

Unterrichtsszenario

<u>Titel</u>: Algorithmen verfolgen und erstellen ¹

Praktische Informationen

Organisation:

Als Workshops

(Ideale) Anzahl von Lernenden: 4 pro Workshop

 5 Workshops mit je 4 Schülern, darunter einer zum Thema Robotik, werden in der Klasse eingerichtet

Alter der Lernenden:

10 bis 12 Jahren

Dauer des Szenarios:

• 4 Einheiten von 50 Minuten

Hauptdisziplin des Szenarios:

• Informatisches und algorithmisches Denken

Beschreibung

Ziel der Aktivität ist es, die Schüler dazu zu bringen, die von ihnen beobachteten Roboterbewegungen zu programmieren und die Eigenschaften eines Algorithmus durch Softwaremanipulation und Roboterbeobachtung zu entdecken.

Am Ende der Aktivitäten werden die Schüler in der Lage sein, einen Algorithmus zu schreiben, der einer Abfolge präziser Aktionen des Roboters entspricht, sowie diese Abfolge von Aktionen mit dem Lego Spike zu programmieren.

¹ Das Szenario ist inspiriert von dem von Elaimrany, K. und Angenot, L. erstellten Szenario. (2019) im Rahmen des PIAF Erasmus + Fortbildungsseminars, das im März und April 2019 in ULiège organisiert wird.













PIAF-spezifische FähigkeitenZiele

Spezifische PIAF-Kompetenzen (siehe PIAF-Kompetenzrahmen²)

5.1. **5. Umgang mit formalen Repräsentationen**

- → 5.1. Objekte oder Aktions-Sequenzen in einer formalen Repräsentation darstellen
- ⇒ Beobachtung: Darstellung von beobachteten Aktionen (die Bewegungen/Aktionen eines Roboters, die beobachtet werden) in einer textuellen Darstellung.
 - ⇒ Der Schüler ist in der Lage, den Algorithmus zu schreiben, den der Roboter auf der Grundlage seiner Beobachtungen ausführt.

5.2. **5. Umgang mit formalen Repräsentationen**

- → 5.2. Objekte oder Aktions-Sequenzen zwischen formalen Repräsentationen übersetzen
- ⇒ Programmierung: Übersetzung einer Abfolge von Aktionen aus einer textuellen Darstellung in eine Programmiersprache (Text → Aktionsblöcke).

Der Schüler wird in der Lage sein, die Bewegungen des Roboters auf der Grundlage schriftlicher Algorithmen zu programmieren.

4.2. **4. Objekte oder Aktions-Sequenzen bewerten**

- → 4.2. Zwei Aktions-Sequenzen anhand eines bestimmten Kriteriums vergleichen
- ⇒ Vergleich der Sprache von zwei Handlungssequenzen anhand eines Kriteriums.
 - ⇒ Der Schüler ist in der Lage, die Sprache verschiedener Handlungssequenzen zu vergleichen, die nach dem vorgegebenen Kriterium (sprachliche Genauigkeit und Reduzierung der Anzahl der Befehle) geschrieben wurden.

Diese Kompetenz wird nicht bewertet.

3.3. **3. Eine Aktions-Sequenz steuern**

- → 3.3. Eine einfache Bedingung in eine Aktions-Sequenz einfügen
- ⇒ Der Schüler kann eine einfache Bedingung beim Schreiben eines Algorithmus und eines Programms verwenden..

3.1. **3. Eine Aktions-Sequenz steuern**

→ 3.1. Eine Aktions-Sequenz eine bestimmte Anzahl mal wiederholen

² Die in dieser Tabelle dargestellten Kompetenzen sind dem Referenzrahmen entnommen, der im Rahmen des Erasmus+ PIAF-Projekts erstellt wurde (verfügbar unter: https://piaf.loria.fr/contributions/). Sie beziehen sich auf die Nummern der Kompetenzen im Referenzrahmen (z. B. 5. Übersetzen von einer Darstellung in eine andere).













⇒ Der Schüler kann die Schleife beim Schreiben eines Algorithmus und eines Programms verwenden.

