

Robotik

Kategorie:	Programmieren, 11., 12. und 13. Klasse
Thema: (Unterthema)	Programmierung von Mindstorm EV3-Robotern
Autor/Impressum	Michael Ulex, Freie Hochschule Stuttgart
Geeignet für	Klassenstufe 11, 12 und 13
Stundenzahl	Min. 5 Doppelstunden, Dauer nach oben offen.
Pädagogische Ziele	Verständnis des Aufbau eines Programms; Beherrschen eines Computers, Auseinandersetzung mit begrenzten Möglichkeiten.
Kurzbeschreibung	Programmieren der Roboter mit Piktogrammen (erweitert mit Bricx, einer c-ähnlichen Sprache), dabei Lösen von Aufgaben, die Kreativität, Durchhaltevermögen und Teamfähigkeit verlangen.
Nötige Vorbereitungen	Hardware (je einen Lego Mindstorm EV3-Roboter und einen Laptop für 2 Jugendliche), Experimentierraum mit Freifläche auf dem Boden, Programmiersoftware.
Hilfsmittel	Hindernisse, Kreppband und andere Kleinmaterialien.
Involvierbare Fächer	Mathematik, Physik, Biologie

Ausführliche Beschreibung

Die Jugendlichen lernen die Grundzüge des Programmierens anhand von LEGO Mindstorm Robotern kennen. Sie programmieren Maschinen (ohne Sensoren) und Roboter (die auf die Sensoren reagieren). Neben festgelegten Aufgaben wird ein selbst gewähltes Projekt durchgeführt. Dabei kommt es nicht nur auf das Programmieren, sondern auch auf das Kommunizieren an, sowohl über die Planung des Projekts als auch rückblickend über die Ergebnisse an.

Version vom September 2019

1 Vorbereitung des Unterrichtsmaterials

Materialien

Einige Tage vor Beginn sollten die **Akkus der Programmierbausteine** vollständig geladen werden. Je nach Aktivität muss der Akku nach ca. drei Tagen wieder geladen werden. Dafür muss der Roboter so zusammengebaut sein, dass die Buchse für das Ladegerät zugänglich ist. Alternativ kann auch mit **Einweg-Batterien** gearbeitet werden, diese halten deutlich länger, in der Regel die ganze Epoche. Batterien sind ökologisch fraglich, das könnte peinliche Nachfragen von SchülerInnen erzeugen.

Ebenfalls vorher sollten die **Bausteine in den Baukästen durchgezählt** werden. Alternativ kann das auch in der ersten Stunde durch die Jugendlichen erfolgen, dafür wird eine Kopie der Bauteilliste benötigt. Sinnvoll ist es, dafür eine Kopie von der Pappe mit der Sortiervorrichtung zu verwenden, dann liegen die Teile gleich zu Beginn alle in den richtigen Fächern. An jedes Fach wird ein Haken gemacht (wenn die Teile stimmen) oder mit +n (n Teile zu viel) oder -n (n Teile zu wenig) dokumentiert, dass die Zahl nicht stimmte. Auf die Kopie kommt die Nummer des Baukastens und die Namen der beiden Jugendlichen. Am Ende der Epoche werden die Teile wiederum gezählt und mit einem Stift in einer anderen Farbe die Haken gemacht bzw. Fehlmengen notiert.

Weitere benötigte Materialien sind **kleine Hindernisse** wie z. B. Pylone (Spielzeugladen) zum Umfahren sowie **Tesakrepp**, möglichst weiß.

Falls der Raum zu wenig Steckdosen hat, so werden **Mehrfachsteckdosen** benötigt. Sofern als Rechner **Laptops** verwendet werden, sollten deren **Akkus** ebenfalls geladen sein.

Software

Zunächst ist eine Entscheidung zu treffen, mit welcher Software programmiert werden soll. Wer als LehrerIn das erste Mal mit den EV3-Robotern arbeitet, sollte vorab einige Stunden investieren, um die Befehle sicher bedienen zu können. Folgende Programme stehen zur Wahl:

- [Lego EV3 Software](#) ist eine von Lego zur Verfügung gestellte Programmiermöglichkeit, bei der mithilfe einer graphischen Oberfläche die wesentlichen Funktionen ohne große Vorkenntnisse verwendet werden können. Sie läuft auf Windows ab Vista sowie auf Mac ab OS 10.6.
- Wer lieber textbasiert arbeitet, kann mit [EV3 Basic](#) arbeiten. Das ist eine Variante von Simple Basic by Microsoft und läuft auf Windows und Mac.

- Eine weitere Möglichkeit, textbasiert zu arbeiten, ist [EV3 Python](#). Vorteil des Programms ist die Möglichkeit auf **Linux** zu arbeiten – es gibt inzwischen eine Reihe von Schulen, die mit ihrem Schülernetzwerk nur noch mit Linux arbeiten. Für dieses Programm muss das Betriebssystem der Roboter durch ein alternatives auf einer SD-Karte ersetzt werden. Das ursprüngliche Betriebssystem wird dabei nicht angetastet. Sobald die SD-Karte entfernt ist, wird das originale Betriebssystem wieder gebootet. Weitere Informationen dazu sind unter der oben angegebenen Adresse zu finden.

