und bewusst nutzen

6.2 Algorithmen erkennen

Algorithmische Muster und Strukturen in verschiedenen Kontexten erkennen, nachvollziehen und reflektieren

6.3 Modellieren und Programmieren

Probleme formalisiert beschreiben, Problemlösestrategien entwickeln und dazu eine strukturierte, algorithmische Sequenz planen; diese auch durch Programmieren umsetzen und die gefundene Lösungsstrategie beurteilen

6.4 Bedeutung von Algorithmen

Einflüsse von Algorithmen und Auswirkung der Automatisierung von Prozessen in der digitalen Welt beschreiben und reflektieren

Klassisches Offline-Coding

Exakte Wegbeschreibung: Hol bitte mal die Gummibärchen



- **Offline-Coding** ist ein Konzept, das das Programmieren ohne digitales Endgerät simuliert.
- Geht schon on Klasse 1/2
- Motivierender, spielerischer Zugang
- Algorithmen nutzen, um Abläufe in Teilschritte zu zerlegen und verschiedene Problemstellungen zu lösen.
- SuS lernen algorithmische Strukturen in ihrer Lebenswirklichkeit zu erkennen

Klassisches Offline-Coding Grundstruktur

Grundsetting könnte beispielsweise sein:



Schritt vorwärts ↑

Drehe rechts ↻

Drehe links ↻

nur
drei
Befehle

Verschiedene Aufgaben sind denkbar:

- ein **Schatz wird sichtbar** hingelegt
 - wie kann der Roboter gehen
 - welche anderen Möglichkeiten findest du?
 - es werden Hindernisse eingebaut
- ein **Schatz wird versteckt** hingelegt
 - einer legt die Befehle (schreibt den Code) der andere führt die Befehle aus

<https://nawitas.uni-koeln.de/unterricht/bs-1-2/offline-coding-klasse-1-2>

Klassisches Offline-Coding Analyse / Vereinfachung

Codes verkürzen: Schleifen einbauen

Schritt vorwärts↑

Schritt vorwärts↑

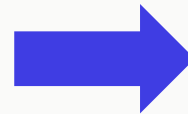
Schritt vorwärts↑

Schritt vorwärts↑

Schritt vorwärts↑

Schritt vorwärts↑

Wie kann man
diesen Code
verkürzen?