4.3 4. Objekte oder Aktions-Sequenzen bewerten

- → 4.3 Eine Aktions-Sequenz anhand eines bestimmten Kriteriums verbessern
- ⇒ Der Schüler kann einen schriftlichen Algorithmus verbessern, um ihn klarer und genauer zu machen.
- ⇒ Der Student kann herausfinden, wie man ein Programm so verbessern kann, dass es möglichst wenige Zeilen hat.

Diese Kompetenz wird nicht bewertet.

6.1 **6. Eine Aktions-Sequenz iterativ erstellen**

- → 6.1. Überprüfen, ob eine Aktions-Sequenz ein vorgegebenes Ziel erreicht
- ⇒ Testen Sie ein Programm mit der Lego Spike Software.
 - ⇒ Die Schüler können ihr Programm auf Spike testen.

6.2. **6. Eine Aktions-Sequenz iterativ erstellen**

- → 6.2. Fehler in einer Aktions-Sequenz erkennen
 - ⇒ Die Schüler sind in der Lage, Fehler in ihrem Algorithmus zu erkennen.

6.3 **6. Eine Aktions-Sequenz iterativ erstellen**

- → 6.3. Eine Aktions-Sequenz korrigieren, um ein gegebenes Ziel zu erreichen
- ⇒ Korrektur des Programms oder eines geschriebenen Algorithmus, bis er mit den Bewegungen des Modellroboters übereinstimmt.

Die Schüler sind in der Lage, das Programm der Lego Spike Software oder den geschriebenen Algorithmus so zu korrigieren, dass er mit den Bewegungen des Roboters des Lehrers übereinstimmt.

<u>Voraussetzungen für die Aktivität</u>

- Tipp: Eine Reihe praktischer Aktivitäten zum Bau von Robotern mit dem Lego Spike Prime-Bausatz (Bausteine, Motoren und Sensoren) und der auf Scratch basierenden Softwareoberfläche Spike Education.
- Schulen Sie die Schüler im Umgang mit der Hilfefunktion der Spike Education Software.











Benötigte Materialien

Technisch	Didaktisch
Zwei Computer mit der Software zur Programmierung des Lego Spike Roboters (https://education.lego.com/en-us/downloads/spike-prime/software)	 ⇒ Blätter, auf die die Schüler ihre Algorithmen schreiben (Anhang 4).
Je nachdem, wie gut die Schülerinnen und Schüler die Scratch-Software beherrschen, entscheidet die Lehrkraft, ob sie die Lernenden alle Blöcke durchsuchen lässt (sehr fortgeschrittenes Niveau), ob die Lernenden die vorgegebenen Blöcke durchsuchen müssen (fortgeschrittenes Niveau - Anhang 6) oder ob die Lernenden die vorgegebenen Blöcke in eine Reihenfolge bringen müssen (Anfänger-Niveau - Anhang 5).	
Drei Lego Spike-Roboter Modell: Rhino Spike. Für die Beobachtungszeitpunkte (Schritt 1): Die Lehrkraft hat zwei Roboter programmiert (siehe Programme in den Anhängen 1 und 3).	Schreibmaterial: Blätter, Stifte und Fußmatten, damit sich der Roboter besser bewegen kann.
Für die Vergleichsmomente (Schritt 3) hat der Lehrer nur einen Roboter, aber beide Projekte in der Software geöffnet, um seine Programmierung leicht mit der von Gruppe 1 und dann mit der von Gruppe 2 vergleichen zu können.	
FER ACE Description Descr	









