

# Einstieg in Open Roberta

Mit LEGO Robotern zur World Robot Olympiad



# Agenda

- Vorstellung
- Motivation
- World Robot Olympiad
- Open Roberta Lab
  - Vorstellung der Plattform
  - Arbeiten in Kleingruppen



# Vorstellung



# Referenten / Kontaktdaten



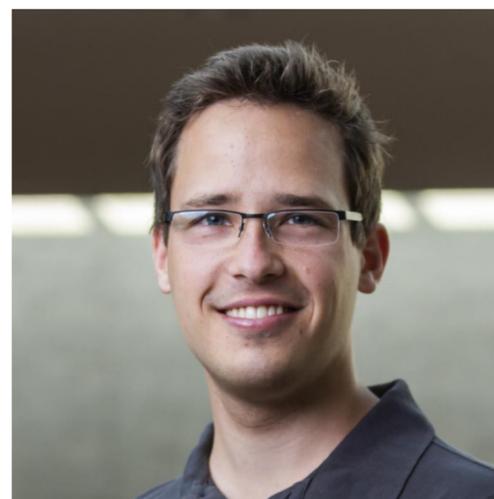
**Lukas Plümper**



**Egbert Langer**



**Jenny Kociemba**



**Tobias Schepers**

**Rückfragen gerne an**

Lukas Plümper

[lp@technik-begeistert.org](mailto:lp@technik-begeistert.org)

0176 / 98 331 332

Weitere Informationen unter:

<https://wro2021.de/mint-ec>



# Unser Verein TECHNIK BEGEISTERT e.V.

- Ziel: Förderung von MINT-Bildung  
Schwerpunkt Informatik und Technik
- Engagement durch Ehrenamtliche, Partner und Vereinsmitglieder
- Koordination der World Robot Olympiad
- Durchführung von Workshops im Bereich LEGO-Robotik

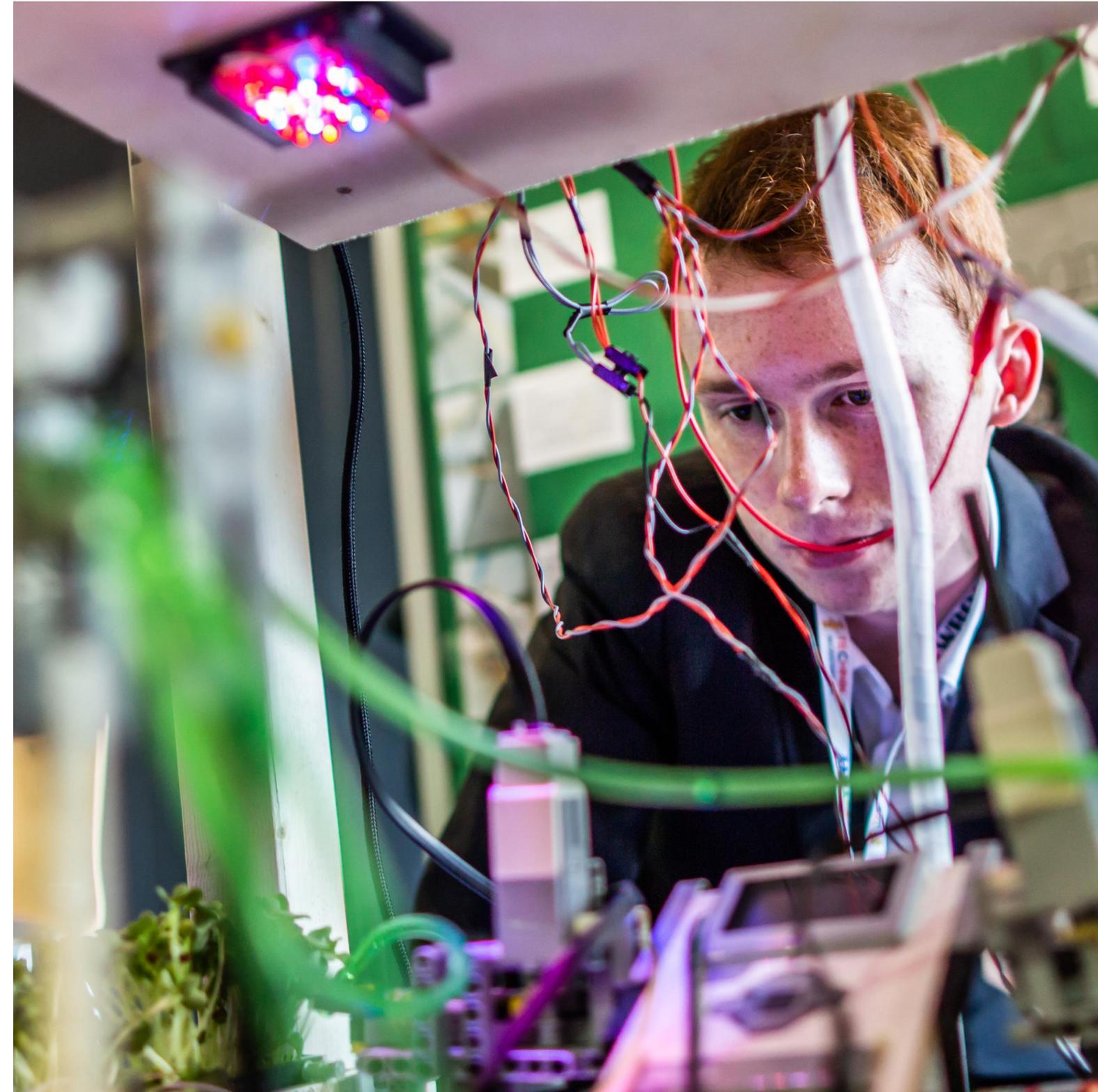


# Motivation



# Warum Roboterwettbewerbe?

- Neugier der Teilnehmer wecken
- Spielerisches Lernen
- Förderung von Softskills
- Spaß beim Wettbewerb
  
- Video:  
<https://www.youtube.com/watch?v=Yvg3W1dfNmo>



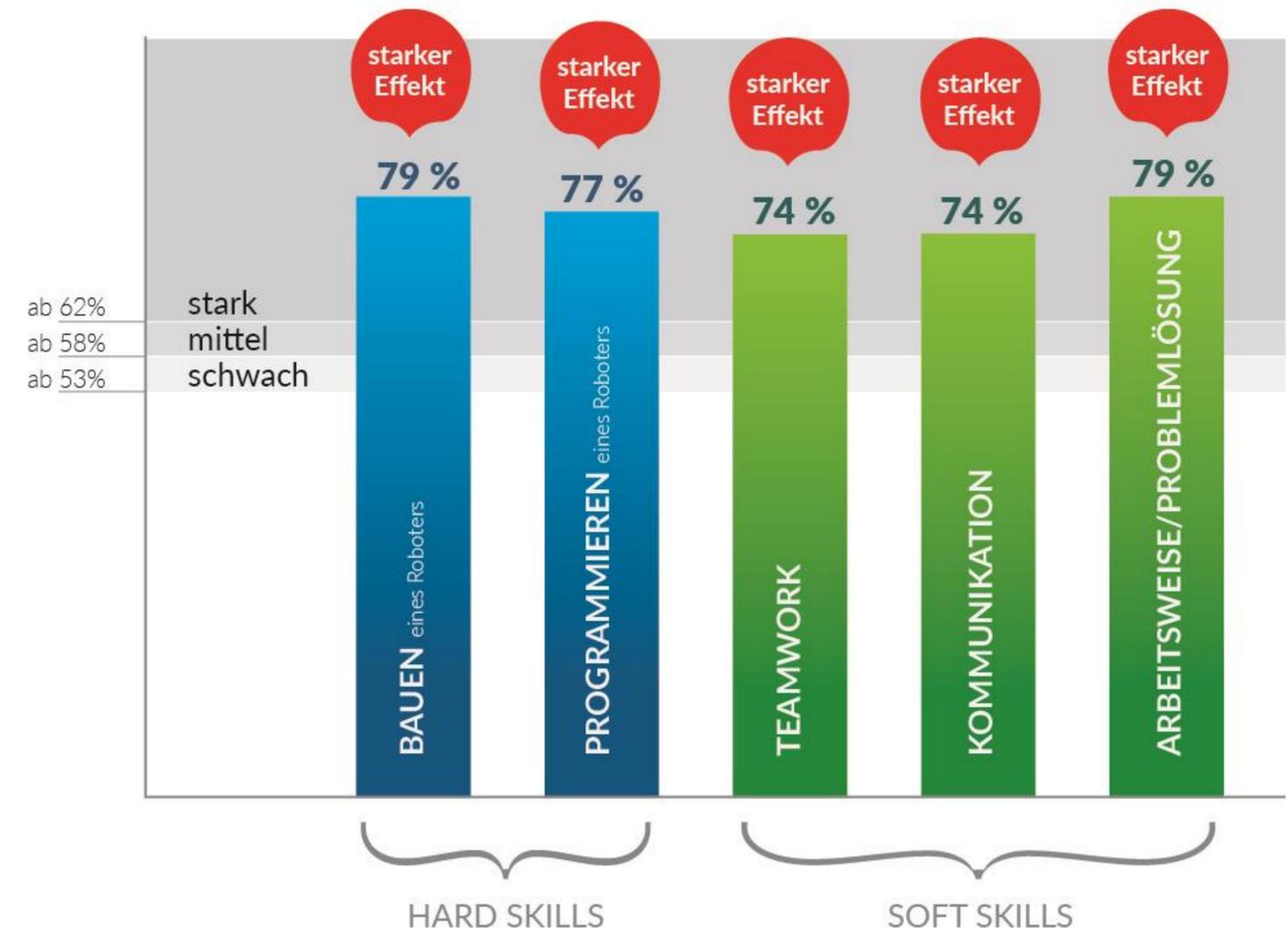
# Evaluationsstudie 2019 bestätigt: WRO wirkt!

- Kinder und Jugendliche entwickeln bei WRO-Wettbewerbe sowohl Hard- als auch Soft-Skills
- Kompetenzen entwickeln sich unabhängig von Altersklasse, Geschlecht, Erfolg & Erfahrung
- Teilnahme fördert das Selbstkonzept und Interesse an Informatik + Technik
  - Indikatoren für Studien- und Berufswahl

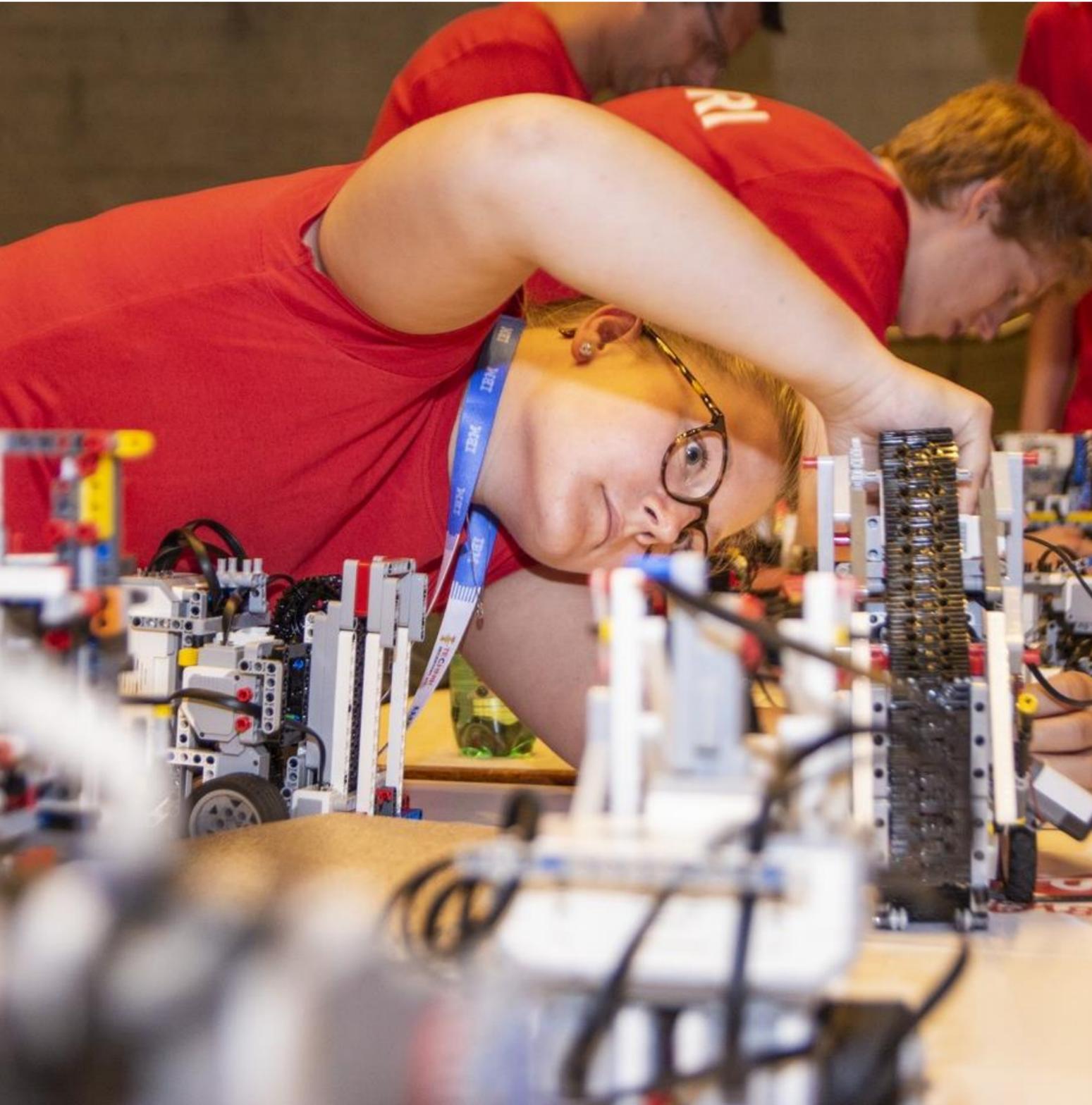


Darstellung der Einflussstärke der WRO auf die Kompetenzentwicklung der Teilnehmerinnen und Teilnehmer

in CLES (Common Language Effect Size)



**CLES:** Common Language Effekt Size: Maßeinheit zur Angabe der Effektstärke eines Einflusses. CLES gibt die Wahrscheinlichkeit an, mit der ein zufällig aus der einen Gruppe gezogener Fall einen höheren Wert hat als ein zufällig gezogener Fall aus der anderen Gruppe (z.B. vorher – nachher, Vergleiche von Untergruppen etc.)