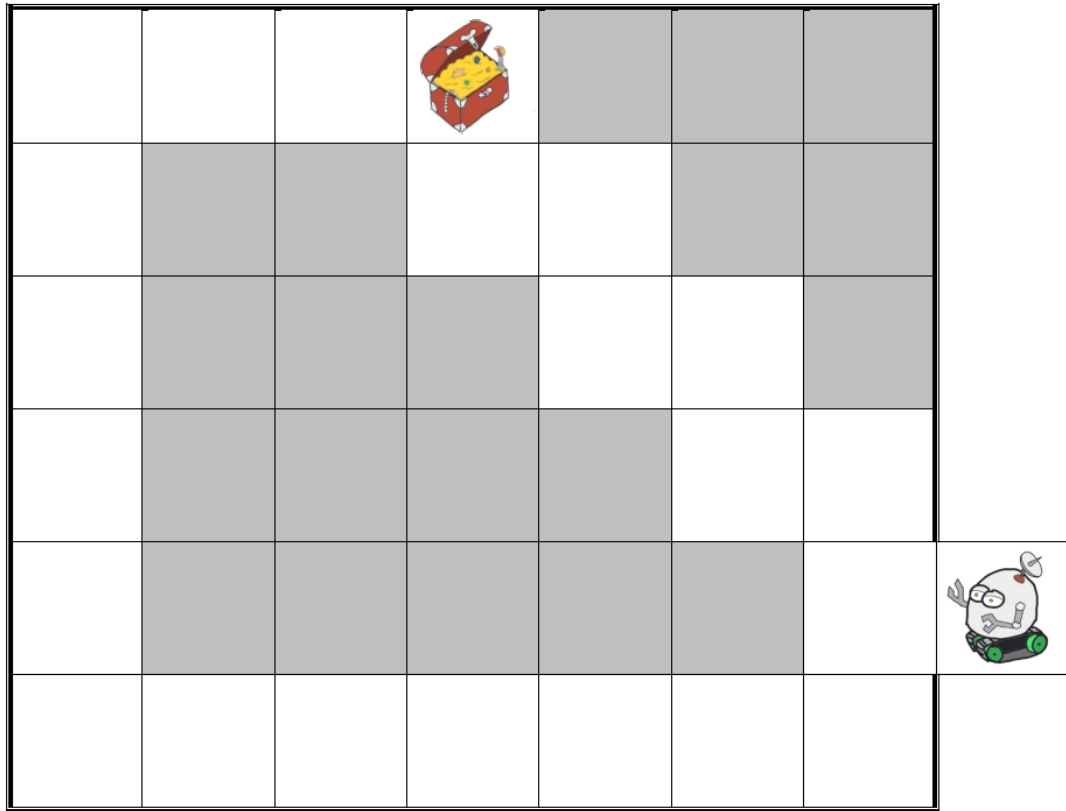
Schleife
einbauen

Wiederhole 6 mal

Schritt vorwärts↑

Klassisches Offline-Coding Blockbildung

Codes verkürzen: Schleifen einbauen



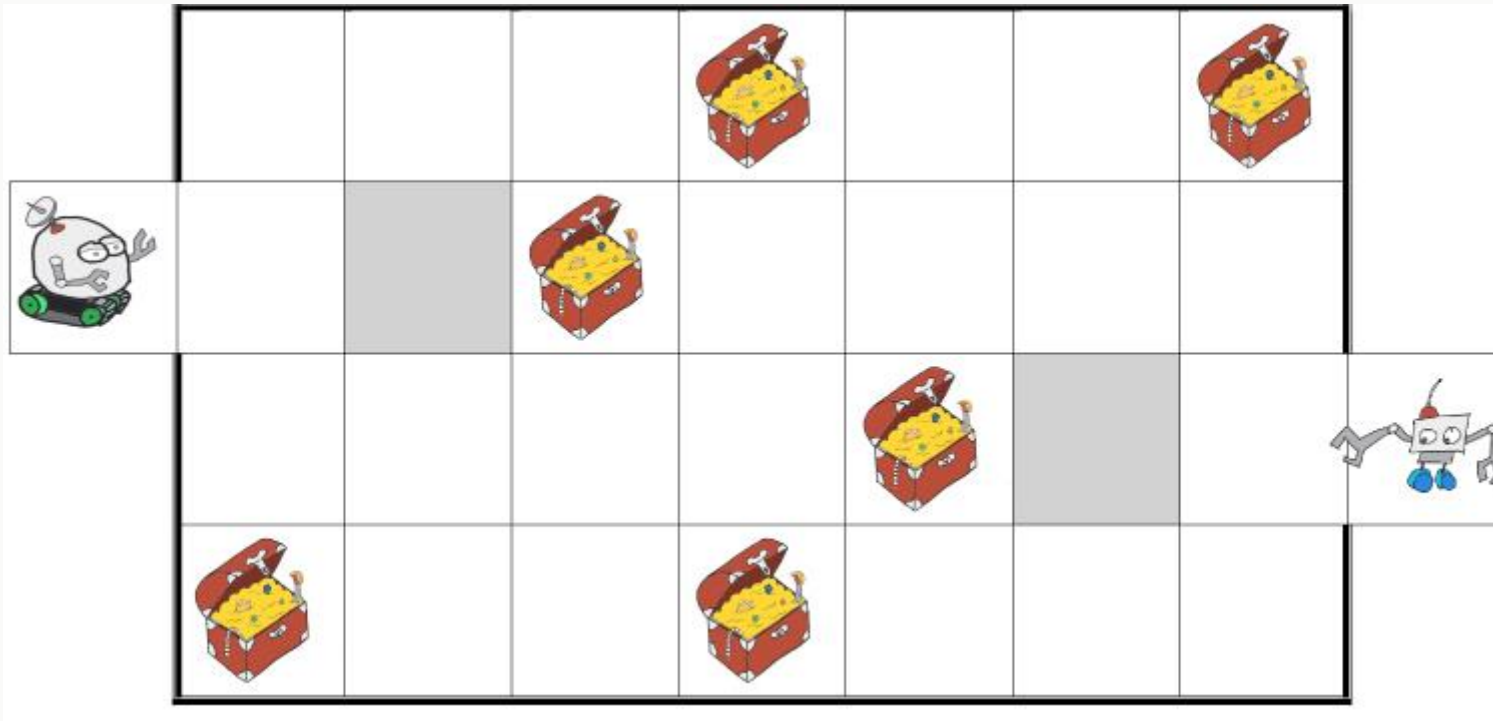
<https://nawitas.uni-koeln.de/unterricht/bs-1-2/offline-coding-klasse-1-2>



Klassisches Offline-Coding

Einplanung von Unwägbarkeiten

Die Roboter Arena



<https://nawitas.uni-koeln.de/unterricht/bs-1-2/offline-coding-klasse-1-2>

- Zwei Roboter bewegen sich durch die Arena, indem sie **abwechselnd** je einen Befehl ausführen.
- Eine Schleife gilt als ein Befehl.
- Ein Schatz wird eingesammelt, sobald ein Roboter auf dem Feld stehen bleibt. Dieser Schatz kann dann nicht mehr von dem anderen Roboter eingesammelt werden.
- Gewonnen hat der Roboter, der zuerst drei Schätze eingesammelt hat.