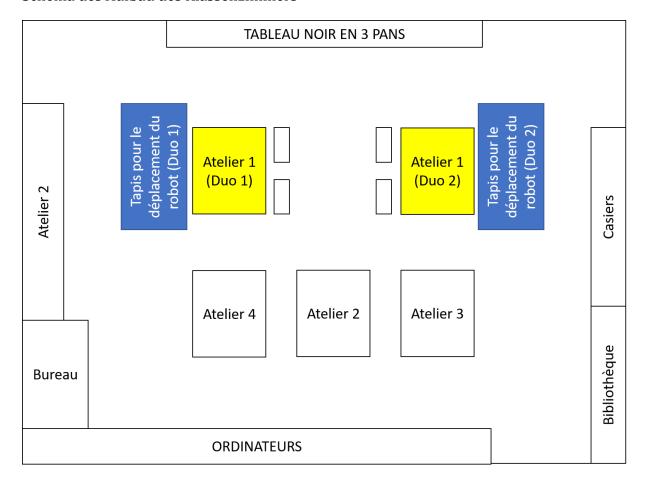




Organisation des Klassenzimmers

- ⇒ Fünf Gruppen mit je vier Workshops wechseln sich über mehrere Wochen ab, so dass jeder den "Roboter"-Workshop durchläuft.
 - 1) WORKSHOP 1: PIAF Lego Spike Aktivität
 - 2) WORKSHOP 2: Thema Ihrer Wahl
 - 3) WORKSHOP 3: Thema Ihrer Wahl
 - 4) WORKSHOP 4: Thema Ihrer Wahl

Schema des Aufbau des Klassenzimmers













Szenarios (Abfolge der Aktivitäten)

Aktivitäten	Beschreibung (Anweisungen - Rolle des Ausbilders - Aktivität des Lernenden)	PIAF- Kompetenzen (Nr.) und andere Fähigkeiten
Aktivität 1 - Beob	achten des Roboters und Schreiben von Algorithmen (50')	
1. Erläuterung der Anweisungen und Schreiben der Algorithmen (20')	Erste Beobachtung der Bewegungen des Roboters und Schreiben der Algorithmen Anweisungen des Lehrers "Ich habe zwei Roboter so programmiert, dass sie eine Reihe von bestimmten Aktionen ausführen. Deine Aufgabe ist es, einen der Roboter sehr genau zu beobachten, um seine verschiedenen Aktionen in Etappen zu ordnen und sie in einfachen Sätzen aufzuschreiben (Beispiel: Der Roboter bewegt sich 30 cm rückwärts)". Material: Roboter, Matte, Blatt Paier (Anhang 4) und ein Bleistiff pro Paar Ablauf der Aktion 1. Weisen Sie jedem Paar einen Tisch, eine Matte und einen (bereits programmierten) Roboter zu. 2. Geben Sie ihnen das Blatt, auf das sie ihre Algorithmen schreiben sollen (Anhang 4). 3. Zeigen Sie jeder Gruppe zum ersten Mal die Bewegung des Roboters. Die Schülerinnen und Schüler sollten sich einigen und in Gruppen die verschiedenen Aktionen (den Algorithmus), die der Roboter ausführt, aufschreiben.	5.1. (3.1.)









	Jeder Roboter ist in Bezug auf die Aktionsparameter unterschiedlich programmiert, hat aber die gleichen Aktionen (z. B. x oder y cm vorwärts fahren, x oder y Sekunden lang nach rechts abbiegen). Algorithmus von Roboter Nr. 1: siehe Anhang 1. Algorithmus von Roboter Nr. 2: siehe Anhang 1. Die Gruppen können das Roboterprogramm so oft wie nötig ausführen. In dieser Phase ist es wichtig, dass jede Gruppe den Roboter des Nachbarn nicht sehen kann. Die Schüler können Geräte verwenden, um z. B. zurückgelegte Entfernungen zu messen. Von einem Lehrer vorgeschlagene Alternative: Geben Sie die gleichen Aktionen mit den gleichen Parametern vor und bitten Sie jede Gruppe, ihren Algorithmus zum Vergleich an die Tafel zu schreiben.	
2. Vergleich der Bewegungen der beiden Roboter (20')	Um die Unterschiede in den Parametern und die Ähnlichkeit der Aktionen zu verdeutlichen, bittet der Lehrer die Schüler, sich zu versammeln und den Vergleich des Verhaltens der beiden Roboter zu beobachten.	4.2.
	Die Programme der einzelnen Roboter werden nacheinander gestartet. Die Schülerinnen und Schüler beider Gruppen werden gebeten, sich zu dem zu äußern, was	5.1.
	sie gerade beobachtet haben. Die Schüler sollen zu dem Schluss kommen, dass die	(3.1.)
	Handlungen zwar identisch sind, nicht aber die Parameter (Entfernung, Zeit). "Führen sie die gleichen Handlungen aus? "Welche Unterschiede stellen Sie fest? Können wir sagen, dass es die Handlungen sind, die anders sind?	(6.2.)
	Die Schüler werden dann aufgefordert, in ihre jeweiligen Werkstätten zurückzukehren, um	6.3.
	ihren Algorithmus so zu verfeinern, dass er möglichst genau der Realität ihres Roboters entspricht.	