Neben der Betriebssoftware auf den Robotern muss die dazugehörige Software auf den Rechnern installiert werden. Es ist empfehlenswert, vor jeder Epoche zu überprüfen, ob auf allen Rechnern die gleiche Software installiert ist.

Unterrichtsraum

Als Raum kann jeder beliebige Klassenraum verwendet werden, genügend Steckdosen für die Rechner können über Mehrfachsteckdosen zur Verfügung gestellt werden. Es muss **genügend freie Fläche** auf dem Boden vorhanden sein (ca. 2 m² pro Gruppe). Falls der Boden nicht einfarbig ist (**Parkett** ist schwierig), so kann es sinnvoll sein, Papier oder Pappe auszulegen. Ebenso spielt die **Beleuchtung** eine Rolle, sämtliche Aufgaben mit dem Beleuchtungssensor sind sehr lichtempfindlich, d. h. bei Sonnenschein ändern sich die Parameter.

Der Raum sollte vor der Unterrichtsstunde **gefegt** oder, besser noch, gewischt werden.

2 Die erste Unterrichtsstunde

In der 12. Klasse kann man die **Zusammensetzung der Gruppen** den Jugendlichen überlassen. Sinnvoll ist es, wenn die Begabung gleich verteilt ist, damit beide gleich viel lernen. Ein grundsätzliches Problem sind aufgrund von Krankheit fehlende Jugendliche. Dann muss man ggf. Gruppen zwischenzeitlich neu zusammensetzen.

Falls genügend Zeit vorhanden ist, beginnen die Gruppen mit dem Zählen der Teile und **bauen** anschließend **den Roboter** anhand der Anleitung im Heft (noch ohne Sensoren) **zusammen**. Diesen Teil kann man aus Zeitgründen weglassen, pädagogisch ist es aber sinnvoll, da sich die Jugendlichen durch diese erste Beschäftigung dem Gerät langsam nähern können.

Schnelle Jugendliche können nach dem Zusammenbau sofort die **erste Aufgabe** lösen: Lasse den Roboter über eine Strecke von ca. 3 m geradeaus fahren und durch ein Tor fahren. Die Lehrkraft sollte sicherstellen, dass zum Ende der ersten Stunde alle Roboter betriebsbereit sind. Dabei können schnelle Jugendliche oder die Lehrkraft selber den Langsamem helfen.

3 Programmieren ohne Sensoren

Am **zweiten Tag** führt die Lehrkraft zunächst die wichtigsten Befehle des Programms ein (Motor drehen, Sensor auslesen, Variablen definieren). Benötigte Kenntnisse sind die Befehle für die Drehgeschwindigkeit der Räder und die Zeit, wie lange diese drehen sollen. Als Aufgabe bekommen die Jugendlichen einen Parcours mit etwa 5 Hindernissen, mit exakt gleichen Abständen (ca. 50 cm). Da die Hindernisse oft weggeschoben werden, sollten die Standorte mit Tesakrepp markiert werden. Der Roboter soll im **Slalom** um diese Hindernisse herum fahren. Begonnen wird die Fahrt an einem Tesakreppstreifen als Startlinie und beendet mit einer Fahrt durch ein Tor (ca. doppelt so breit wie ein Roboter).

In der Regel gibt es nur einen kleinen Teil der Gruppen, die die Aufgabe vollständig lösen. Das ist nicht schlimm, weil auch Teilerfolge zählen. Schon das Umfahren des zweiten Hindernisses beweist, dass die Gruppe das Prinzip beherrscht. Für die weiteren Hindernisse muss genau die selbe Programmierung wie zu Beginn erfolgen. Daraus folgt die **gewünschte erste Erkenntnis: Schleifen** erleichtern die Programmierung.

Je weiter der Roboter fährt, desto ungenauer wird die Fahrt. Daraus ergibt sich die **gewünschte zweite Erkenntnis: Ohne Sensoren geht es irgendwann nicht mehr**, da die Fahrtrouten nur begrenzt reproduzierbar sind.

4 Programmieren mit Sensor

Am **dritten (und ggf. vierten und fünften) Tag** werden als neues Programmiererelement **Schleifen** eingeführt. Im Anschluss oder gleichzeitig werden **Sensoren** neu eingeführt. Je nach Lerngruppe bekommen entweder alle die gleiche Aufgabe (also alle z. B. den Entfernungssensor) oder unterschiedliche Aufgaben mit unterschiedlichen Sensoren. Möglich wäre es auch, alle den Parcours vom Vortag durchfahren zu lassen, nur mit unterschiedlichen Sensoren:

- **Ultraschallsensor** (Hindernisse müssen hoch genug sein)
- **Berührungssensor** (Hindernisse müssen fest verankert sein)
- **Lichtsensor** (mit Kreppband Kreise um die Hindernisse kleben)
- **Schallsensor** (Programmierer klatscht, wenn der Roboter nah am Hindernis ist oder eine Schallquelle am Hindernis wird als zu laut wahrgenommen, wenn der Roboter zu nah kommt – ist sehr schwer exakt zu programmieren)

Dritte gewünschte Erkenntnis: Sensoren helfen, aber nur, wenn die richtige Aktion daraus erfolgt.