Offline Coding mit dem Ozobot

Kalibrierung

1



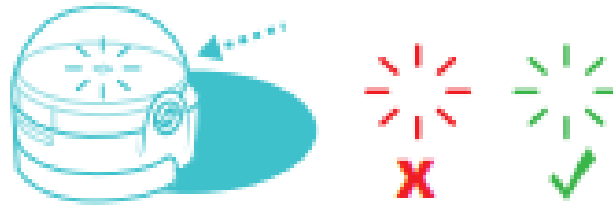
2



3



4



Offline Coding mit dem Ozobot

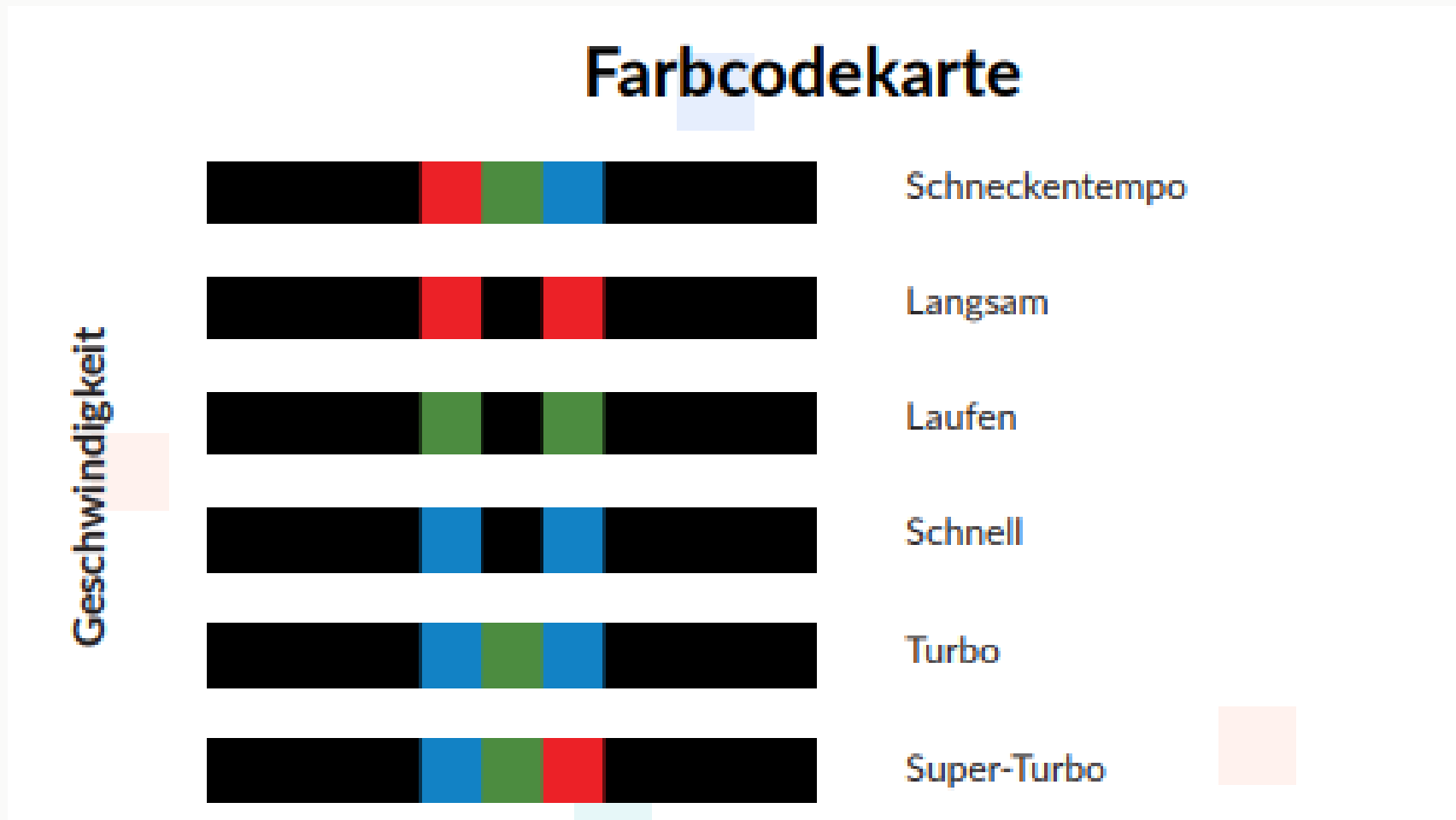
Der Ozobot erkennt **vier Farben**: 

Dadurch kann er **mit Hilfe von Farbcodes programmiert** werden.

Er liest die Farbcodes immer in Fahrtrichtung.