## Wettbewerbe sind wichtig für das Schulprofil

- Besonders für Roboter-AGs liefern Wettbewerbe immer wieder neue Aufgaben und damit neue Herausforderungen für die Kinder & Jugendlichen
- Teilnahme an Wettbewerben hilft den Schulen bei der Herausstellung des eigenen MINT-Profiles (z.B. Zertifizierung MINT freundliche Schule)

# WRO-Wettbewerb

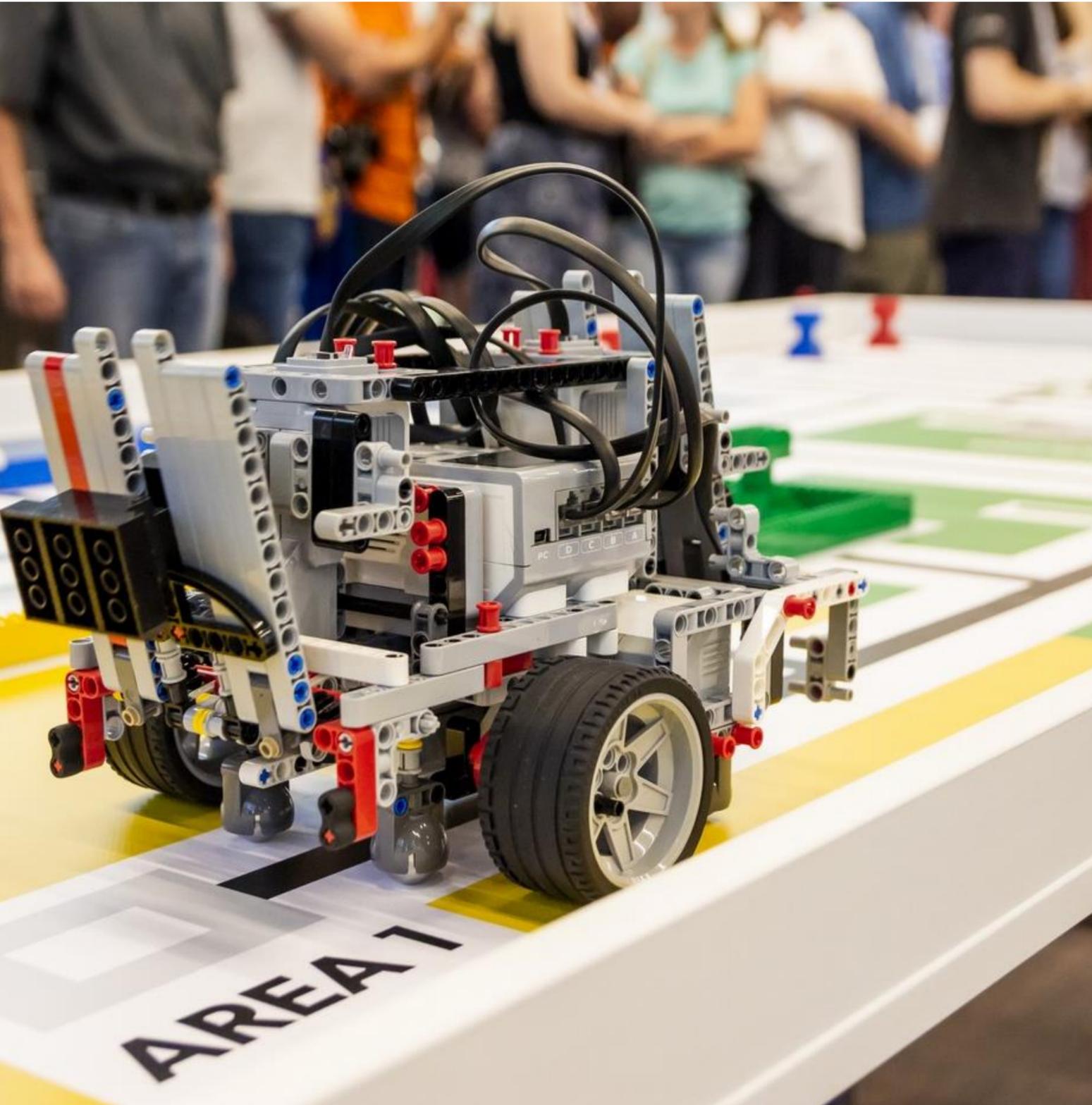
Struktur und Größe des Wettbewerbs



# Was ist die World Robot Olympiad (WRO)?

- Wettbewerb in über 80 Ländern
- Weltweit identische Aufgaben
- Jährlich wechselndes Thema
- Verschiedene Kategorien & Altersklassen
- Für Mädchen & Jungen von 6 – 19 Jahren
- 2er oder 3er-Teams + Team-Coach
- Ein Team nimmt in einer Kategorie / Altersklasse-Kombination teil
- **Kurzfilm:** [www.wro2020.de/film](http://www.wro2020.de/film)



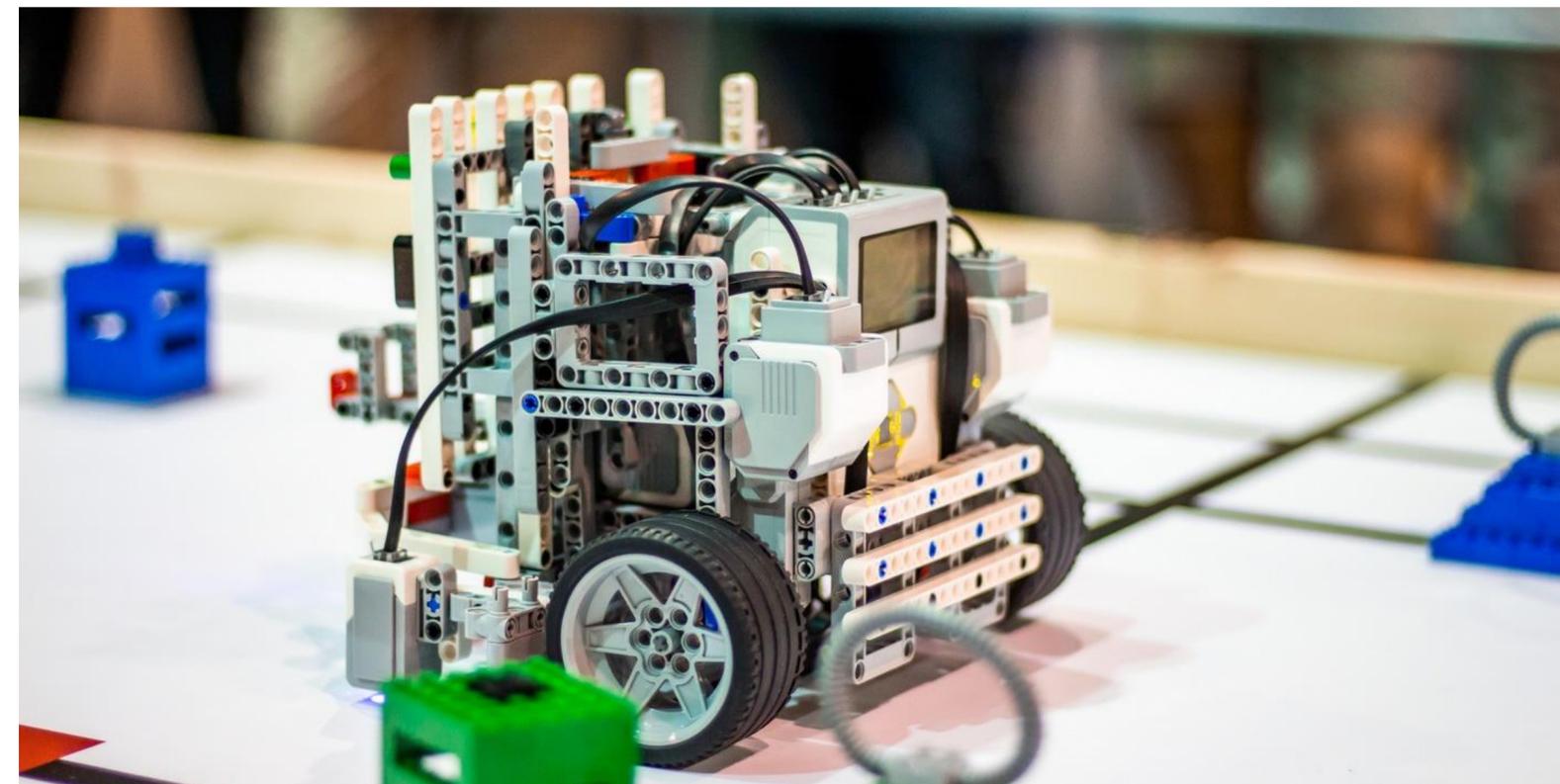


## WRO-Wettbewerb ist mit aktuellen Themen besetzt

- **Digitale Bildung:** Förderung der Kenntnisse im Bereich Informatik & Technik
- **21 century skills:** Teamwork, Problemlösefähigkeiten, Kreativität und Innovation sind bei den WRO-Wettbewerben gefragt
- **MINT-Nachwuchsförderung:** Wir bilden die Ingenieure und IT-Experten von Morgen aus

# Besonderheiten der WRO

- Teams konzentrieren sich auf Teilnahme in einer der 3 Wettbewerbskategorien
- **Starter-Programme** für WRO-Einsteiger
- Große Altersspanne (6 – 19 Jahre) mit altersgemäße Aufgaben und Bewertung
- Überall freie **Wahl der Programmiersprache**



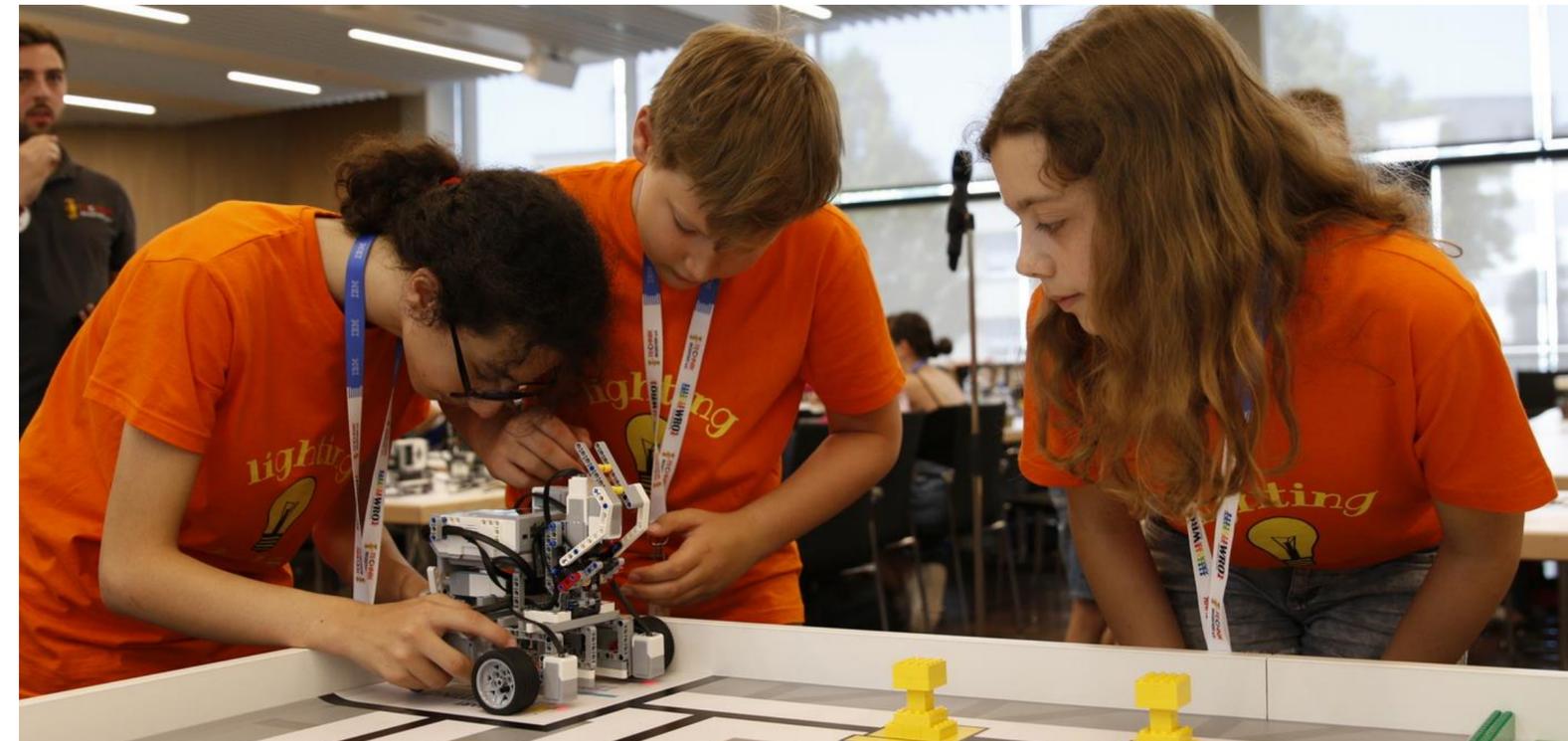
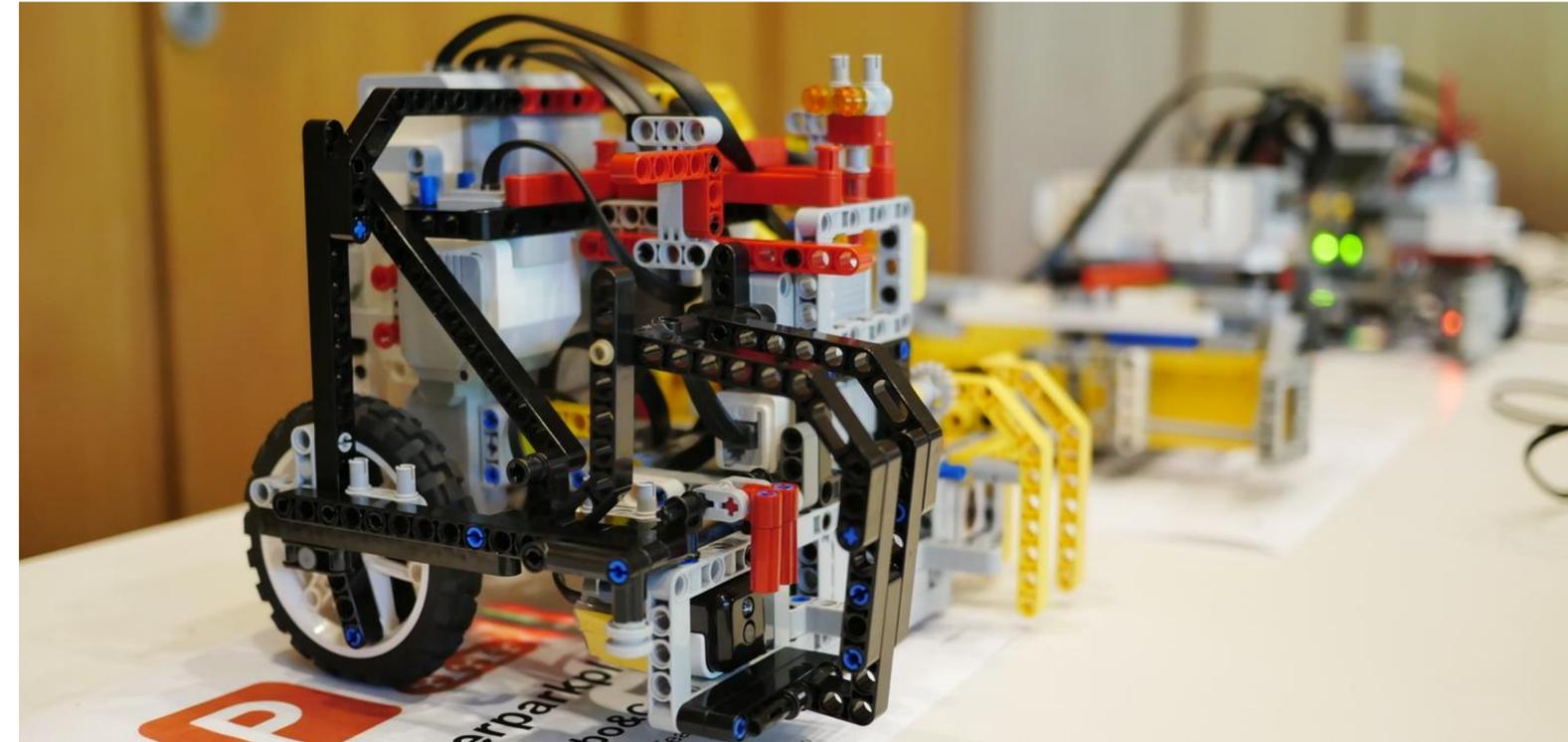
# Regular Category