3. Hervorhebung der Kriterien für die Erstellung von Algorithmen (15')	Sie werden zunächst aufgefordert, die Kriterien für das Schreiben eines Algorithmus hervorzuheben: "Du hattest die Möglichkeit, die Anweisungen zu schreiben, die der Roboter erhalten hat, um das auszuführen, was du beobachtet hast. Welches sind Ihrer Meinung nach die Kriterien, damit diese Anweisungen verstanden und korrekt ausgeführt werden? Die Lehrkraft kann beschließen, die von den Gruppen erstellten Algorithmen zu vergleichen, um sie zu diskutieren und die Kriterien für einen guten Algorithmus hervorzuheben: - Der Algorithmus ist in klar definierte Schritte unterteilt. - Die Reihenfolge der Schritte/Aktionen des Algorithmus richtet sich nach der Reihenfolge der Bewegungen des Roboters. - Das verwendete Vokabular ist präzise (Entfernungsangaben, Zeitangaben usw.) und unzweideutig.	(4.2.) 4.3. (3.1.)
	einführen (das kürzeste Programm schreiben), um zu zeigen, dass sich bestimmte Verhaltensweisen wiederholen. Die Schülerinnen und Schüler werden aufgefordert, ihre Handlungsreihenfolge auf der Grundlage dessen zu ändern, was während der Gruppenbildung gesagt wurde.	
4. Testen der Algorithmen mit der	Der Lehrer erklärt den Schülern, dass sie mit Hilfe einer Software überprüfen müssen, ob das Geschriebene mit den Aktionen des Roboters übereinstimmt.	5.2.
Lego Spike Software (25')	"In Aufgabe 1 haben Sie die Bewegungen des Roboters anhand von zuvor erstellten Programmen beobachtet. Jetzt sind Sie dran! Ihr werdet die Programmiersprache Scratch verwenden und den Roboter auffordern, die Aktionen auszuführen, die ihr in eurer Gruppe geschrieben habt."	6.1. (6.2.) (6.3.)
	Die Lehrkraft öffnet die Spike-Software auf dem Computer jedes Paares und überprüft, ob jeder Roboter angeschlossen ist. Die Lehrkraft öffnet das Projekt auf jedem Computer mit den verschiedenen Blöcken, die von den Schülerinnen und Schülern verwendet werden können	(3.1.)













	(Anhang 5 oder Anhang 6).	
	Dann lässt er die Paare die Software so manipulieren, dass sie das vorgegebene Ziel erreichen: den geschriebenen Algorithmus in die Scratch-Sprache zu übersetzen und ihn mit ihrem Roboter zu testen.	
	Die Schüler können das, was sie programmiert haben, mehrmals testen. Sobald sie denken, dass sie fertig sind, informieren sie die Lehrkraft.	
	Antizipation von Schwierigkeiten und möglichen Lösungen	
	1) Probleme beim Verschieben der Blöcke mit der Maus (zeigen Sie den Kindern, wie man sie direkt benutzt);	
	2) Technisches Problem mit dem Roboter (versuchen Sie selbst es zu lösen);	
	3) Unverständnis des auf den Blöcken verwendeten Vokabulars (Erklärungen/Übersetzungen/Synonyme);	
	Anmerkung: Es wurde beschlossen, die Drehungen in Sekunden zu programmieren, da die Gradzahl nicht der vom Roboter erreichten Amplitude entspricht.	
5. Zusammenführen	Die Lehrkraft fasst die beiden Duos zusammen und zeigt zunächst die Aktionen des Roboters	3.1.
der Erkenntnisse	von Duett 1 im Vergleich zu denen seines Roboters (und dessen Programm entspricht). Dann werden die Aktionen des Roboters des Duos 2 mit denen seines Roboters (zweites	4.2.
aus Aktivität 1 (20')	Programm) verglichen.	(4.3.)
	In diesem Stadium sind mehrere Beobachtungen möglich:	(5.2.)
	 Die Roboter tun genau das, was die Roboter des Lehrers tun. In diesem Fall wird der Lehrer vorschlagen, die Programme der Duos mit denen seiner Roboter zu vergleichen, 	6.1.
	um neue Merkmale eines guten Algorithmus hervorzuheben (siehe oben) und die	(6.2.)
	Schleife (Optimierung - Verringerung der Anzahl der Codezeilen) hervorzuheben, wenn dies nicht geschehen ist: "Wie können wir dein Programm reduzieren?", "Gibt es Aktionen, die zusammengefasst werden könnten?	(6.3.)
	Falls nötig, kann die Lehrkraft auch ein oder zwei Beispiele eines Programms mit einer	











