

Ablauf „Programmieren mit Erstklässlern“:

	Lernziele (Kompetenzen)	Materialien
<p>Optional: In der Aula: Roboter vorstellen, besprechen Was ist das? Wo gibt es Roboter? Habt ihr einen Roboter zu Hause? (Mähroboter, Staubsauger,...) Wie funktioniert ein Roboter? Woher weiß er, was er tun muss?</p>	<p>Begriffe kennenlernen: Roboter, Energie, Programm Roboter benötigen ein Programm, das ihnen sagt, was sie tun sollen. Programme werden von Programmierern erstellt („ausgedacht“) und sind im Roboter gespeichert.</p>	<p>Roboterfigur</p>
<p>Ihr sollt jetzt selbst Programmierer sein. Teppiche als Feld auslegen (3x4?) Freiwilliger steht auf einem Teppich, das Ziel auf einem anderen Teppich in derselben Reihe (Abb.1). Dies ist unser Roboter. Er kann immer nur von einem Teppich zum anderen gehen. Unser Roboter soll zum Ziel laufen. Welche Befehle müssen wir ihm geben? SuS nennen die Befehle (3mal nach vorne gehen, etwas aufheben) Befehlskarten vorstellen und in der genannten Reihenfolge an die Tafel hängen. Roboter starten – Befehle dabei nennen und an der Tafel mitverfolgen.</p>	<p>Erkennen der Teilschritte einer einfachen Handlung Anordnen der Teilschritte in eine sinnvolle Reihenfolge</p>	<p>Teppiche Freiwilliger (Erwachsener oder von Erwachsenen geführte Roboterfigur) Ziel (Obst? Kekse?...) Befehlskarten, Tafel Magnete</p>
<p>Roboter wieder in Startposition, Ziel an anderer Stelle, sodass Drehung erforderlich wird (Abb. 2) Was muss unser Roboter nun zusätzlich können? → Drehen nach links bzw. rechts SuS nennen Befehle, Befehlskarten anhängen, Roboter starten, Befehle und Ausführung mitverfolgen. Wichtig ist, dass die Drehung als eigener Schritt ausgeführt wird! Ab jetzt kann auch ein Schüler/Schülerin die Rolle des Roboters übernehmen. Es können mehrere Wege programmiert und verglichen werden. →</p>	<ul style="list-style-type: none"> - Erkennen neuer notwendiger Fähigkeiten, die der Roboter benötigt, um die gestellte Aufgabe erfüllen zu können - Unterscheiden von Rechts- und Linksdrehung - Perspektivübernahme – Relativität der Bezeichnungen Rechts und Links wahrnehmen und bei der Programmierung korrekt anwenden - Verstehen, dass ein Roboter gemäß seiner Programmierung handelt und Programmfehler nicht selbstständig verbessert 	

<p>vgl. Eckenhausen (Das Zahlenbuch, Klett Verlag)</p>	<ul style="list-style-type: none"> - Wege im Koordinatensystem vergleichen können - erkennen: Der Weg ist immer gleich lang, solange man nicht diagonal laufen kann - erkennen: es gibt mehr als eine „richtige“ Lösung - erlernen des Optimierungsprozesses bei Programmen: Programmieren, Testen, Verbessern 	
<p>Roboter in Startposition, auf dem Spielfeld werden Hindernisse verteilt, die umlaufen werden müssen (Abb. 3). „Auf diese Felder kann der Roboter nicht treten“. SuS nennen Befehle, Befehlskarten anhängen, Roboter starten, Befehle und Ausführung mitverfolgen. Optional: Neue Befehlskarte „springen“ einführen. Roboter in Startposition SuS nennen Befehle, Befehlskarten anhängen, Roboter starten, Befehle und Ausführung mitverfolgen. Beide Programme aus diesem Block miteinander vergleichen.</p>	<ul style="list-style-type: none"> - Wege im Koordinatensystem vergleichen können - erkennen: es gibt mehr als eine „richtige“ Lösung - erkennen: es gibt geschickte (kurze, einfache) und weniger geschickte (lange, umständliche) Programme 	<p>Hindernisse (z.B. Schulranzen, Bausteine als symbolisches Hindernis oder farbige Teppiche, Wellen aus Tonkarton o. ä.)</p>
<p>Im Klassenraum: Am IWB Run Marco! Starten. https://www.allcancode.com/web/hourofcode → play online (Button recht weit unten) → Sprache wählen → Spielfigur wählen → Dschungelszenario auswählen Jetzt ist Marco/Sophia unser Roboter. Wir müssen ihr/ihm genau sagen, was er/sie tun soll. Spiel bis Level 6 gemeinsam durchspielen.</p>	<ul style="list-style-type: none"> - Übertragen der Erkenntnisse auf ein Computerspiel - Kennenlernen einer grafischen Programmierumgebung - Programmieren innerhalb einer grafischen Programmierumgebung - Anwenden der bisher gewonnen Erkenntnisse, insbesondere des Optimierungsprozesses 	<p>Interaktives Whiteboard</p>
<p>In PA das Spiel Run Marco! so weit spielen, wie die Zeit reicht oder es interessant ist. Achtung: Bei Auftreten der Schleife benötigen die Kinder eine neue individuelle Einführung in dieses Programm-Element! Es wird erst in der zweiten Einheit gemeinsam erarbeitet.</p>	<ul style="list-style-type: none"> - Sichern der erworbenen Kenntnisse und Kompetenzen durch eigenständiges Arbeiten - evtl. selbstständiges Erweitern der Kompetenzen (Schleifen, Funktionen) 	<p>Tablets mit Run Marco! -App bzw. Tablets mit WLAN</p>