# Regular Category

- Wettbewerb in **3 Altersklassen**
- Individuelle **Aufgabenparcours** pro Altersklasse
- Bau und Programmierung eines **LEGO-Roboters**
- **Maximale Robotergröße: 25cm x 25cm x 25cm**
- **Überraschungsaufgabe** am Wettbewerbstag
- Roboter muss beim Wettbewerb aus allen Einzelteilen **erneut zusammengebaut** werden
- Starter-Projekt für alle von 6-12



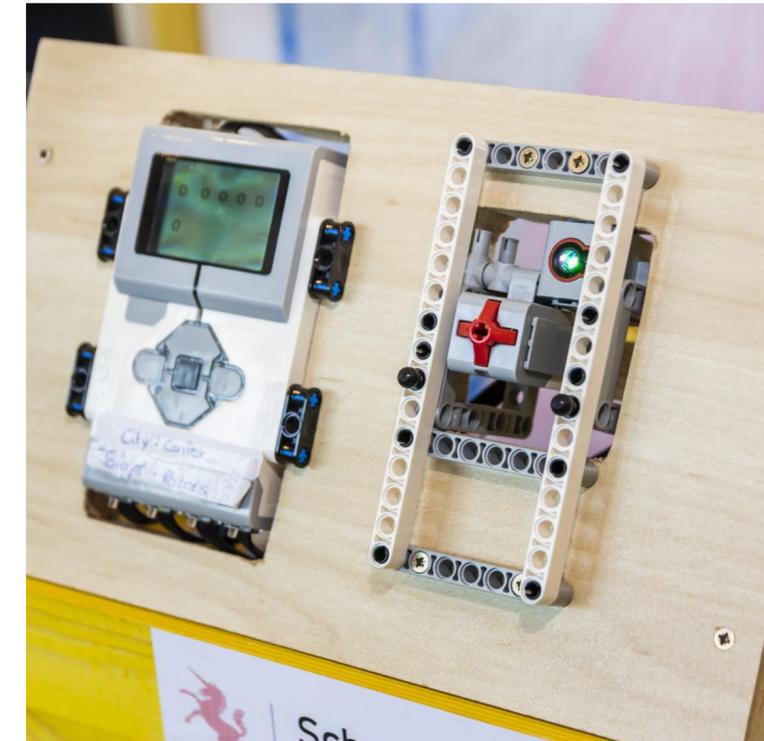
# Open Category





# Open Category

- Wettbewerb in **3 Altersklassen**
- Alle Altersklassen haben die **gleiche Aufgabe**
- Bau eines **Robotermodells** zum Thema der Saison
- Größe des Robotermodells bis zu **2m x 2m x 2m**
- **Freie Wahl** der Baumaterialien & Software
- Schriftliche **Dokumentation** des Projekts
- **Präsentation** vor einer Jury am Wettbewerbstag



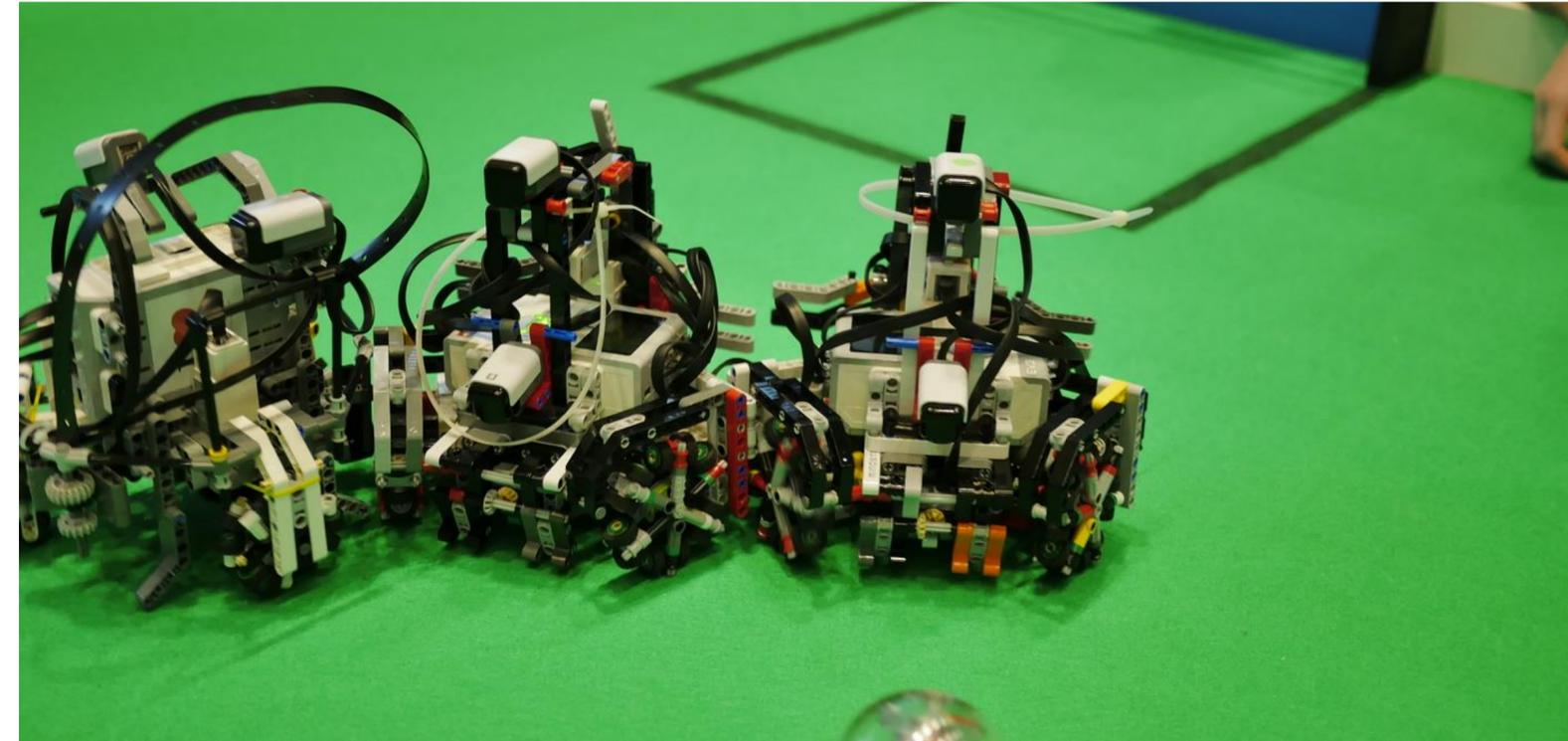
# Football Category



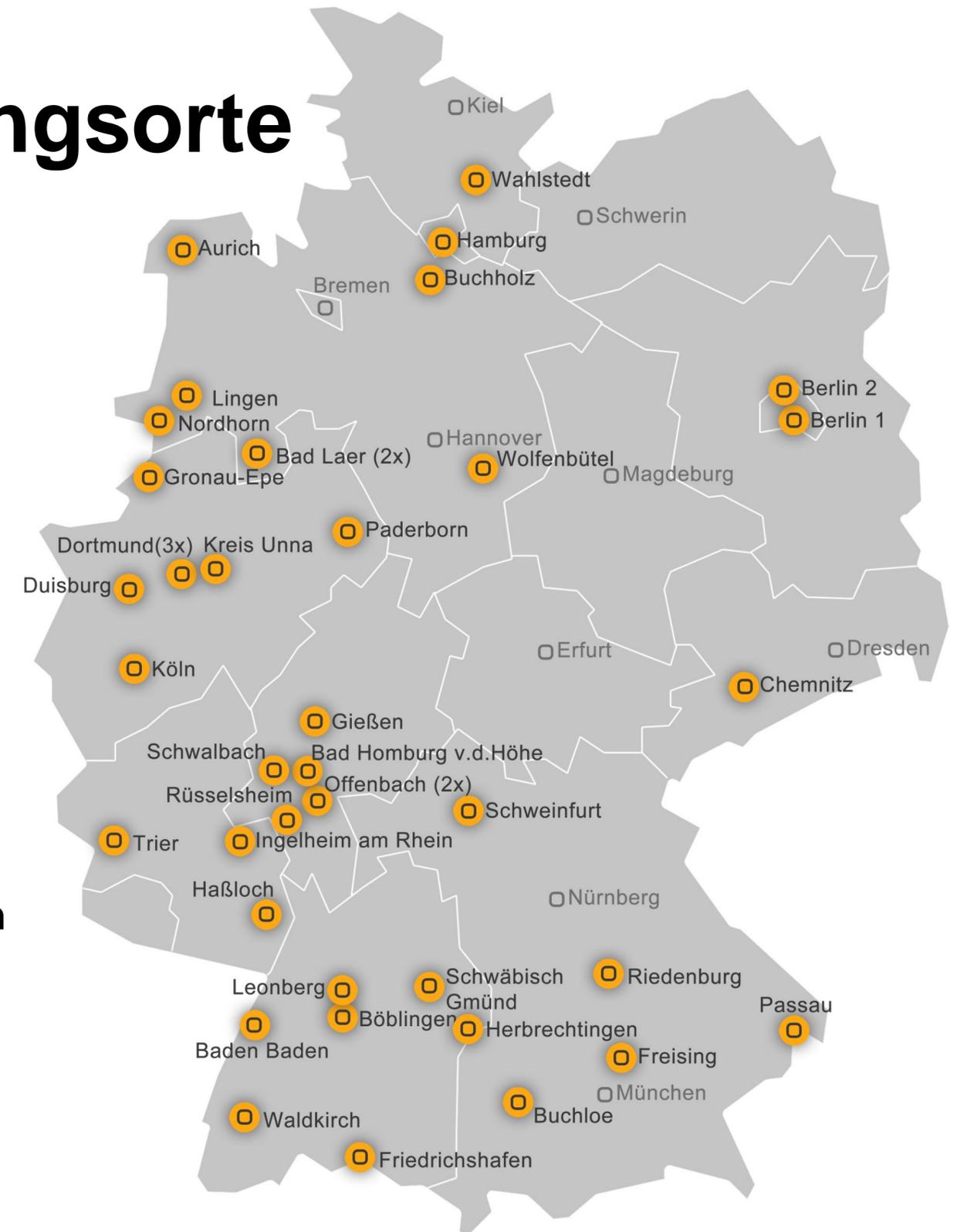


# Football Category

- Eine Altersklasse von **6 – 19 Jahren**
- Bau und Programmierung von **LEGO-Robotern**
- Fußballspielen mit **Infrarotspielball**  
**und Kompassensoren**
- Roboter müssen beim Wettbewerb aus Einzelteilen **erneut zusammengebaut** werden
- **Freie Wahl** der Programmiersprache
- Gespielt wird im **Liga-Modus** (wie Bundesliga)
- **Ein oder zwei Roboter** pro Team (Starter / Traditionell)