Offline Coding mit dem Ozobot



<https://ozobot-deutschland.de/lehrmaterial/>

Offline Coding mit dem Ozobot

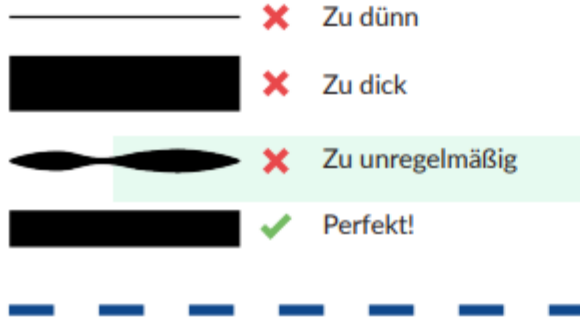
Wichtig

- Der Ozobot muss bei Inbetriebnahme und auch bei Veränderungen der Lichtverhältnisse kalibriert werden.
- Farbcodes werden am besten bei Tageslicht gelesen.
- Beim Zeichnen der Codes müssen bestimmte Vorgaben eingehalten werden.

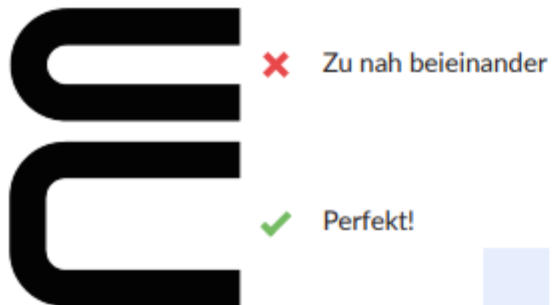
Offline Coding mit dem Ozobot

Tipps & Tricks: Linien & Codes zeichnen

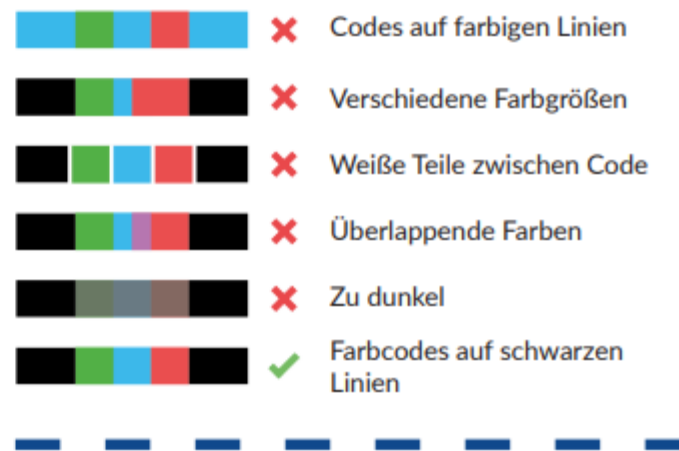
Linien zeichnen: Dicke



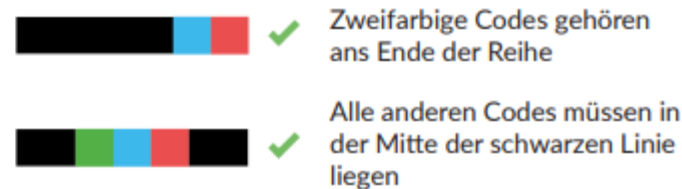
Kurven



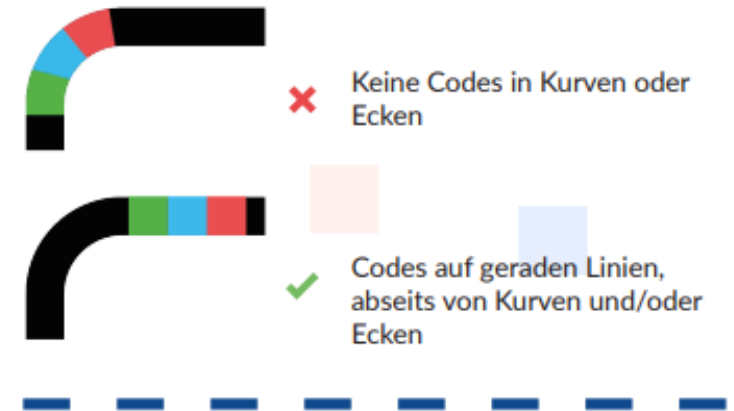
Linien zeichnen: Farbe



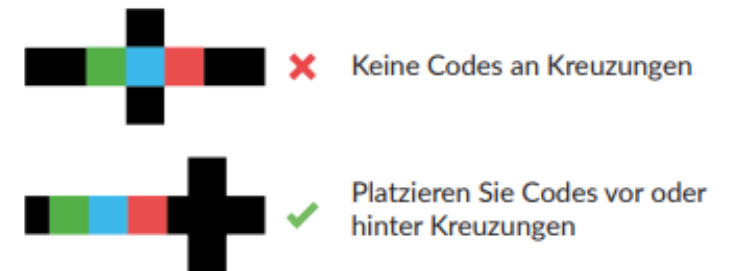
Position



Kurven und Farben



Kreuzungen



Coding in der Grundschule

leichter

schwieriger →

Ozobot Puzzlestücken



<https://images.robotworld.sk/texts/3900/3919/ozobot-drevene-puzzle-1.jpg>

Ozobot mit
vorgefertigten
Plänen



Ozobot mit Stift
und Papier
selbst zeichnen



<https://southcarolina.csteachers.org/Stories/ozobot-donation-program>

Projektideen für Farbprogrammierung

- Weglängen messen, Geschwindigkeit bestimmen
- Würfeln (Ein Start, 6 Ziele, Farbcodierung für zufällige Abzweigungen)
- Endlos-Parkour (beliebige Form) programmieren
- Irrgarten bauen
(bel. Wege, aber richtige Befehle zum Umdrehen zufälligen Abbiegen und Zielerkennung)
- Geschichte filmen (Landkarte malen, Gesten ausführen lassen)
- Heimweg darstellen (Ampel langsam, Bürgersteig schnell, am Kiosk halten)
- Befehlskarten bauen

Coding in der Grundschule

Links

<https://nawitas.uni-koeln.de/unterricht/bs-1-2/offline-coding-klasse-1-2>

<https://ozobot-deutschland.de/lehmaterial/> (geht nicht mehr!!!!)

<https://ozobot-deutschland.de/unterrichtsmaterial/>

[Lernroboter | Unterrichtsmaterial \(unihttps://www.robot-advance.com/EN/art-ozobot-bit-unit-2129.htm-muenster.de\)](https://www.robot-advance.com/EN/art-ozobot-bit-unit-2129.htm-muenster.de)

„Freies Programmieren“ mit OzoBlockly.com/editor

The screenshot shows the OzoBlockly.com editor interface. On the left, there is a sidebar with the OzoBlockly logo, a level selector (Levels 1-5), and a category menu with options: Movement, Light Effects, Timing, Loops, and Sounds. The main workspace displays a program titled "Lauf bis schwarz" with the following code blocks:

```
set top light color [white]
repeat forever
do
  move forward distance 1 step speed medium
  if (get surface color = surface color)
  do
    break out of loop
set top light color [red]
```

On the right side of the workspace, there are icons for "Programs" and "Menu".

- Befehle als Blöcke
- 5 Schwierigkeitslevel
- Befehle nehmen mit jedem Level zu
- Übertragung per Lichtsequenz
- Teilen der Programme per temp. Link oder lokal

Quelle: OzoBlockly.com/editor

OzoTown: Online Aufgaben zum Üben

Line Navigation

- follow line to next intersection or line end
- pick direction: left
- set line-following speed to medium

Goal

Program Ozobot to visit each site in number order and then travel home. For sites with a stop sign, stop and wait. Remember: Obey speed signs by coding the speed before the bot reaches the sign.

Run

Load Ozobot

- Kleine Aufgaben (Challenges) zum Verstehen der Logik
- Menüs „wachsen mit“
- Test/Demo direkt auf der Webseite (rechts)
- Überspielen des Programms auf den Ozobot via Licht (Flashing) ist nicht stabil!
- Leider nur auf Englisch

Quelle:

<https://games.ozoblockly.com/ozotown-basic>

Ozobot Blockly: Ipad & Web Aufgaben zum Üben

The screenshot displays the Ozobot Blockly interface. On the left, a sidebar contains a close button (X), an 'Evo' button, a 'Levels' section with '2' selected, and several navigation icons. The main workspace shows a Blockly script titled 'Lauf bis schwarz' with blocks for 'set to', 'repeat forever', 'do', and 'do'. A menu is open over the script, listing: Programs, Help, Glossary, Examples, Challenges, Code Preview, and Settings. The 'Challenges' panel on the right features three challenge cards:

- Rectangle Walk**
Level: 2 or higher
Program Ozobot to walk in a rectangle. The challenge is to make Ozobot move and its LED light up at certain points during the walk.
[Show Instructions]
- Slot Car Race Track**
Level: 3 or higher
Author: Prof. Richard Born
Slot car racing has been of interest to many for years and years. In this challenge you will be creating an ozoBlockly program that makes Ozobot Bit behave like a slot car. Create your program so that Ozobot will run five laps using the provided track.
[Show Instructions] [Download Map]
- Color Visit Counts**

At the bottom of the challenges panel, there is a link: 'Ready to code a real robot? [Shop Ozobots](#)'.