	,	
	Schleife zeigen (nicht die der Roboter hier), um sie gemeinsam zu betrachten.	
	 Der/die Roboter tun nicht genau das, was der Roboter des Lehrers tut. In diesem Fall bringt der Lehrer die Schüler dazu, über die Ursache des Problems nachzudenken: Die geschriebenen Aktionen sind nicht klar genug und/oder das Programm entspricht nicht dem des Roboters des Lehrers. 	
	Die Lehrkraft schlägt dann vor, dass die Schüler ihre geschriebenen Algorithmen vergleichen. Der Lehrer lässt die Schüler darüber nachdenken, was einen Algorithmus klarer und präziser machen könnte (Merkmale eines guten Algorithmus): "Was sind die wichtigsten Elemente für das Schreiben eines Algorithmus?" Dazu kann der Roboter des Lehrers verwendet werden, um zu vergleichen, was geschrieben steht und was der Roboter tut. Sie kann die Schülerinnen und Schüler auch zum Nachdenken über die Schleife anregen: "Kann dein Algorithmus reduziert werden oder hast du einen Weg gefunden, es zu tun? Nach der Korrektur wird er den Schülern vorschlagen, ihren neuen Algorithmus mit der Software zu testen."	
	Wenn die Algorithmen am Ende eindeutig die Aktionen des Roboters zeigen, schlägt er vor, die Programme der Paare zu betrachten, die nicht mit den Aktionen des Modellroboters übereinstimmen. Er/sie wird seinen/ihren Roboter benutzen, um dessen Aktionen mit denen des Schülerprogramms zu vergleichen.	
	In dieser Phase wird die Lehrkraft in der Lage sein, in Worte zu fassen, was ein Algorithmus ist (= eine genaue Beschreibung einer Abfolge von Aktionen, die ausgeführt werden müssen, um ein Ziel zu erreichen).	
Erweiterung	Diese Aktivität kann auch mit der Programmierung in Anhang 2 wiederholt werden. Hier wird eine zusätzliche Schwierigkeit deutlich: die Schleife in der Schleife.	
Aktivität 2 - Beobo	achten des Roboters und Schreiben von Algorithmen mit der Bedingung (50')	
1. Schauplatz (50')	Diese Aktivität durchläuft genau die gleichen Schritte wie Aktivität 1. Die einzigen	5.1.
	Unterschiede sind die verwendeten Programme, denn diese beinhalten die Bedingung.	3.3.
	Wie in Aktivität 1 werden zwei Roboter mit denselben Aktionen, aber mit unterschiedlichen Werten programmiert (Anhang 3). Jedes Paar beobachtet das Verhalten des Roboters, der ihm	(3.1.)