# Ablauf einer Saison / Austragungsorte



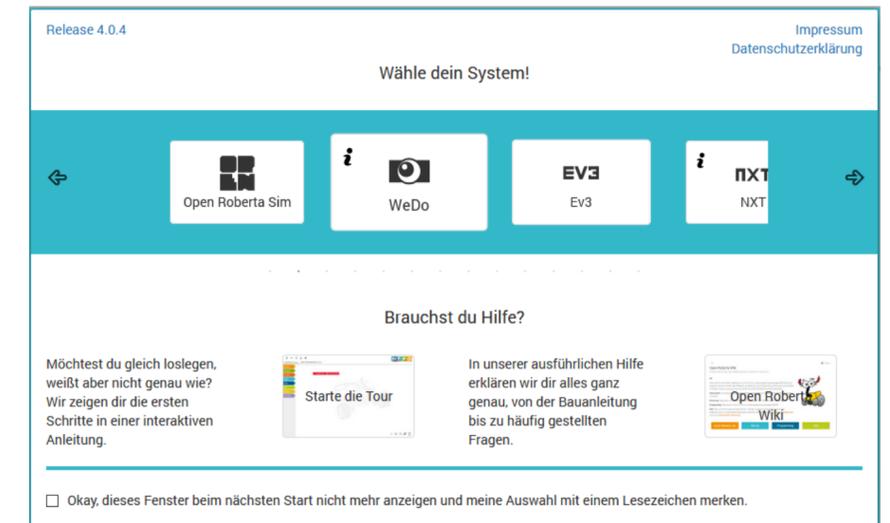
# Open Roberta Lab

Einblick in die Programmierung

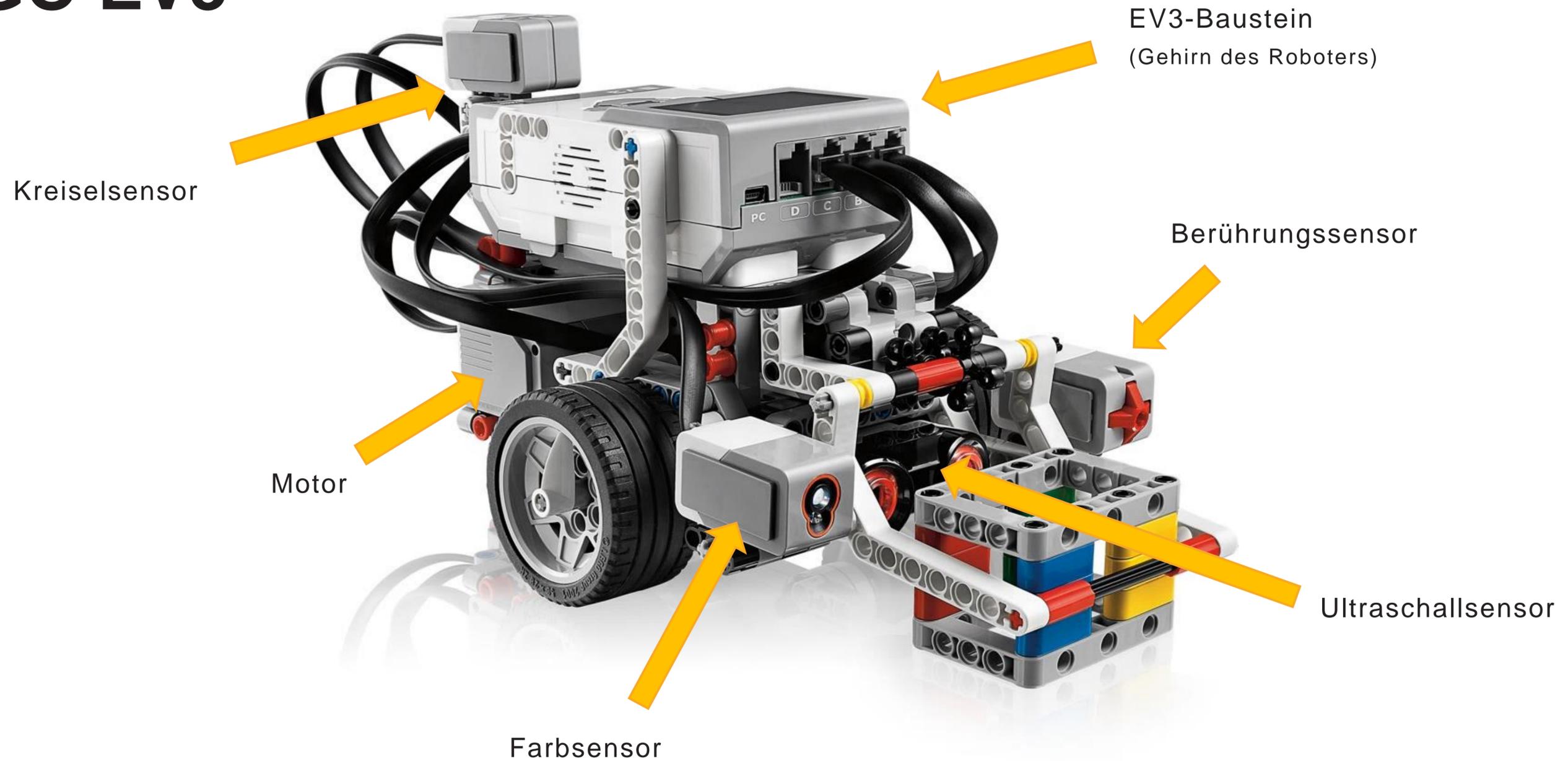


# Funktionen von Open Roberta Lab

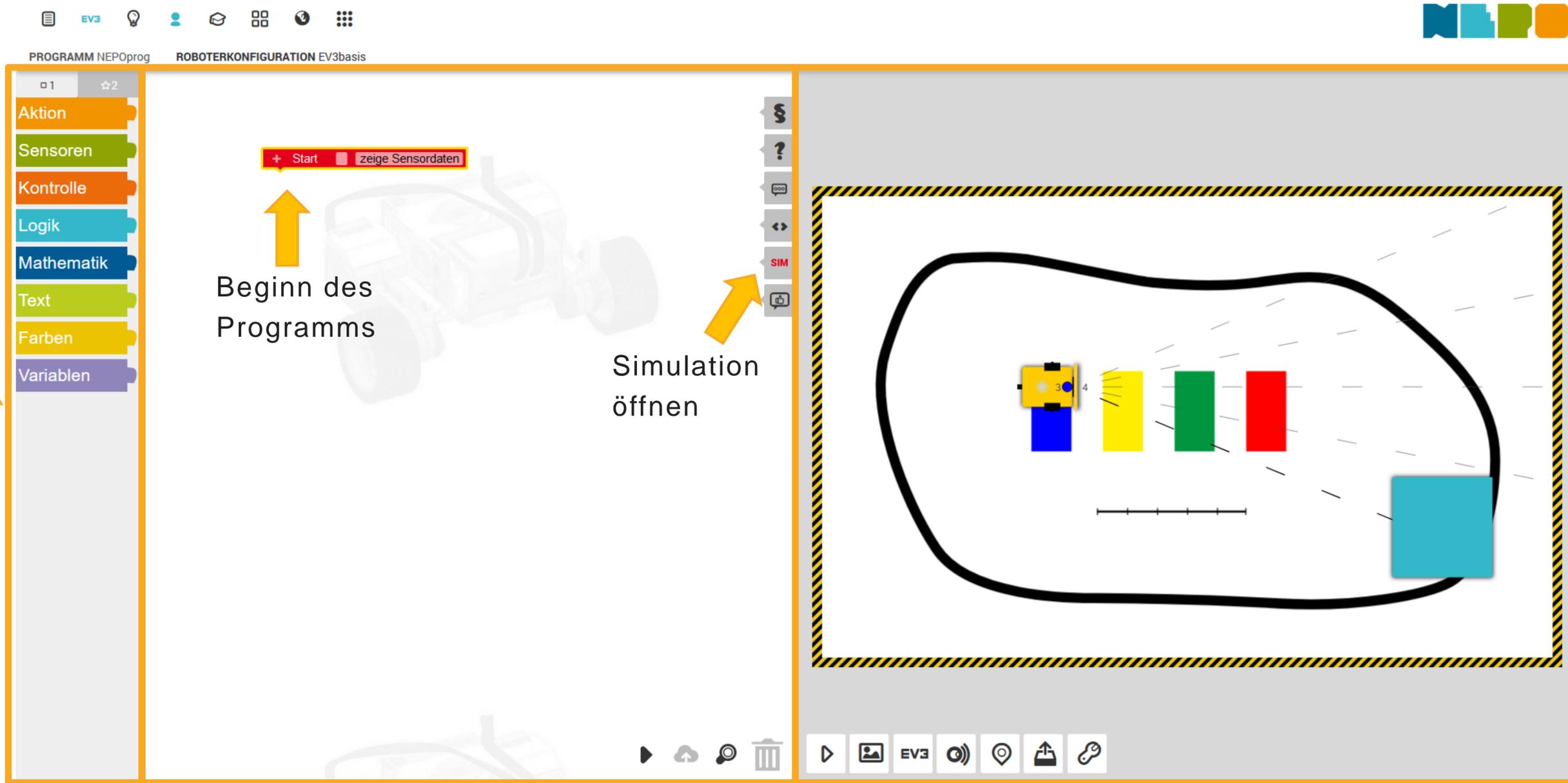
- Einfache, grafische Programmierung (NEPO)
- Unterstützt unterschiedlichste Roboter-Modelle
- Verwendung im Browser
  - Online-Simulation der Programme
  - Kein Roboter erforderlich
- Kostenfrei nutzbar
- Einsatz mit Schüler:innen möglich (Datenschutz)
- Gruppenfunktionalitäten für Klassen verwendbar



# Wie sieht so ein Roboter aus? – LEGO EV3



# Vorstellung der Umgebung



Programmsymbole

Programmbereich

Simulationsbereich

# Konfiguration der Sensoren & Motoren

Daten für die Verwendung von physikalischem Roboter

The screenshot shows the configuration interface for an EV3 robot. It includes the following settings:

- EV3**
- Raddurchmesser: 5.6 cm
- Spurbreite: 18 cm
- Sensor 1: Berührungssensor
- Sensor 2: Kreisel sensor
- Sensor 3: Farbsensor
- Sensor 4: Ultraschallsensor
- Motor A: (empty)
- Motor B: Großer Motor
  - Regulierung: ja
  - Drehrichtung: vorwärts
  - Seite: rechts
- Motor C: Großer Motor
  - Regulierung: ja
  - Drehrichtung: vorwärts
  - Seite: links
- Motor D: (empty)

Konfiguration der Sensoren (Änderungen für Simulation möglich)

Konfiguration der Motoren

# Symbole / Programmablauf

Detailgrad der Programmierbausteine



Kategorien der Programmierbausteine

Bewegungen und Aktionen

Sensorwerte abfragen

Kontrollstrukturen  
z.B. Schleifen oder Abzweigungen

# Simulationsbereich

The image shows a simulation environment for a robot. A yellow robot is positioned on a black track within a rectangular arena bounded by a yellow and black striped border. In the center of the arena, there are four vertical bars of different colors: blue, yellow, green, and red. To the right of these bars is a light blue square obstacle. A data panel in the top right corner displays the following information:

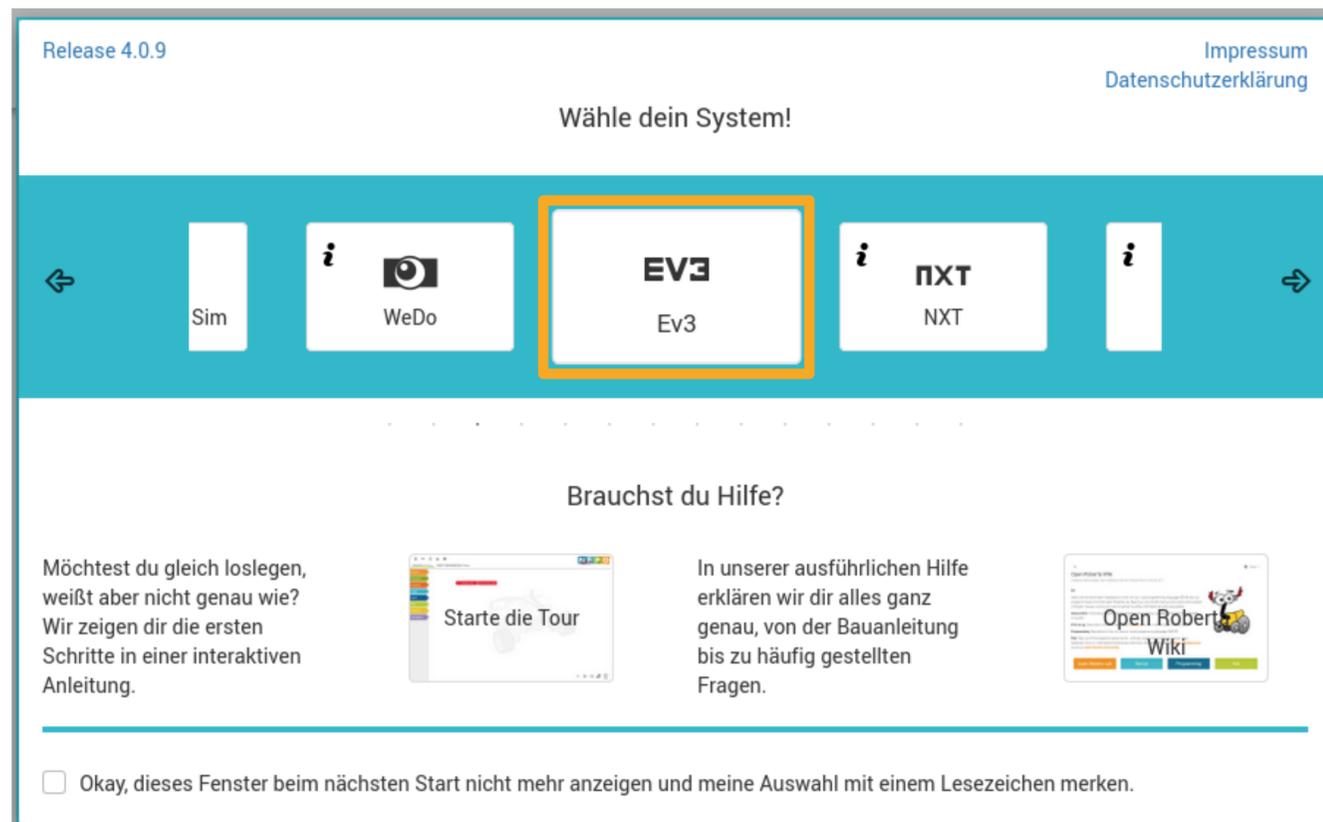
Robot	[Dropdown]
FPS	53
Time	0s
Robot X	177
Robot Y	146
Robot θ	0°
Motor left	0°
Motor right	0°
Touch Sensor 1	0
Light Sensor 3	100%
Ultra Sensor 4	112cm
Color Sensor 3	

At the bottom of the simulation window is a control bar with several icons: a play button, a camera icon, a button labeled 'EV3', a speaker icon, a location pin icon, a refresh icon, and a key icon. Yellow arrows point from text labels to these elements:

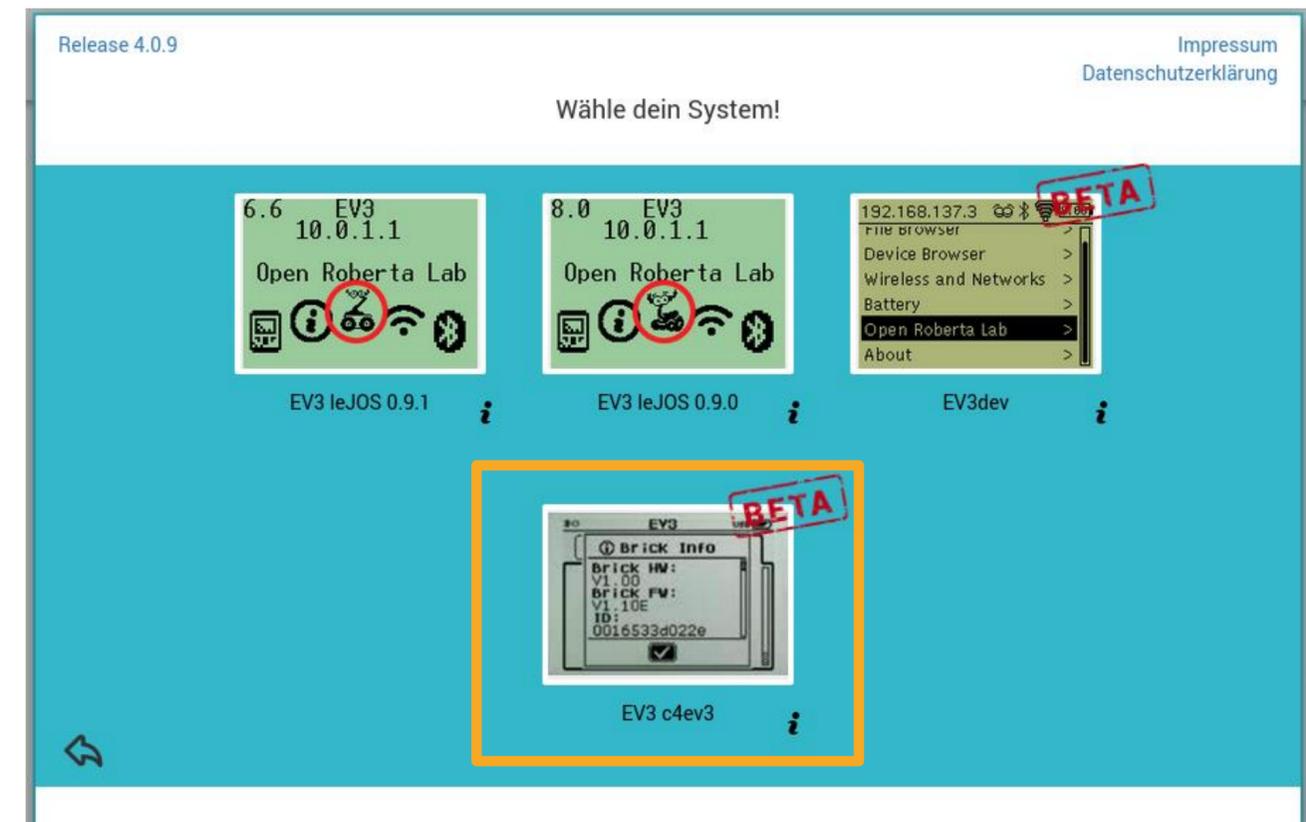
- Roboter**: Points to the yellow robot.
- Sensordaten anzeigen**: Points to the 'EV3' button.
- Start der Simulation Ausführen des Programms**: Points to the play button.
- Simulation zurücksetzen**: Points to the refresh icon.
- Sensordaten des Roboters**: Points to the data panel.
- Hindernis (unüberwindbar)**: Points to the light blue square obstacle.
- Linien / farbige Elemente (überwindbar)**: Points to the colored bars.