- Evo App
- Bereich Meta
- Virtuellen Parkour bauen

Aufgabe 1: 1 Schritt vor, 2 zurück (Level 1)

← ↻ 🏠 <https://ozoblockly.com/editor?lang=en&robot=evo&mode=1> 🔊

Movement

Bit+ Evo
MetaBot

Levels
1

→

💡

🕒

🔊

1 2 5 10

1 2 5 10

1 2 5

1 2

1 2

Aufgabe 1: 1 Schritt vor, 2 zurück (Level 2)

← ↻ 🏠 <https://ozoblockly.com/editor?lang=en&robot=evo&mode=2> 🔊

Movement

- move forward distance 1 step speed medium
- ↻ rotate slight left
- ⚡ zigzag medium
- 🛹 skate medium forward
- 🌀 spin left
- 🕒 small circle medium forward left 1 second
- 🕒 big circle medium forward left 1 second

Bit+ Evo MetaBot Levels 2

→

💡

🕒

🔄

🔊

Lev

→ move forward distance 1 step speed medium

→ move backward distance 1 step speed medium

Aufgabe: TopLicht als Blaulicht (Level 1 - klappt nicht richtig)

← ↻ 🏠 <https://ozoblockly.com/editor?lang=en&robot=evo&mode=1> 🔊

Movement

Bit+ Evo
MetaBot

Levels
1

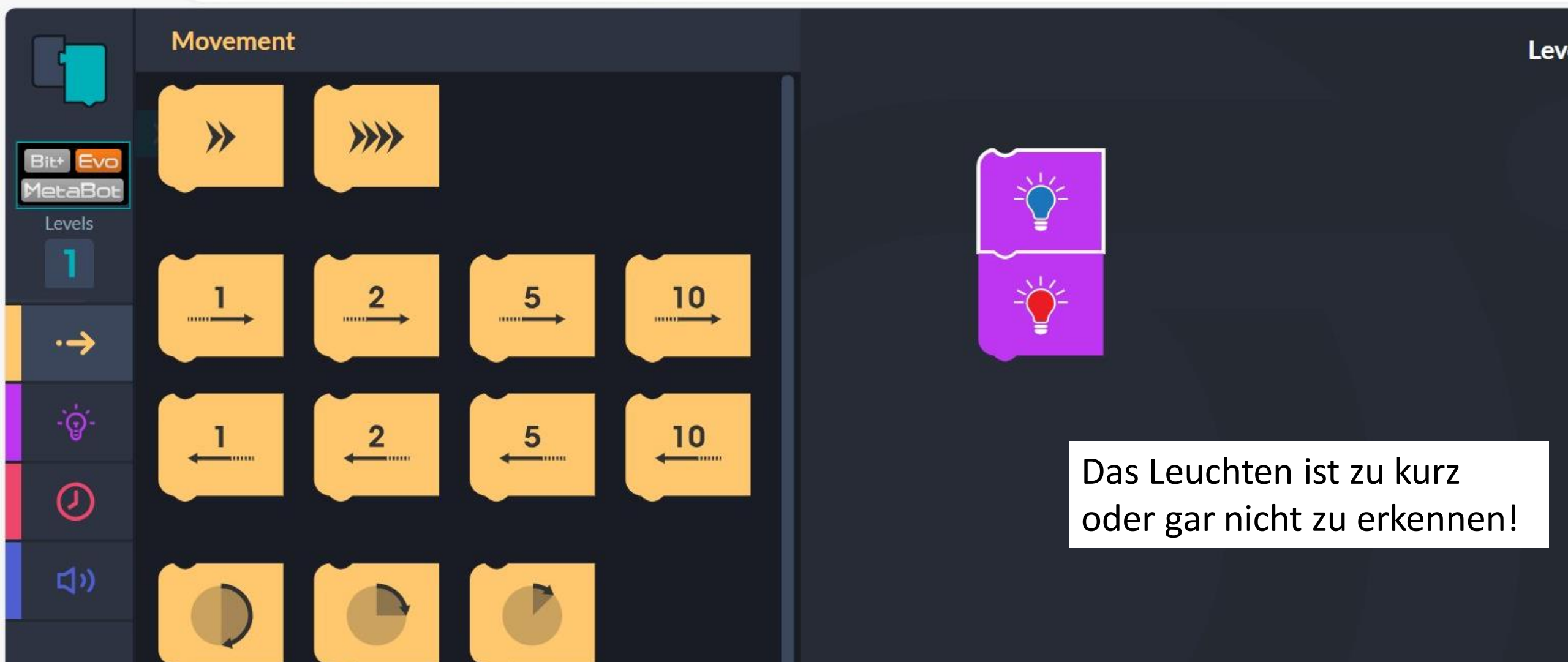
→

💡

🕒

🔊

Lev



The screenshot shows the Ozoblockly editor interface. On the left, there is a sidebar with a 'Movement' category selected. Under 'Movement', there are several blocks: two blocks with double and triple right-pointing arrows, a row of four blocks with right-pointing arrows labeled '1', '2', '5', and '10', a row of four blocks with left-pointing arrows labeled '1', '2', '5', and '10', and three blocks with circular arrows indicating rotation. On the right side of the workspace, there are two light blocks stacked vertically, one with a blue lightbulb icon and one with a red lightbulb icon. A white text box at the bottom right contains the text: 'Das Leuchten ist zu kurz oder gar nicht zu erkennen!'.