		((0)
	zugewiesen wurde. Dazu stellt die Lehrkraft den Roboter vor eine Wand und aktiviert ihn. Die Schülerinnen und Schüler können das Roboterprogramm mehrmals ausführen, um den Algorithmus auf das von der Lehrkraft zuvor ausgeteilte Blatt Papier zu schreiben (Anhang 4).	(6.2.) (6.3.)
	Anschließend wird ein Treffen der beiden Gruppen organisiert, damit sie das Verhalten der beiden Roboter vergleichen können. Die erwartete Schlussfolgerung ist, dass die Aktionen identisch sind, aber die Werte unterschiedlich sind. Die Schüler werden dann aufgefordert, ihre schriftlichen Algorithmen zu spezifizieren.	
	Diese erste Phase endet mit dem Vergleich der Algorithmen der Duos. Während dieser Diskussion achtet die Lehrkraft darauf, dass die Bedingung betont wird: "Wie hast du den Moment ausgedrückt, in dem der Roboter auf die Wand zugeht? Es ist möglich, dass die Schüler statt einer Bedingung einen Abstand gesetzt haben. In diesem Fall stellt der Lehrer den Roboter von der Wand weg und führt das Programm aus. Die Schüler werden erkennen, dass an dieser Stelle kein Abstand programmiert ist, und müssen einen anderen Weg finden, um diesen Schritt auszudrücken. Ein weiteres Element, das die Schüler herausfordern soll, ist die Tatsache, dass das Programm jedes Mal neu startet, wenn die Taste gedrückt wird.	
	Alternative, die von einer Lehrkraft vorgeschlagen wird: dieselben Aktionen mit denselben Parametern vorschlagen und zum Vergleich der Algorithmen jede Gruppe bitten, ihren Algorithmus zum Vergleich an die Tafel zu schreiben.	
2. Testen der	Dieser zweite Schritt entspricht der Übersetzung des in der Sprache Scratch geschriebenen	5.2.
Algorithmen mit der Software (25')	Algorithmus mit der Software Lego Spike. Die Lehrkraft bereitet wiederum die Computer für die beiden Gruppen und die Projekte vor (Anhänge 5 und 6). Sobald die Paare denken, dass	3.3.
bottware (20)	sie fertig sind, informieren sie die Lehrkraft.	3.1.
		6.1.
		(6.2.)
		(6.3.)
3. Bündelung von Aktivität 2 (25')	Der dritte Schritt ist die Zusammenführung der Ergebnisse: - Wenn die Roboter der Duos genau dasselbe tun wie der Roboter des Lehrers, schlägt der	(5.2.)









	Lehrer vor, dass die Schüler die Programme der Duos mit denen des Lehrers vergleichen.	3.3.
	Die Lehrkraft hebt besonders die Verwendung der Bedingung hervor und bittet die Schüler, zu versuchen, sie zu definieren: "Was hier verwendet wurde, ist die Bedingung, wofür wird	3.1.
	sie verwendet? Wie könnte sie definiert werden? Der Lehrer leitet die Schüler an, es in	6.1.
	Worte zu fassen: Die Bedingung ermöglicht es, dafür zu sorgen, dass der Roboter auf eine bestimmte Weise reagiert oder nicht, wenn ein oder mehrere Kriterien erfüllt sind. Der	(6.2.)
	Lehrer wird auch die Verwendung der Schleife hervorheben, insbesondere die Schleife, die sich unendlich wiederholt.	(6.3.)
	 Wenn einer oder beide Roboter nicht dasselbe tun wie der Roboter des Lehrers, werden die Schüler aufgefordert, über die Ursache des Problems nachzudenken, indem sie zunächst den geschriebenen Text und dann das Scratch-Programm bearbeiten (wenn der geschriebene Text korrekt ist). Wenn es ein Problem mit dem schriftlichen Algorithmus gibt, fordert der Lehrer die Schüler auf, zu versuchen, das Fehlende zu identifizieren oder zu verstehen, was nicht deutlich genug gemacht wurde. Falls erforderlich, weist der Lehrer auf den Moment hin, in dem die Bedingung verwendet wird, und bittet die Schüler, diese Passage mit Worten zu versehen. Dann schlägt er vor, dass die Schüler es noch einmal mit Scratch versuchen. Wenn es das Programm der Lego-Software ist, das Probleme verursacht, schlägt der Lehrer vor, das Programm mit den Aktionen seines Roboters zu vergleichen. Falls dies noch nicht geschehen ist, weist der Kursleiter auf den Moment hin, in dem die Bedingung benutzt wird, und bittet die Teilnehmer, diese Passage mit Worten zu beschreiben: "Was habt ihr benutzt, als der Roboter mit dem Gesicht zur Wand kam? 	4.2.
	Es ist auch möglich, dass die Schüler nicht herausgefunden haben, wie sie das Programm unbegrenzt wiederholen können. In diesem Fall lässt der Lehrer die Schüler darüber nachdenken, was der Roboter macht: "Was macht der Roboter, nachdem er gegen die Wand gefahren ist? "Erläutert in euren eigenen Worten, was von dem Roboter verlangt wurde, damit er es wieder tut."	
Idee für eine Weiterführung	Die Schüler können aufgefordert werden, ihr eigenes Programm mit Schleife(n) und Bedingung(en) zu erstellen. Sie können dies tun, indem sie ihren Roboter erstellen und sich	2.4.
Westersam ang	ein Ziel setzen, das sie erreichen wollen.	3.3.