# Zur Software

- Link: <https://lab.open-roberta.org/>



Erster Schritt: „EV3“



Erster Schritt: „EV3 c4ev3“

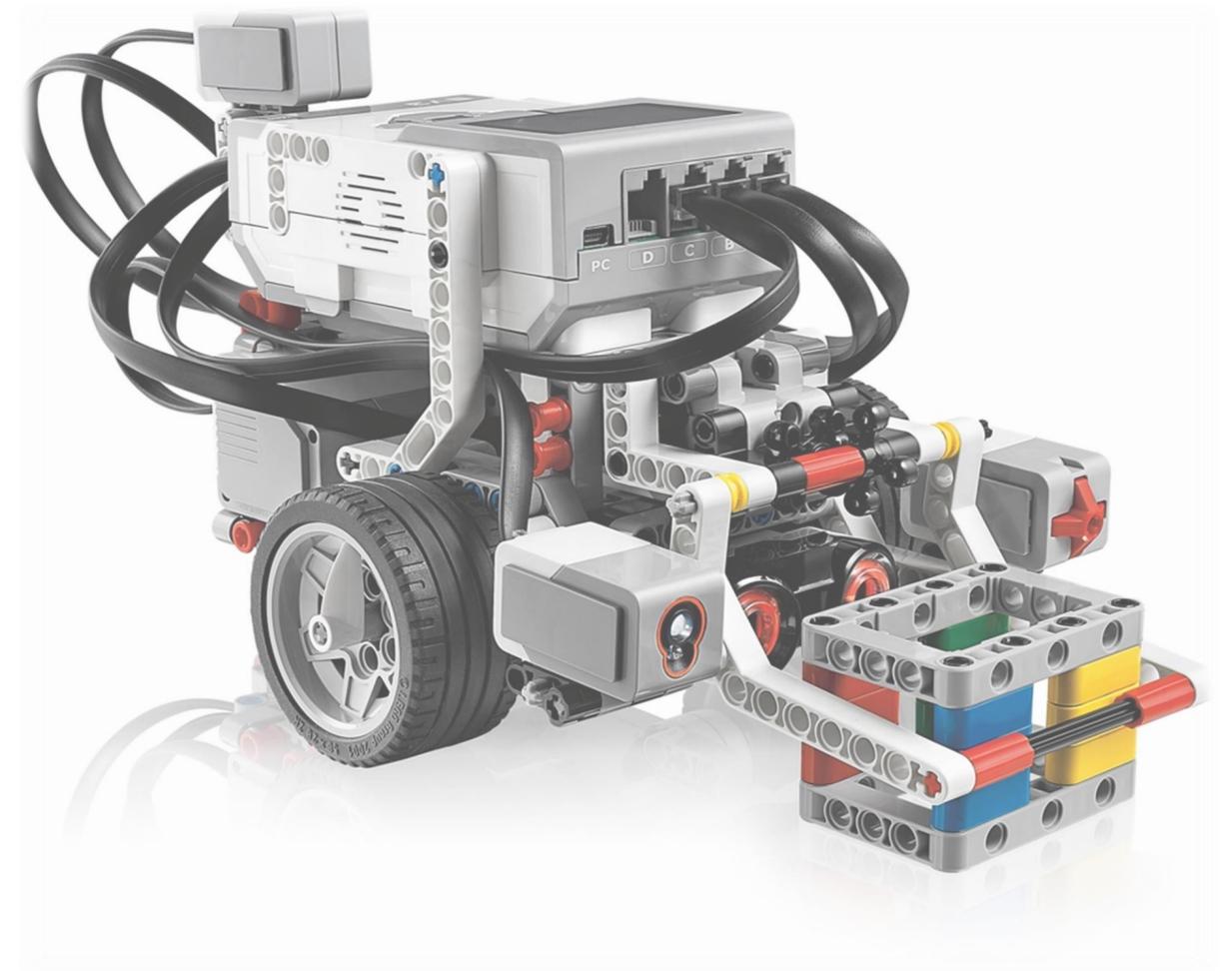
# Aufgaben zum Fahren



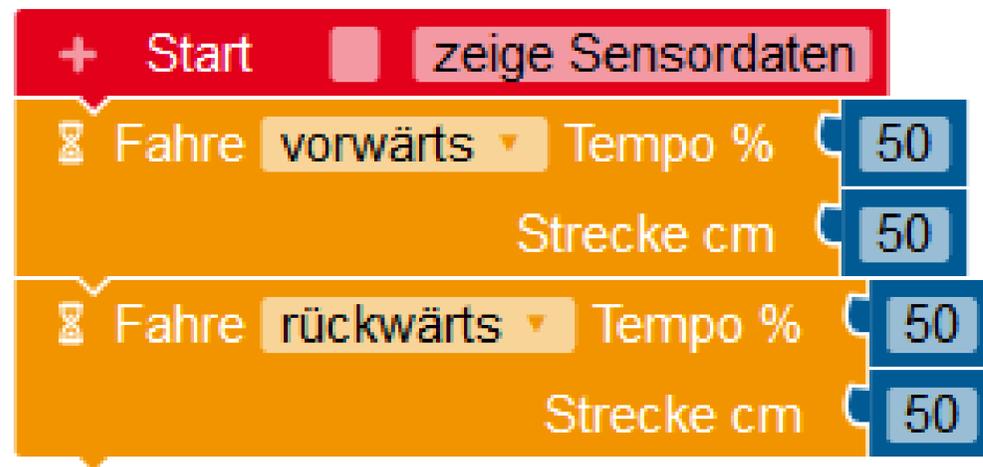
# Programmierung – Aufgabe 1

## Vorwärts und rückwärts fahren

- Fahre 50cm vorwärts
- Fahre 50cm rückwärts



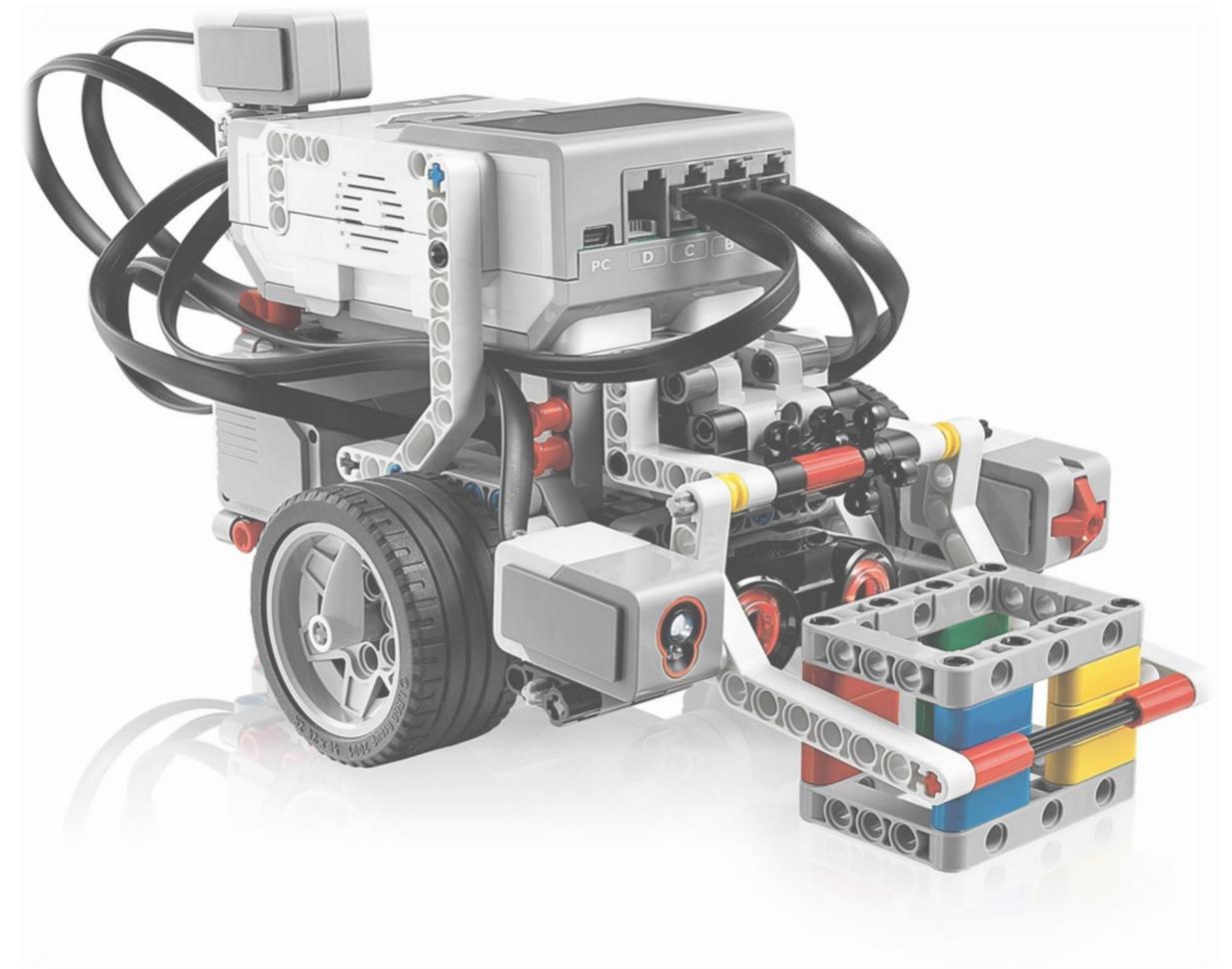
# Programmierung – Aufgabe 1



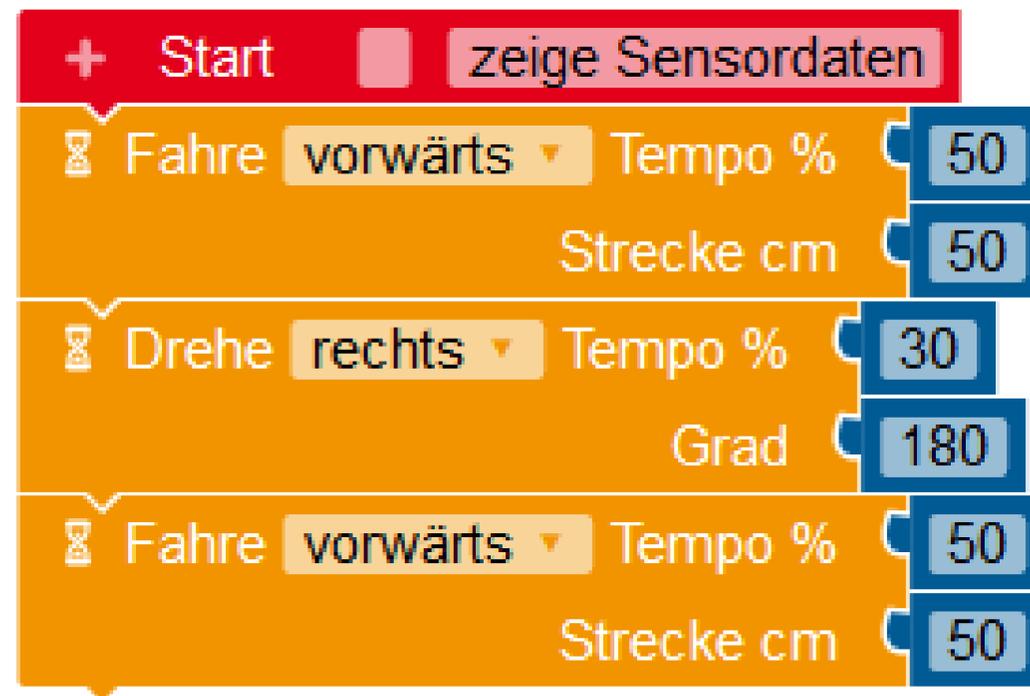
# Programmierung – Aufgabe 2

## Die erste Drehung auf der Stelle

- Fahre 50cm vorwärts
- Drehe auf der Stelle um 180°
- Fahre 50cm vorwärts



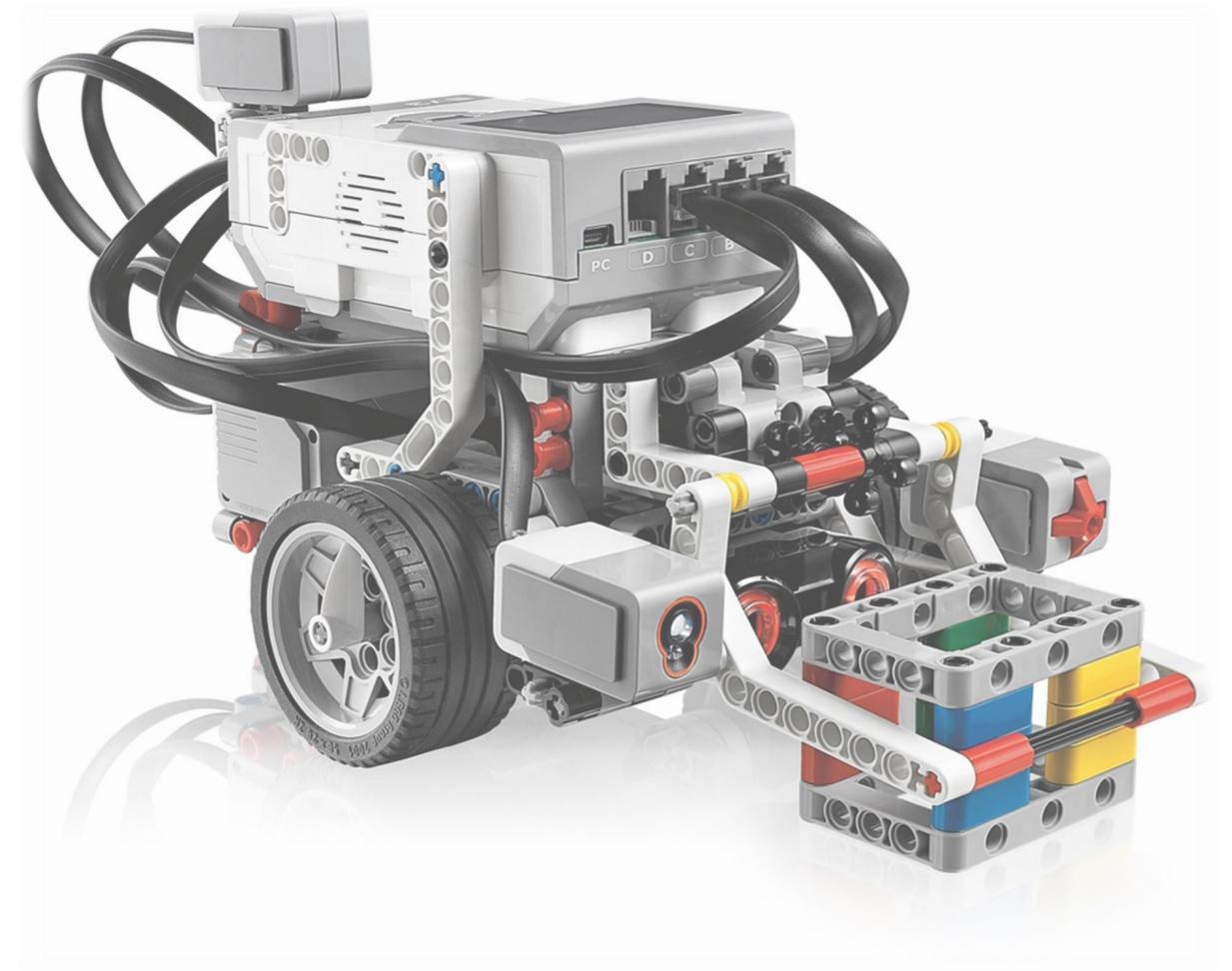
# Programmierung – Aufgabe 2



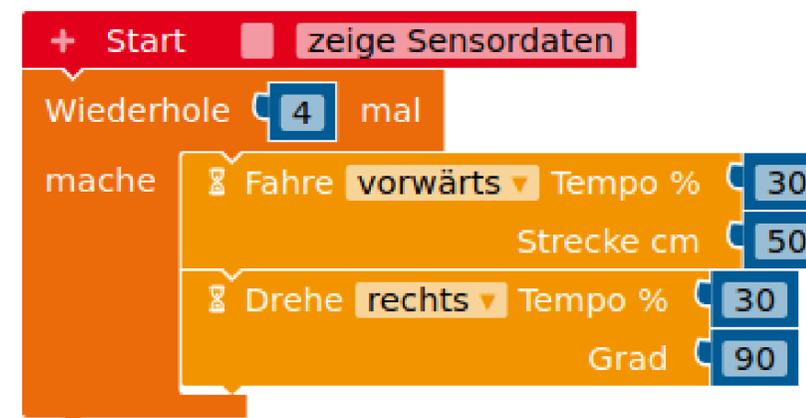
# Programmierung – Aufgabe 3

## Fahre ein Viereck

- Lasse den Roboter ein Viereck mit einer Kantenlänge von 50cm fahren.



# Programmierung – Aufgabe 3



Lösung mit einer Schleife ist übersichtlicher und das Programm ist einfacher zu verstehen.

# Verwendung von Sensoren

Die Augen und Ohren des Roboters