Befehlskarten: ein Schritt vorwärts: 10fach kopieren
 links drehen: 4fach kopieren
 rechts drehen: 4fach kopieren
 springen: 2fach kopieren
 etwas aufheben: 2fach kopieren (um verschiedene Programme parallel erstellen und vergleichen zu können)
 Stop des Programms: 2fach kopieren (um verschiedene Programme parallel erstellen und vergleichen zu können)

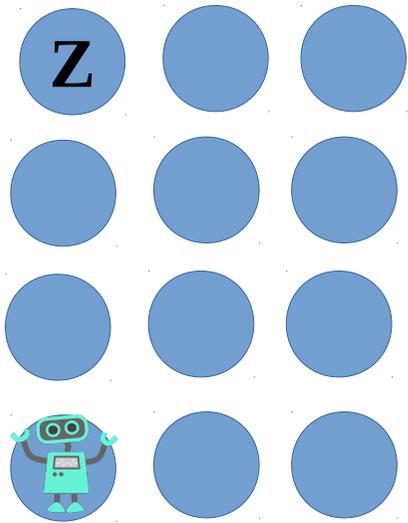


Abb.1

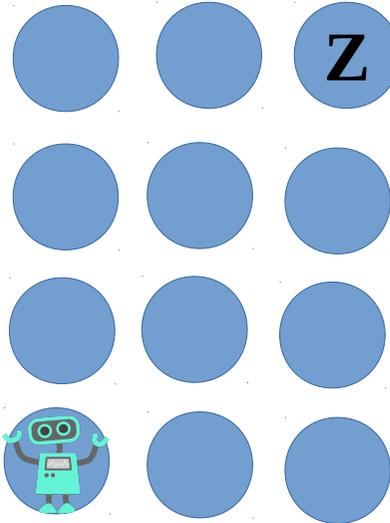


Abb.2

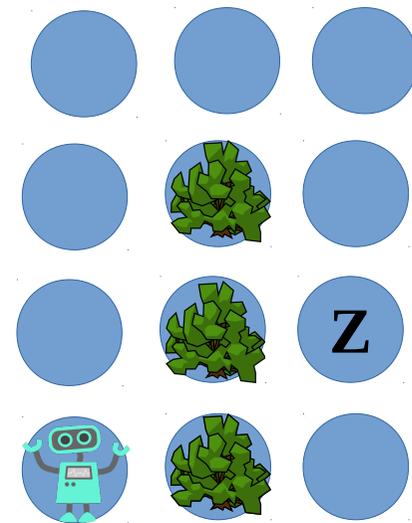


Abb.3

Abbildungen:
<https://openclipart.org>, [CC-0](https://creativecommons.org/licenses/by-sa/4.0/)
 - [Blue Robot](#) by [Scout](#)
 - [Bush](#) by [raemi](#)

Programmieren mit Erstklässlern von Christiane Schicke ist lizenziert unter einer Creative Commons Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International Lizenz.