Je nach zur Verfügung stehenden Zeit können noch weitere Aufgaben gestellt werden, z. B. den **Linienverfolger** (entlang einer hellen Linie fahren), das **Gatter** (eine physische oder optische Barriere nicht überschreiten) oder die **Motte** (fährt dahin, wo es am Hellsten ist).

5 Projekt

Nachdem die Jugendlichen nun die Grundzüge des Programms und die Sensoren kennen, dürfen sie ein Projekt auswählen, welches in einer festgelegten Zahl von Tagen programmiert werden muss. Folgende Projekte haben sich als sinnvoll erwiesen, wobei der Schwierigkeitsgrad unterschiedlich ist. Selbstverständlich ist es möglich, noch andere Projekte zu erfinden, auch den Jugendlichen sollte diese Möglichkeit angeboten werden. Bei von Jugendlichen selbst erfundenen Projekten bedarf es einiger Erfahrung der Lehrkraft um zu beurteilen, wie realistisch ein Projekt ist.

Grundsätzlich sollte jedes Projekt mehrstufig aufgebaut sein. Die erste Stufe muss erreicht werden, weitere Stufen sind dann fakultativ. Es besteht bei Jugendlichen immer die Gefahr, dass sie sich in ein Teilprojekt verrennen und letztlich gar nichts vorstellen können. Das ist unbedingt zu vermeiden.

Das scheue Reh

Das scheue Reh steht auf der Wiese und grast. Ertönt ein Geräusch, so rennt es fort (möglichst vom Geräusch weg), aber nicht blind, sondern achtet darauf, wo es hin rennt. Ist etwas im Weg, so ändert es die Richtung. Ist es weit genug gelaufen, so bleibt es stehen und grast wieder. Für dieses Projekt wird viel Platz benötigt.

Der Sammler

Der Sammler registriert, wo die zu sammelnden Teile (z. B. Holzklötzchen) liegen, sammelt diese (z. B. einzeln) ein bringt sie zu einer Sammelstelle. Optional können die Teilchen nach Farbe, Gewicht, Größe o. ä. sortiert werden.

Das Mannequin

Wie es sich für ein Mannequin gehört, sieht es gut aus. Es läuft über den Laufsteg, dreht am Ende schwungvoll um und läuft zurück. Optional kann es aufpassen, dass es nicht seitwärts herunterfällt, den Hintern schwenken, zusätzliche Drehungen einbauen oder mehrfach hin- und herlaufen.

Der Kampfroboter

Diese Aufgabe ist sehr beliebt, aber auch sehr schwer und sollte nur von begabten Jugendlichen versucht werden. Sie kann auch von zwei Gruppen bearbeitet werden, die sich dann zum Abschluss messen. Der Kampfroboter sucht den Gegner und schlägt auf ihn ein. Die Aufgabenstellung muss vorher klar definiert sein, z. B. den Gegner zu Fall bringen, ihn rammen oder ihn einsperren. Besser ist es, wenn das Ziel leichter erreichbar ist, z. B. einen Ball, der auf dem Roboter liegt, herunterzuwerfen.

Der Orientierungsläufer

Der Orientierungsläufer hat es eilig. Deshalb muss er schnell laufen, gleichzeitig aber mitbekommen, wo Hindernisse und wo Ziele sind. Ziele muss er ansteuern, Hindernissen ausweichen.

6 Präsentationen

Ein wichtiger Teil der Programmierung ist die Kommunikation, da gerade geniale Programmierer oft nicht in der Lage sind, ihre Erkenntnisse zu vermitteln. Das ist aber im Beruf zwingend notwendig. Deshalb hat zum Ende einer jeden Unterrichtsstunde eine Gruppe die Aufgabe, ihre Erfahrungen in ca. 5-10 Minuten zu dokumentieren. Zur Dokumentation gehört:

- Aufgabenstellung
- Lösungsansätze
- Programm vorstellen
- Schwierigkeiten mit Lösungen
- Rückblick über Erfolge und Misserfolge.

Der Vortrag wird von den MitschülerInnen reflektiert.

7 Ablauf einer Unterrichtsstunde

Zu Beginn der Stunde wird etwas Neues vorgestellt und Fragen dazu beantwortet. Anschließend beginnt das Programmieren. Falls sinnvoll, stellen alle Gruppen ihr Ergebnis vor (z. B. bei der ersten Aufgabe), dann wird aufgeräumt. Zum Schluss präsentiert eine Gruppe ihren Prozess.

Falls die Epoche an aufeinander folgenden Tagen erfolgt, kann auch dreigliedriger Unterricht erfolgen, dann würde das Neue zum Unterrichtsende vorgestellt.

8 Rückblick

Die Erfahrung zeigt, dass ein Rückblick nach der Epoche jedes Mal erstaunlich viele neue Rückmeldungen bringt. Als Grund ist neben der Individualität auch die fortlaufende Entwicklung der privat genutzten elektronischen Geräte, die Jahr für Jahr neue Fähigkeiten erfordert.