# Umgebung wahrnehmen / Sensoren

- Open Roberta / EV3
  - Farbsensor (Erkennung von Farben und Linien)
  - Berührungssensor (z.B. Erkennung von Hindernissen – bei Berührung)
  - Ultraschallsensor (z.B. Erkennung von Hindernissen – vor Berührung)
  - Kreisel sensor (z.B. Messung, wie weit sich der Roboter gedreht hat)

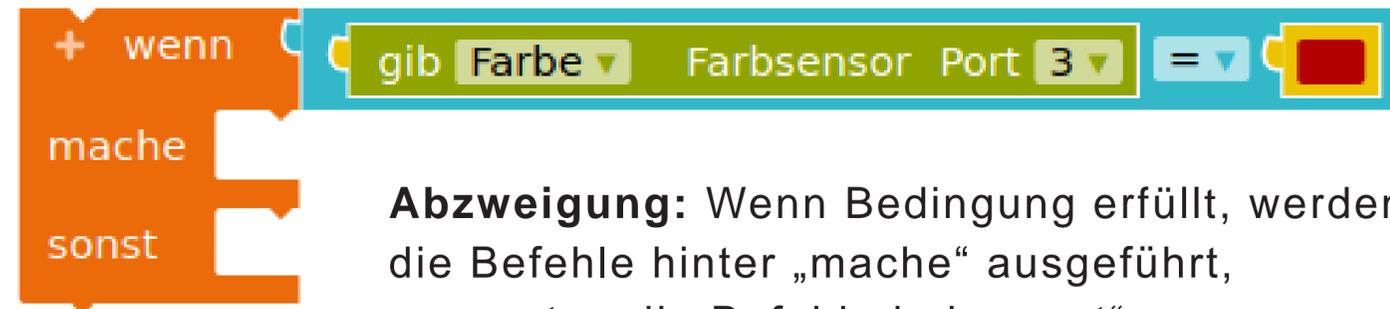


# Abfrage der Sensoren



## Auslesen der Sensoren

Vergleich des Sensors mit dem erwarteten Wert  
(hier: wenn Farbe von Sensor 3 gleich rot)



**Abzweigung:** Wenn Bedingung erfüllt, werden die Befehle hinter „mache“ ausgeführt, ansonsten die Befehle bei „sonst“.

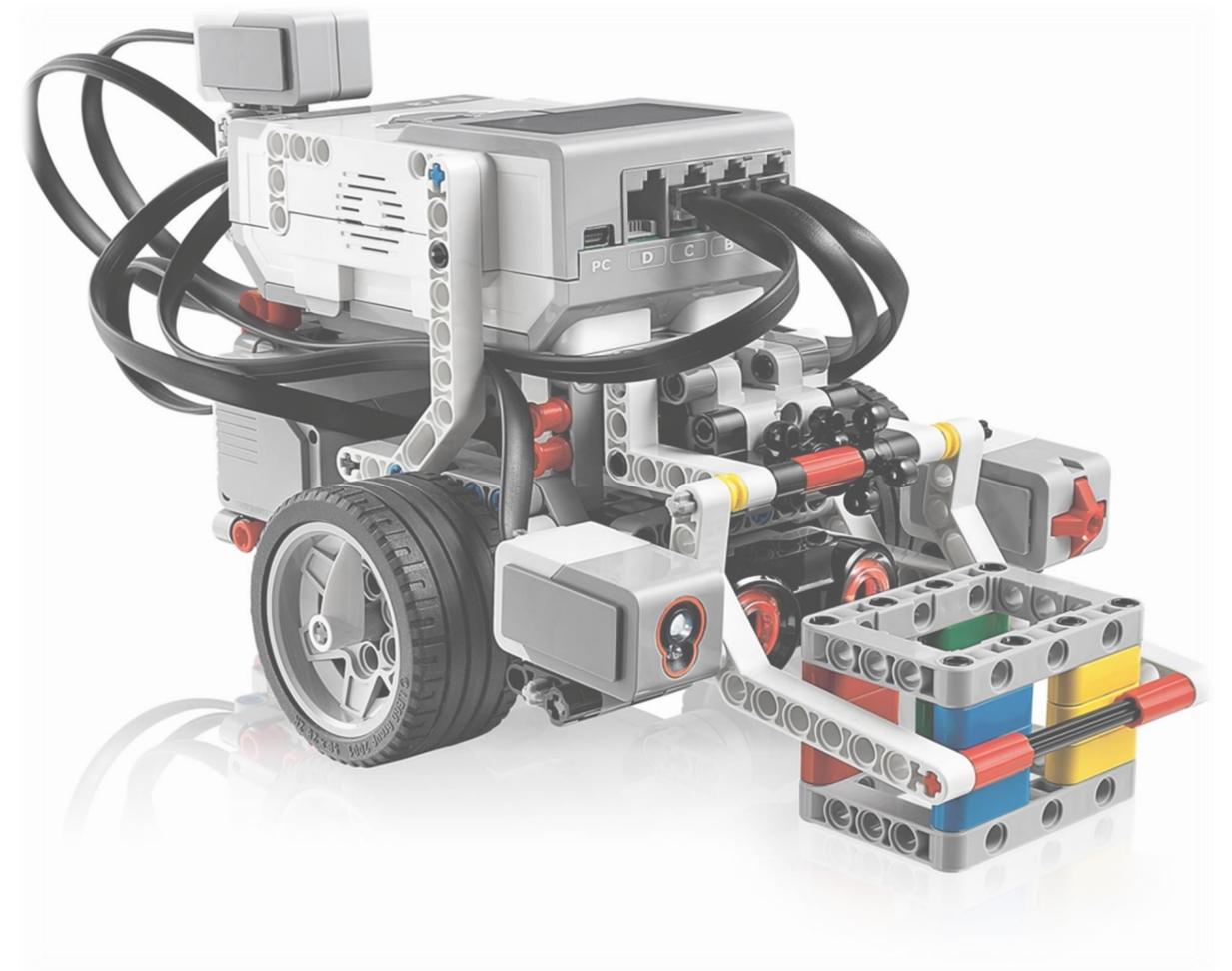


**Warten bis:** Das Programm läuft erst beim nächsten Block weiter, wenn die hier angegebene Bedingung erfüllt ist.

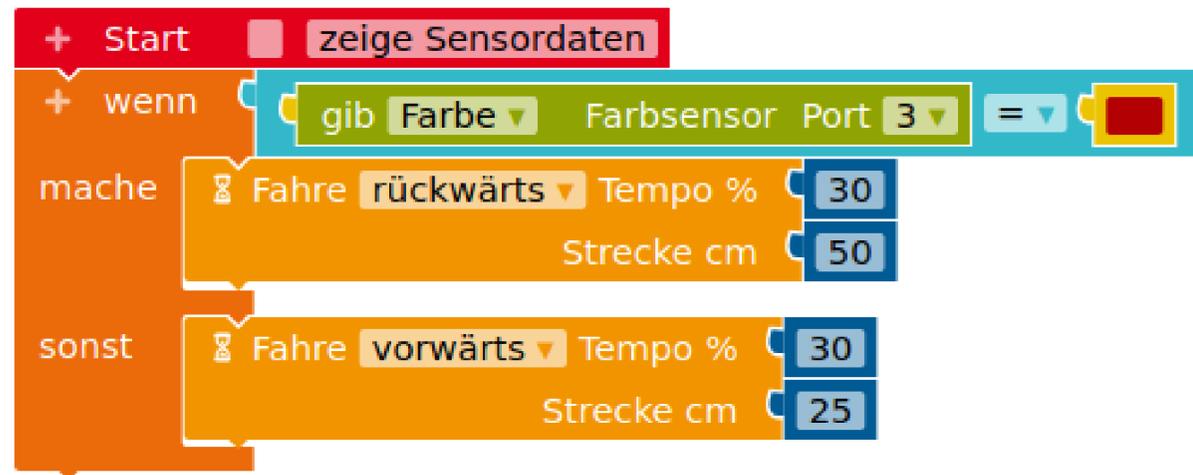
# Programmierung – Aufgabe 4

## Vor oder zurück?

- Wenn der Roboter auf rot startet, lass ihn 50cm zurück fahren
- Ansonsten lass den Roboter 25cm vor fahren