Das Leuchten ist zu kurz
oder gar nicht zu erkennen!

Aufgabe: TopLicht als Blaulicht (Level 1 - besser)

← ↻ 🏠 <https://ozoblockly.com/editor?lang=en&robot=evo&mode=1> 🔒

Light Effects

Bit+ Evo
MetaBot

Levels
1

→

Light Effects palette:

- Row 1: Red, Green, Blue lightbulbs
- Row 2: Yellow, Cyan, Magenta lightbulbs
- Row 3: Traffic light, Christmas tree
- Row 4: Rainbow, Fireworks, Moon

Script stack:

- Blue lightbulb
- Timer 1 (red)
- Red lightbulb
- Timer 1 (red)

Lev

Es blinkt nur ein Mal...

Aufgabe: TopLicht als Blaulicht (Level 2 - richtig mit Schleife)

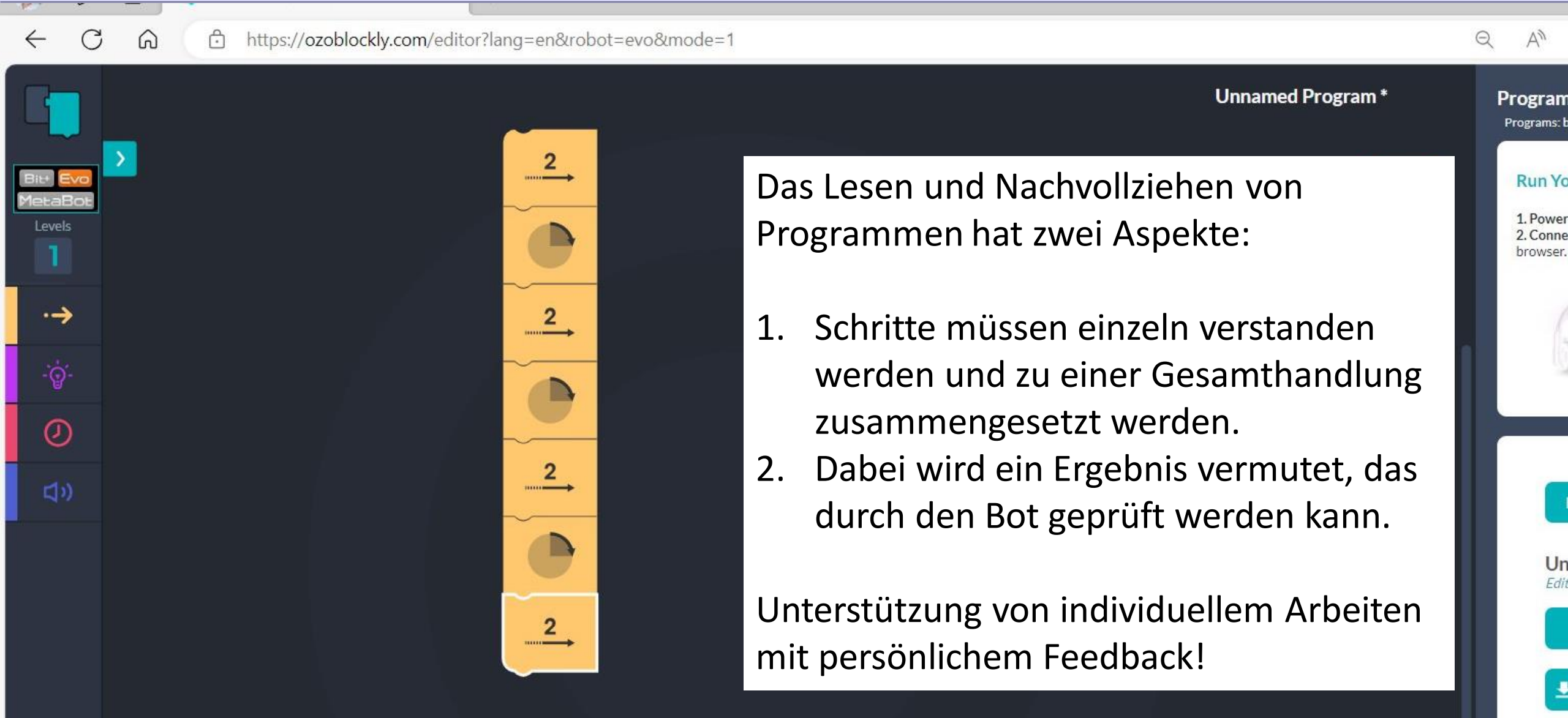
The screenshot shows the Ozoblockly web editor interface. At the top, the browser address bar displays the URL: <https://ozoblockly.com/editor?lang=en&robot=evo&mode=2>. The main workspace is dark-themed and contains a script for Level 2. On the left, the 'OZO Blockly' logo is visible, along with robot selection buttons for 'Bit+', 'Evo', and 'MetaBot'. Below these are level selection buttons numbered 1 through 5, with '2' highlighted. A vertical sidebar on the left lists categories: 'Movement', 'Light Effects', 'Timing', 'Loops', and 'Sounds'. The 'Loops' category is currently selected. The script in the workspace consists of a 'repeat forever' loop block. Inside this loop, there is a 'do' block containing four blocks: 'turn top light off', 'set top light color' (set to red), 'wait 1 second(s)', 'set top light color' (set to blue), and another 'wait 1 second(s)'. The script concludes with another 'turn top light off' block.