Diese Art von Aktivität wird es den Schülern (wahrscheinlich) ermöglichen, neue Befehle zu verwenden und Schleifen und Bedingungen auf komplexere Art und Weise zu nutzen (Verwendung von Operatoren, Verwendung von "ifthenelse", Verwendung mehrerer Schleifen in einer Bedingung, Verwendung einer Bedingung in einer Bedingung, usw.).	3.1.	
--	------	--









Bewertung

Kompetenzen/ PIA- Ziele	Aktivitäten zur Bewertung	Bewertungskriterien
5.1.	Schreiben des Algorithmus auf der Grundlage der Beobachtung des Verhaltens eines Roboters (Anhang 7)	 Der Algorithmus ist in klar definierte Schritte unterteilt. Die Reihenfolge der Schritte/Aktionen im Algorithmus entspricht der Reihenfolge der Bewegungen des Roboters. Das verwendete Vokabular ist präzise und korrekt.
5.2.	Übersetzung des in Scratch geschriebenen Algorithmus Der Schüler hat die Möglichkeit, seinen Roboter so oft zu testen, wie er möchte.	Das Programm entspricht dem von der Schülerin/dem Schüler geschriebenen Algorithmus
3.1.	Dieser Schritt findet gleichzeitig mit den Aktivitäten statt, die zur Bewertung der Kompetenzen 5.1 und 5.2 vorgeschlagen werden: Der Lehrer bittet die Schüler, einen Algorithmus und ein Programm zu erstellen, das die Schleife verwendet, um ihr Programm zu reduzieren.	Verwendung der Schleife im geschriebenen Algorithmus Verwendung der Schleife im Scratch-Programm
3.3.	Dieser Schritt findet zeitgleich mit den Aktivitäten statt, die zur Bewertung der Kompetenzen 5.1 und 5.2 vorgeschlagen werden:	Verwendung der Bedingung im geschriebenen Algorithmus













Das zu übersetzende Programm enthält eine Bedingung.	Verwendung der Bedingung im Scratch-Programm
Wenn keine spezifischen Blöcke vorbereitet werden, teilt der Lehrer den Schülern mit, dass ihr Programm eine Bedingung enthalten muss.	









Bewertungskriterien

Kriterien	Sie ist erfolgreich, wenn	Ergebnis
1) Der Algorithmus ist in klar abgegrenzte Schritte aufgeteilt.	Ich trenne jeden Schritt gut, indem ich zur Zeile gehe und einen neuen Satz beginne.	
2) Die Reihenfolge der Schritte im Algorithmus berücksichtigt die Reihenfolge der Bewegungen des Roboters.	Ich schreibe die Schritte der Reihe nach auf.	
3) Das verwendete Vokabular ist präzise und korrekt.	Ich verwende präzise und unmissverständliche Begriffe.	