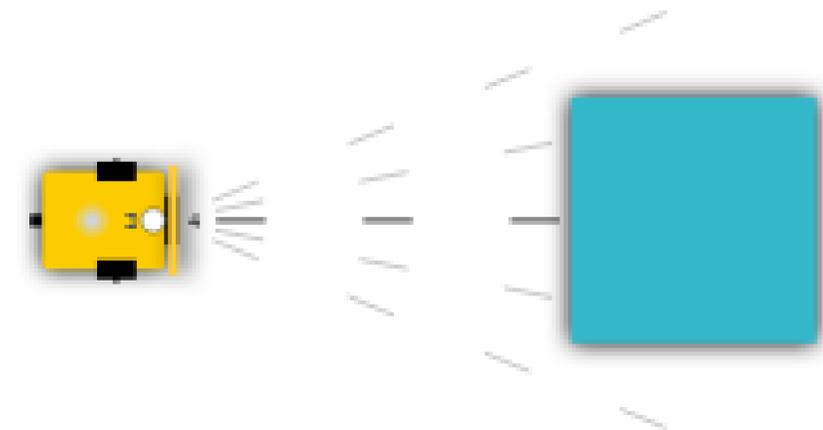
# Programmierung – Aufgabe 4



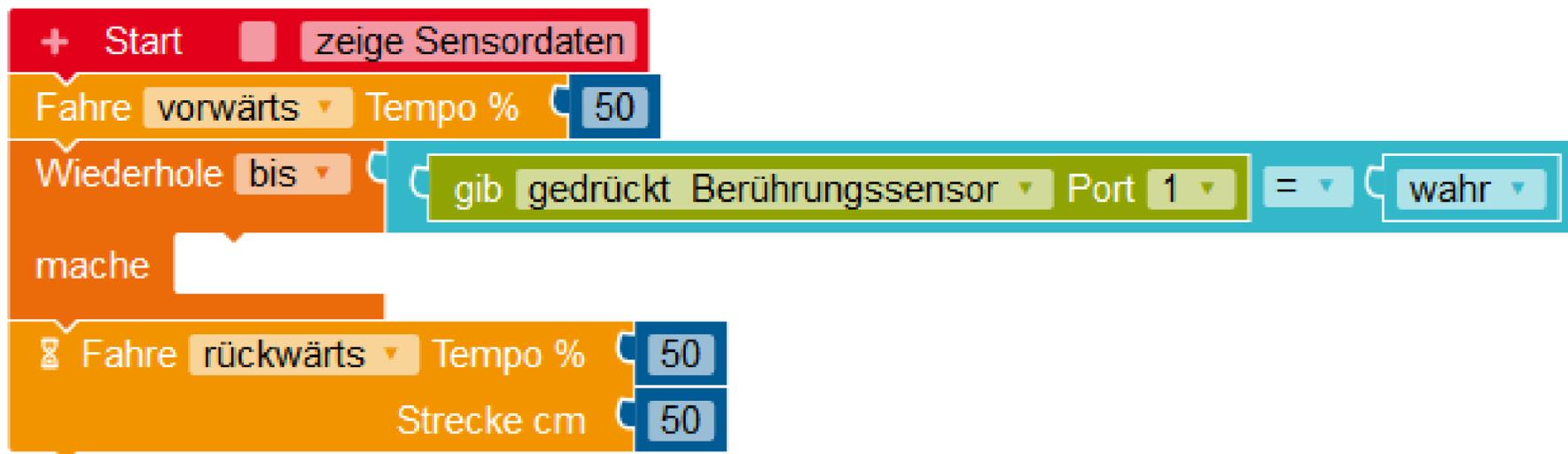
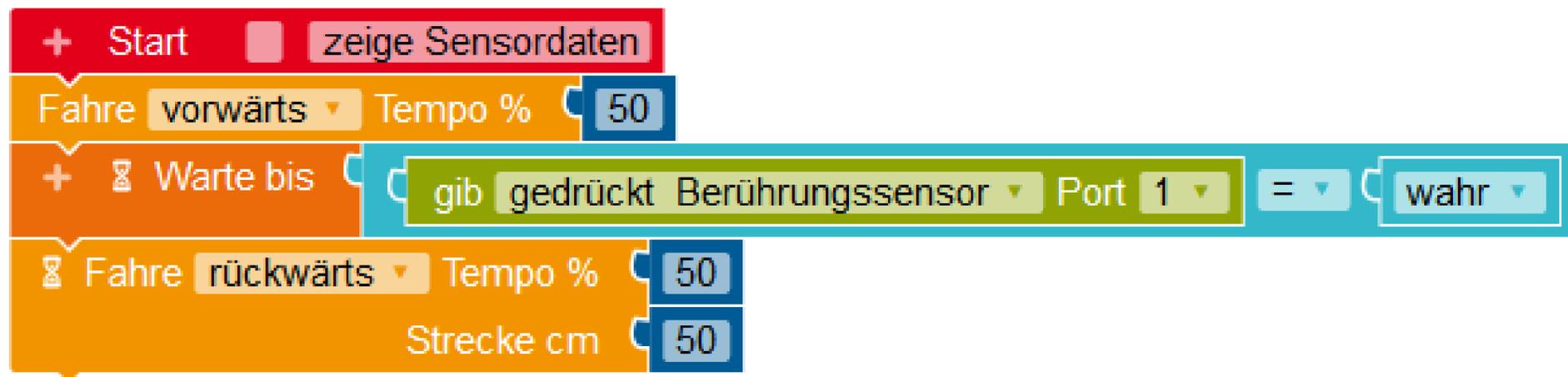
# Programmierung – Aufgabe 5

## Auf ein Hindernis reagieren

- Hindernis in den Weg stellen
- Fahre bis zum Hindernis
- Fahre 50cm rückwärts



# Programmierung – Aufgabe 5

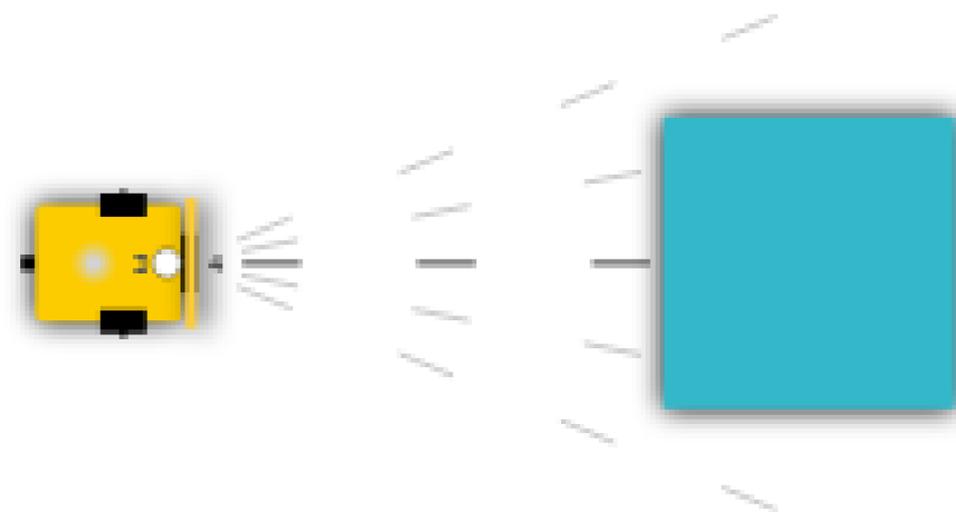


Verwendung einer Schleife statt des „Warten bis“-Blocks. Letztlich führt der Warten-Block ebenfalls eine Schleife aus, die den Sensor immer wieder abfragt.

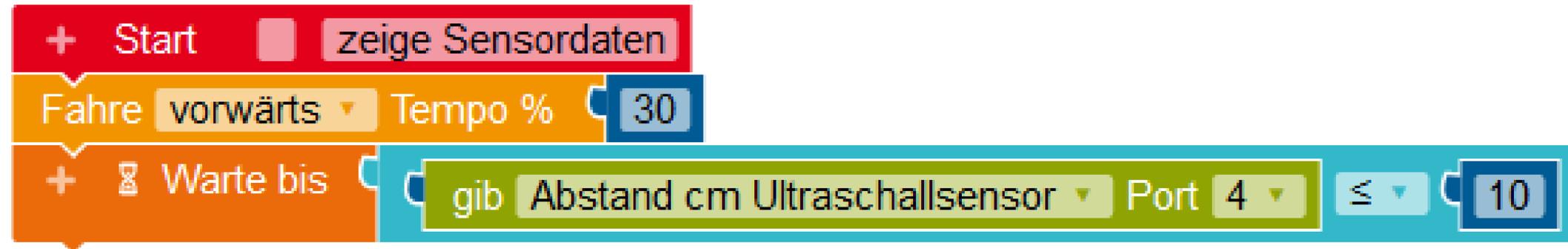
# Programmierung – Aufgabe 6

## Abstand halten

- Fahre vorwärts bis 10cm vor das Hindernis
- Verwendung des Ultraschallsensors



# Programmierung – Aufgabe 6



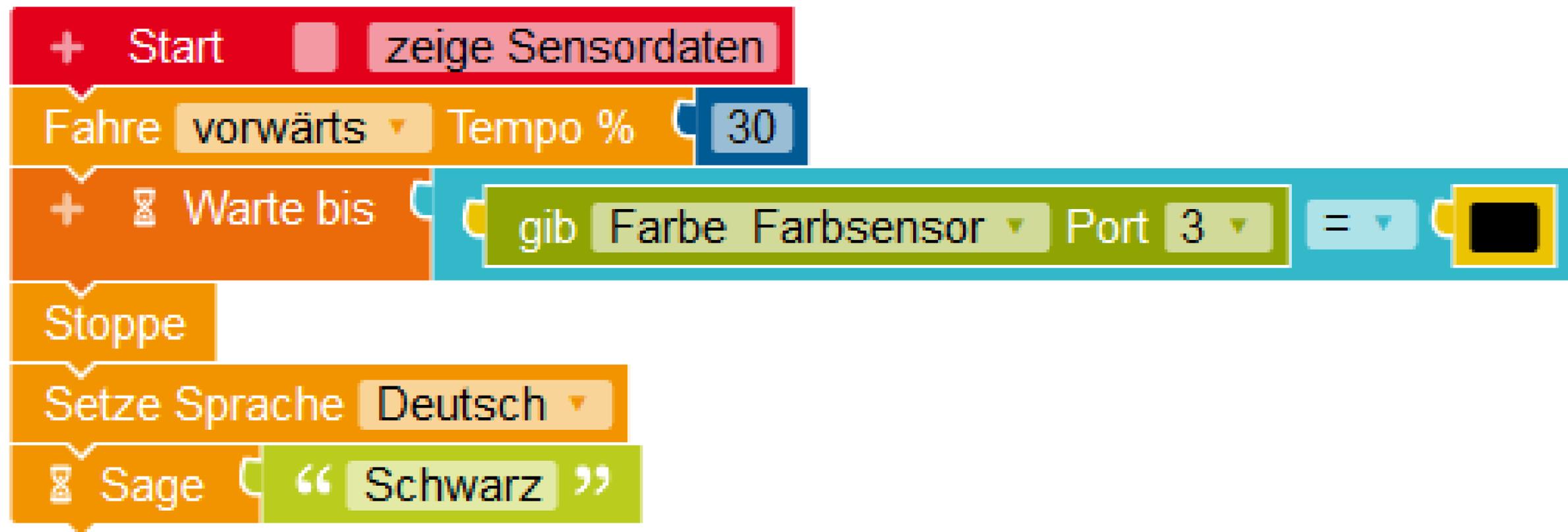
# Programmierung – Aufgabe 7

## Linien finden

- Fahre vorwärts in Richtung der schwarzen Linie
- Halte an der Linie mit Verwendung des Farbsensors
- *Zusatz: Lass den Roboter „schwarz“ sagen*



# Programmierung – Aufgabe 7



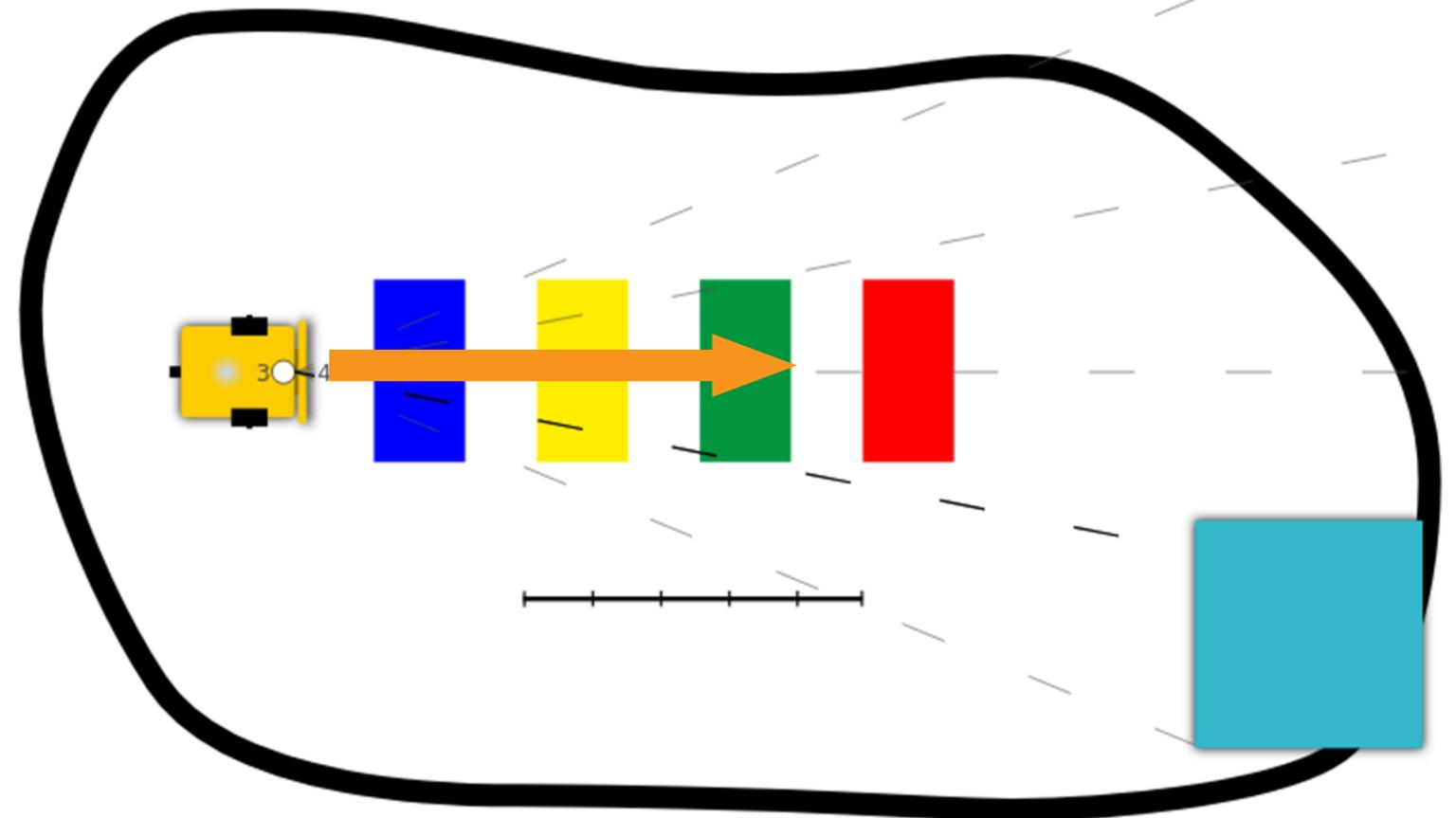
# Bonusaufgaben



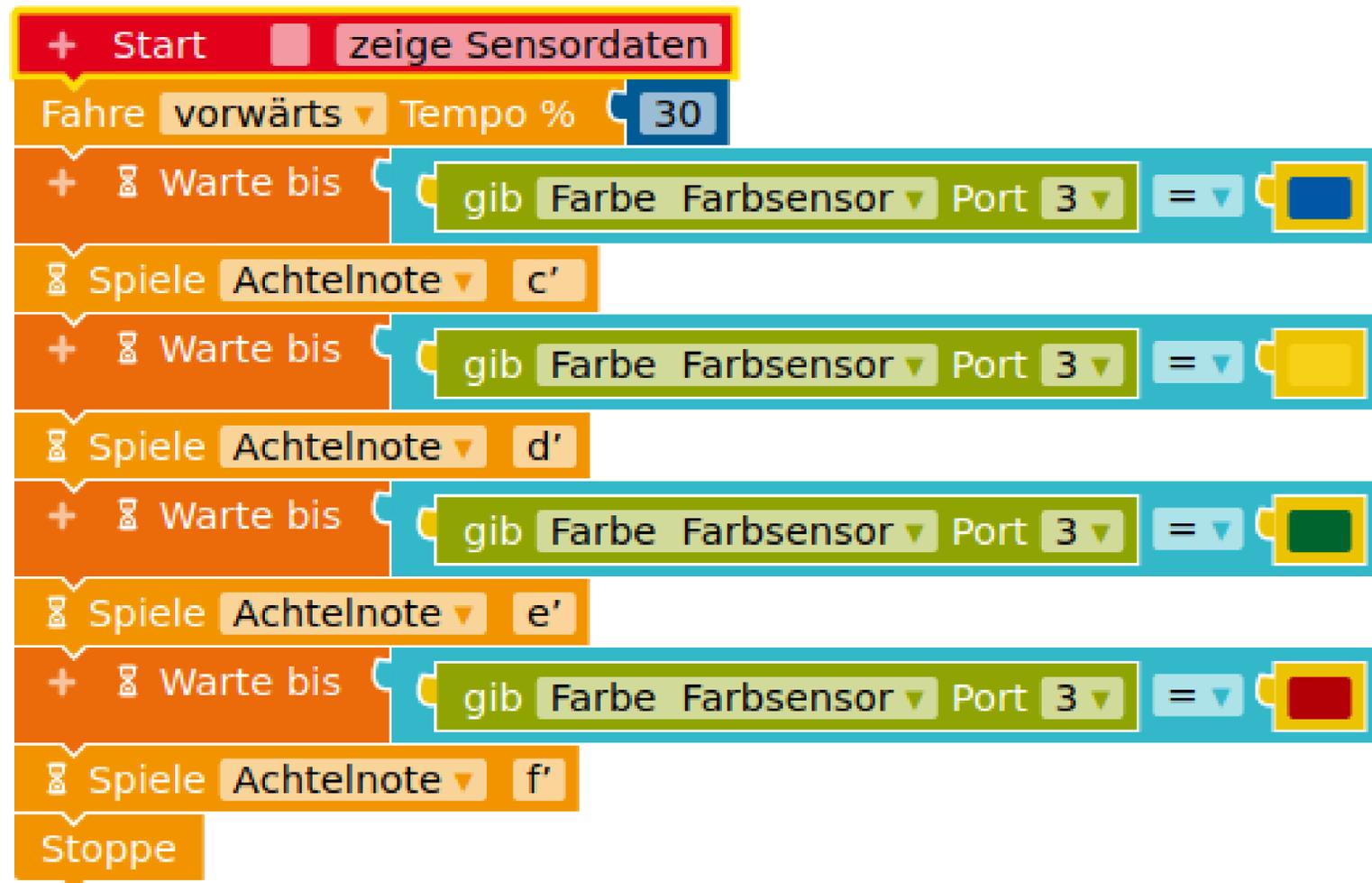
# Programmierung – Bonusaufgabe 1

## Spiele Musik

- Fahre über die vier farbigen Rechtecke und lasse den Roboter jeweils einen Ton spielen, wenn er die nächste Farbe erreicht.