Aufgabe: TopLicht als Blaulicht (Level 5 - toll, mit Abschalten)

The screenshot shows the Ozoblockly editor interface. The browser address bar displays <https://ozoblockly.com/editor?lang=en&robot=evo&mode=5>. On the left sidebar, the robot is identified as 'MetaBot' and the current level is '5'. The main workspace contains the following script:

- when green flag clicked** (implied start block)
- if object in front** (condition)
- capture button press events** (set to `true`)
- set button press count** (set to `0`)
- repeat forever** loop:
 - do** (lightbulb icon): **set top light color** (set to red)
 - wait** (clock icon): `1` second(s)
 - turn top light off** (lightbulb with slash icon)
 - do** (lightbulb icon): **set top light color** (set to blue)
 - wait** (clock icon): `1` second(s)
 - turn top light off** (lightbulb with slash icon)
- if** (condition): **get button press count** (set to `>`) `0`
- do** (lightbulb icon): **terminate program and** (set to `switch to idle`)

Code verstehen: Was passiert und wie steht Ozobot am Ende?



The screenshot shows the Ozoblockly web editor interface. The browser address bar displays the URL: <https://ozoblockly.com/editor?lang=en&robot=evo&mode=1>. The editor title is "Unnamed Program *". On the left sidebar, there are icons for "Bit+ Evo MetaBot", "Levels 1", and various tool icons. The main workspace contains a vertical stack of seven code blocks: a "2" block with a right-pointing arrow, a "90-degree turn" block, another "2" block with a right-pointing arrow, another "90-degree turn" block, another "2" block with a right-pointing arrow, another "90-degree turn" block, and a final "2" block with a right-pointing arrow. On the right side, there is a "Programs" panel with a "Run Your Program" button and a list of steps: "1. Power on the robot" and "2. Connect to the browser".

Das Lesen und Nachvollziehen von Programmen hat zwei Aspekte:

1. Schritte müssen einzeln verstanden werden und zu einer Gesamthandlung zusammengesetzt werden.
2. Dabei wird ein Ergebnis vermutet, das durch den Bot geprüft werden kann.

Unterstützung von individuellem Arbeiten mit persönlichem Feedback!

Mögliche Aufgaben mit OzoBlockly (Level 1, 2)

Browser Edge: URL <https://ozoblockly.com/editor>

1. Lass das Toplicht abwechselnd rot und blau blinken
Tipp: Nutze den Timer, um das Licht lange genug leuchten zu lassen
2. Fahre einen Schritt vor, und zwei zurück
Tipp: Nutze den Code aus Aufgabe 1, um in Programm 2 anzuzeigen, wenn das Programm zuende ist
3. Fahre in einem großen Rechteck
Frage: Was musst Du tun, um das Rechteck zu vergrößern?

Mögliche Aufgaben mit OzoBlockly (Level 3-5)

4. Gefängnis: Male einen großen schwarzen Kreis und lasse Ozobot solange in eine Richtung fahren, bis die Grenze erreicht ist.
Wie dick muss man die Grenze malen, damit Ozobot sie erkennt.
Tipp: Mache Dir vorher Notizen, welche Befehle Du für diese Aufgabe nutzen möchtest und bringe Sie in die richtige Reihenfolge.
5. Erweitere Aufgabe 4 so, dass das Toplight des Ozobots immer die Farbe des Untergrundes anzeigt.
6. Mars Expedition: Male einen großen schwarzen Kreis und einige rote und blaue Felder am inneren Rand des Kreises. Der Ozobot soll nun erkunden, welche Farbe der Untergrund am Rande der Fläche hat. Dafür soll er einfach in eine Richtung losfahren, bis zum Rand, erkennen welche Untergrundfarbe dort herrscht, in dieser Farbe die Lampe anschalten und zurückkommen.
Erweiterung: Wie weit weg ist der Rand eigentlich?