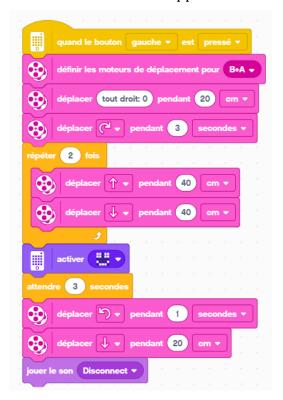




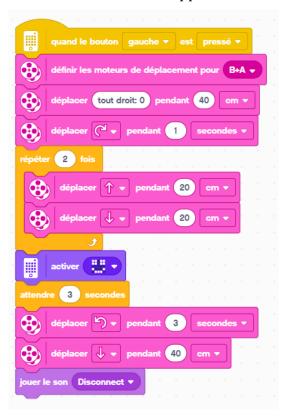


♦ Anhang 1: Verwendete Algorithmen

Aktivität 1 - Gruppe 1



Aktivität 1 - Gruppe 2







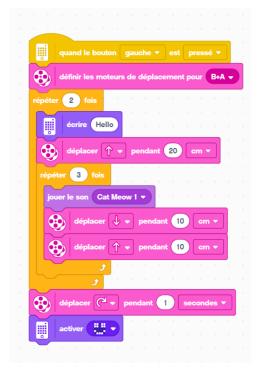






♦ Anhang 2: Erweiterung

Gruppe 1



Gruppe 2

```
2 fois
déplacer ↑ ▼ pendant 10 cm ▼
  déplacer (2 → pendant 3 secondes
```





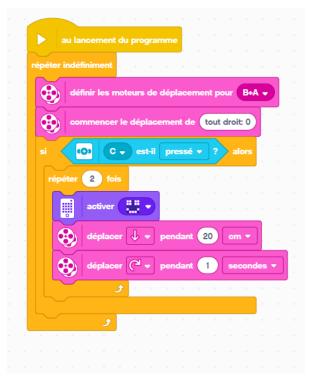






♦ Anhang 3: In Aktivität 2 verwendete Algorithmen

<u> Aktivität 2 - Gruppe 1</u>



Aktivität 2 - Gruppe 2

```
er le déplacement de tout droit: 0
répéter 2 fois
                       pendant 3
```











♦ Anhang 4

Was macht unser Roboter?

Schreiben wir unseren Algorithmus (die Schritte, die der Roboter durchläuft)



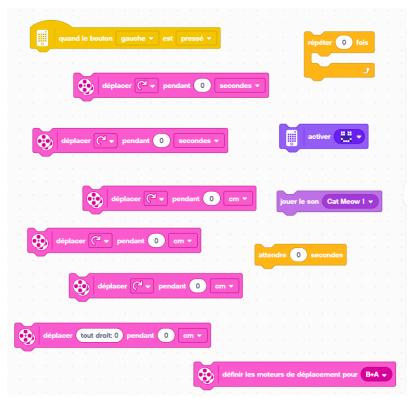






♦ Anhang 5: Blöcke zur Vorbereitung für Anfänger

Tätigkeit 1



Tätigkeit 2

```
C → est-il pressé ▼
                       nt de (tout droit: 0
                                 er 🔁 ▼ pendant 0 secondes ▼
```





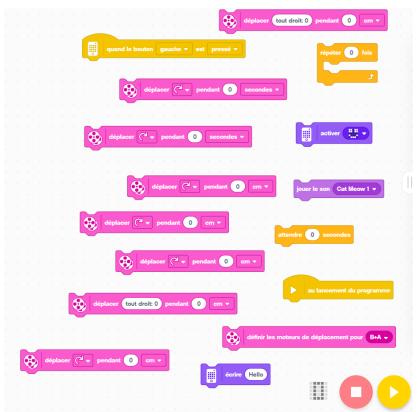






♦ Anhang 6: Bausteine zur Vorbereitung auf die weiterführenden

Tätigkeit 1



Tätigkeit 2

```
définir les moteurs de déplacement pour B-A v quand pressé v

au lancement du programme

répéter ① fois

répéter indéfiniment

si alors

con est-tilelle plus proche que v 15 % v 2

commencer le déplacement de tout droit ②

déplacer v pendant ② cm v déplacer v pendant ② secondes v

déplacer v pendant ② secondes v

déplacer v pendant ② secondes v

déplacer v pendant ② secondes v
```











♦ Anhang 7