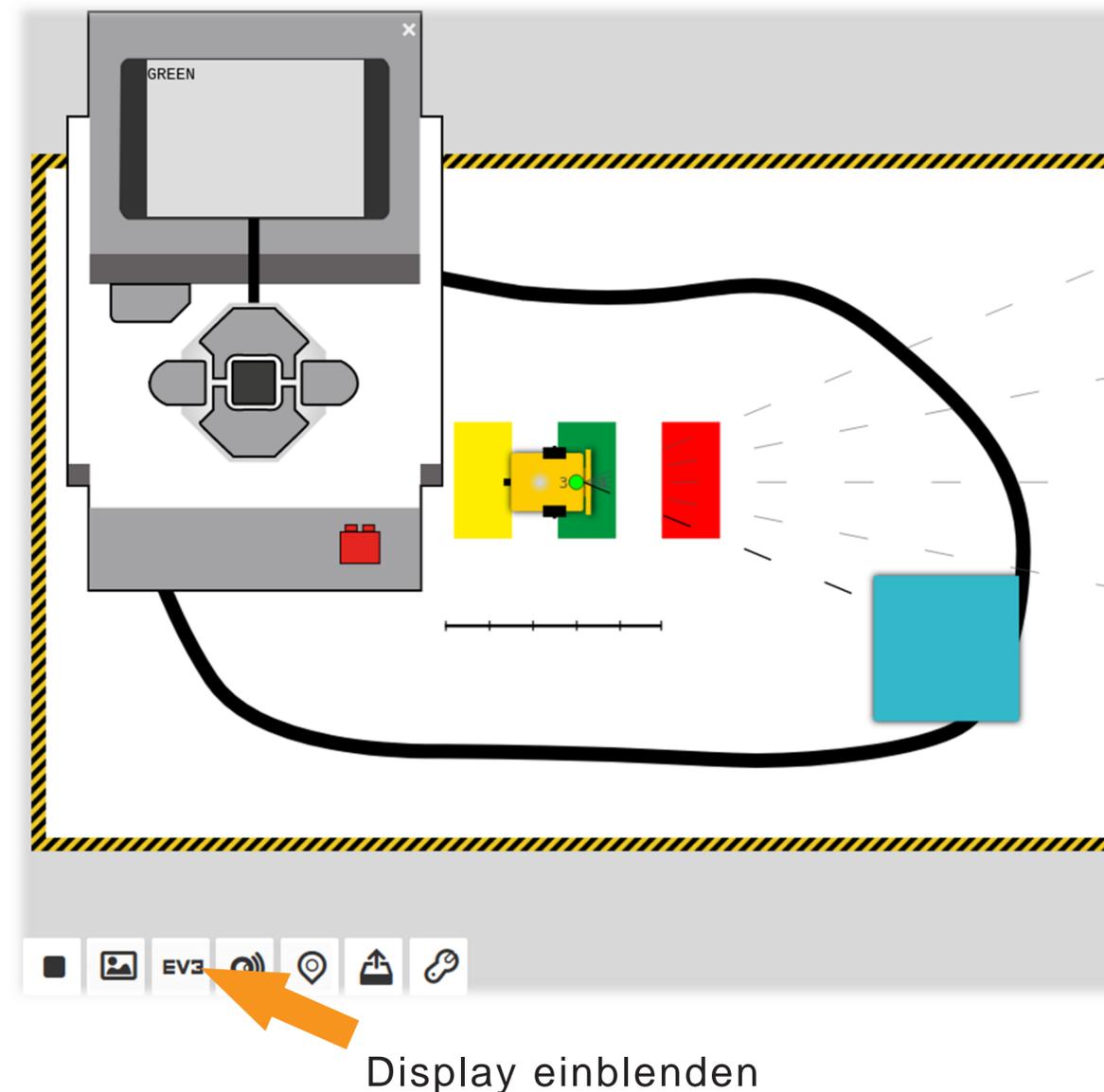
# Programmierung – Bonusaufgabe 1



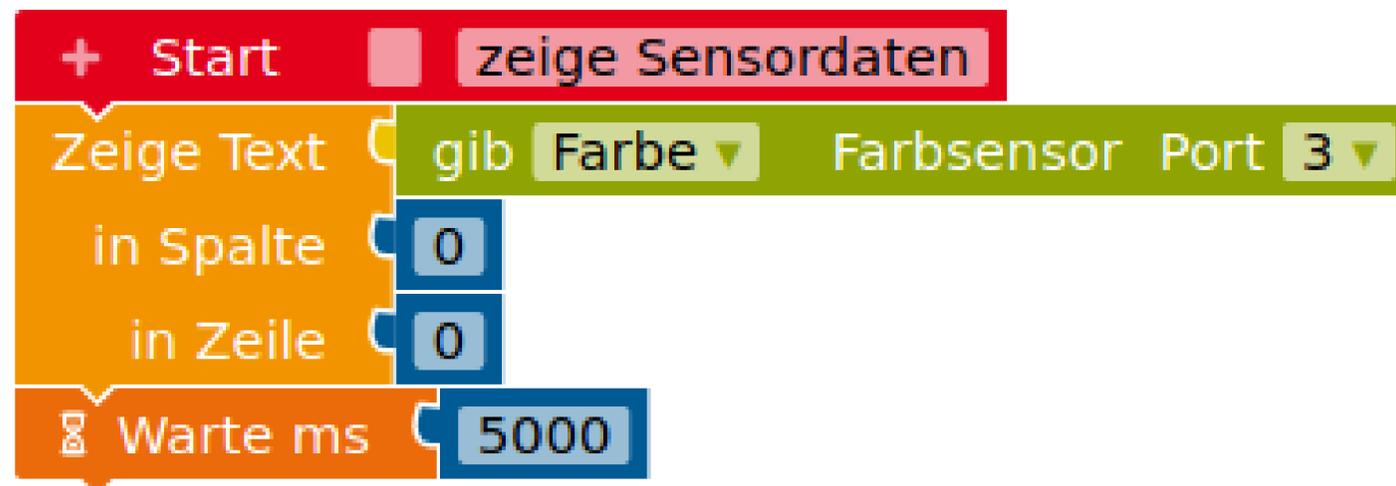
# Programmierung – Bonusaufgabe 2

## Farben erkennen

- Zeige die Farbe auf dem Display des Roboters an, auf der er beim Starten des Programmes steht.
- *Tipp: Nach der Anzeige muss eine Pause eingefügt werden, da das Programm sonst umgehend endet und die Anzeige gelöscht wird.*



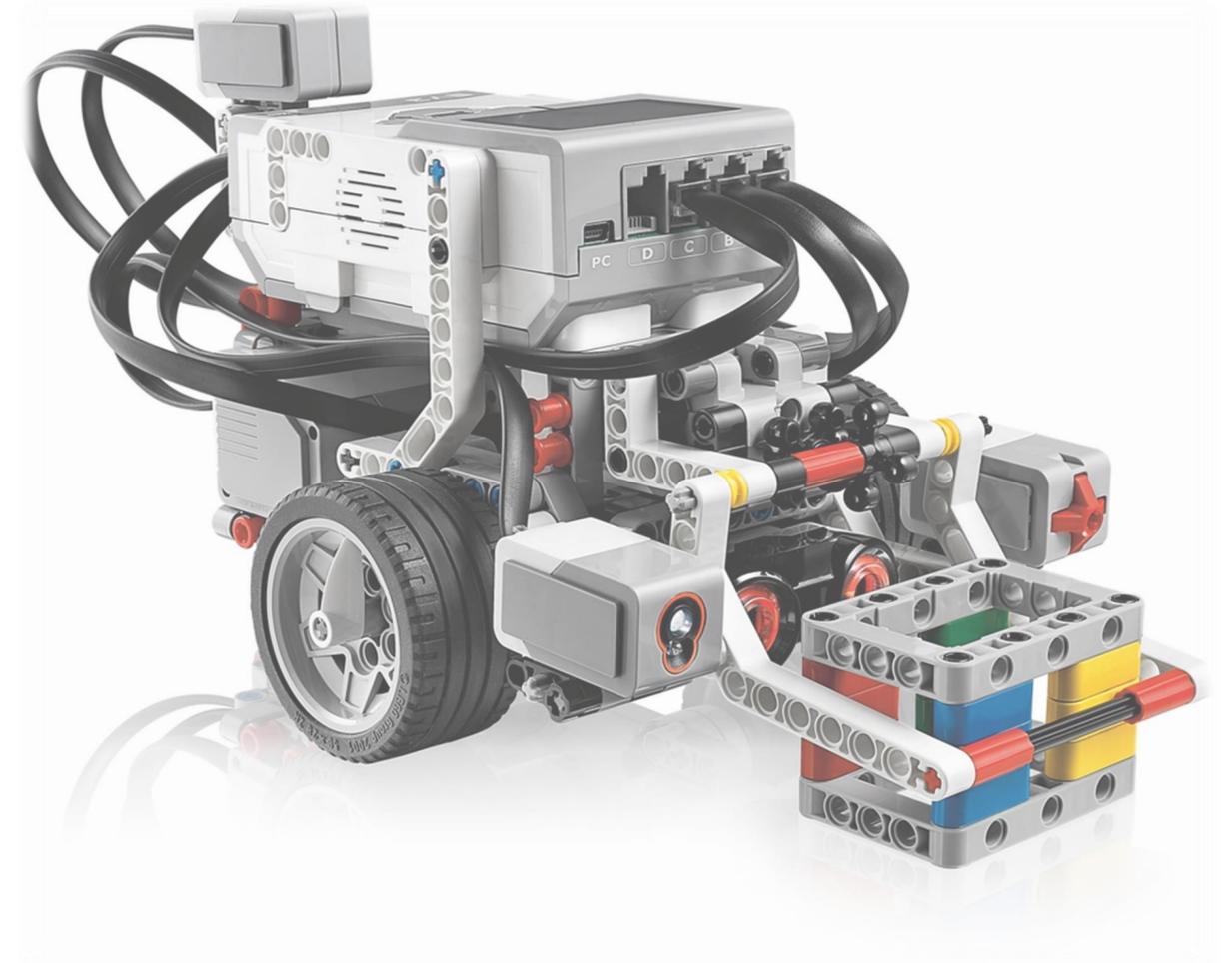
# Programmierung – Bonusaufgabe 2



# Programmierung – Bonusaufgabe 3

## Fahre ein Achteck

- Erinnere dich zurück an Aufgabe 3, in der ein Viereck gefahren wurde.
- Lasse den Roboter nun allerdings ein Achteck fahren.
- **Frage:** Wie hilft der Einsatz von Schleifen um die notwendigen Anpassungen am Programm zu reduzieren?

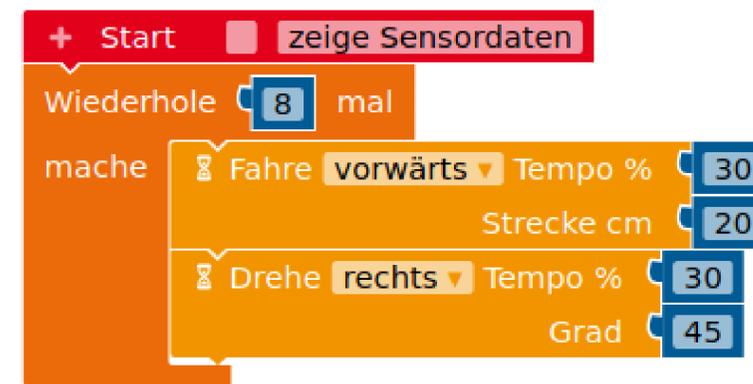


# Programmierung – Bonusaufgabe 3

Lösung des Vierecks mit einer Schleife:



Lösung des Achtecks mit einer Schleife:



Antwort: Mit einer Schleife reduzieren sich die Änderungen von einem Viereck auf ein Achteck, auf die Anpassung der Anzahl an Wiederholungen und der Grad der Drehung. Ohne Schleife müssen doppelt so viele Befehle verwendet werden wie zuvor.

# Vielen Dank

Viel Spaß beim weiteren Programmieren



# Referenten / Kontaktdaten



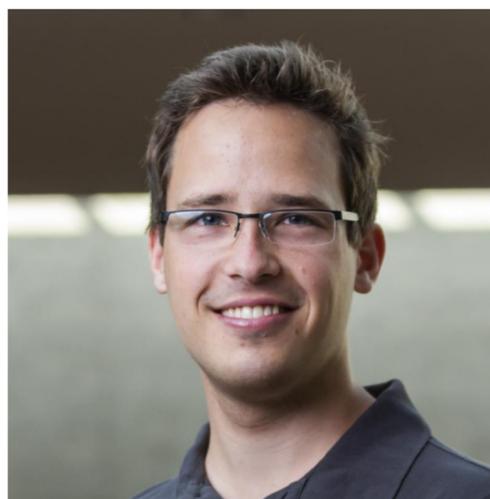
**Lukas Plümper**



**Egbert Langer**



**Jenny Kociemba**



**Tobias Schepers**

**Rückfragen gerne an**

Lukas Plümper

[lp@technik-begeistert.org](mailto:lp@technik-begeistert.org)

0176 / 98 331 332

Weitere Informationen unter:

<https://wro2021.de/mint-ec>





Schreibe deine Robotergeschichte