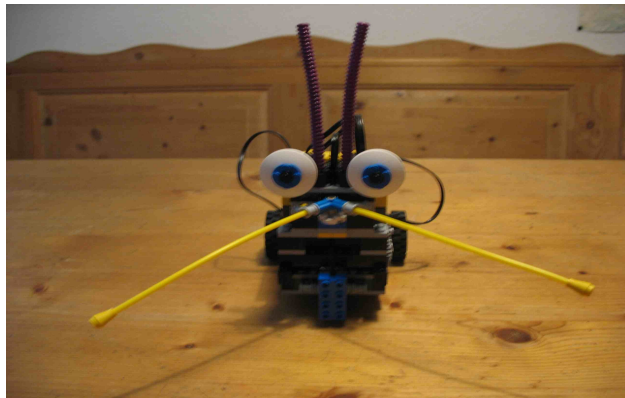


Albert-Ludwigs-Universität Freiburg  
Fakultät für Angewandte Wissenschaften  
Sommersemester 2007  
Thomas Schaller  
Fachdidaktik Informatik

**Unterrichtsgang:  
Grundlegende Elemente von Algorithmen  
am Beispiel von LEGO Mindstorms**



22.06.2007  
Christoph Griesser

Englisch, Informatik  
(Staatsexamen, 4. Fachsemester)  
[christoph.griesser@gmail.com](mailto:christoph.griesser@gmail.com)

# Inhaltsverzeichnis

1. Einleitung .....	2
2. Einführung in LEGO Mindstorms.....	3
2.1 Zielformulierung.....	3
2.2 Sozialform.....	4
2.3 Nötige Vorbereitungen .....	4
2.3 Durchführung.....	4
3. Anweisungen & Algorithmen.....	5
3.1 Zielformulierung.....	5
3.2 Sozialform.....	6
3.3 Durchführung.....	6
4. Flussdiagramme.....	7
4.1 Zielformulierung.....	7
4.2 Sozialform.....	7
4.3 Durchführung.....	7
5. Variablen, Entscheidungen und Wiederholungen.....	11
5.1 Zielformulierung.....	11
5.2 Sozialform.....	11
5.3 Durchführung.....	11
6. Abschlussprojekt.....	13
6.1 Zielformulierung.....	13
6.2 Sozialform.....	13
6.3 Durchführung.....	13

# 1. Einleitung

„Informatik besitzt einen wesentlichen Anteil am hohen Entwicklungsstandard unserer technisierten Welt. Informatikunterricht konzentriert sich auf die grundlegenden informatischen Prinzipien, Konzepte, Arbeitsweisen und Methoden. Er liefert damit einen wichtigen Beitrag zur Allgemeinbildung.“<sup>1</sup>

Dieses Zitat aus dem aktuellen Bildungsplan macht die Zielsetzung des Informatikunterrichts deutlich. Anstelle der Vermittlung von „Anwendungswissen“ hat der Informatikunterricht seine Aufgabe in der Vermittlung von grundlegenden Konzepten und Ideen mit denen die Schülerinnen und Schüler vertraut gemacht werden sollen. Einige solcher fundamentalen Konzepte, namentlich Anweisungen, Anweisungsfolgen, Entscheidungen und Wiederholungen, sollen in dieser Arbeit mit der Einführung in die Programmierung umgesetzt werden.

Zur Umsetzung dieser Aufgabe habe ich mich zur Arbeit mit Robotern von LEGO Mindstorms<sup>2</sup> entschieden. Aufgrund ihrer vielfältigen Möglichkeiten eignen sie sich gut für den Informatikunterricht. Der Unterrichtsgang ist in mehrere Phasen unterteilt, die aufeinander aufbauen. Da der Mindstorms-Bausatz in der Anschaffung teuer ist, kann davon ausgegangen werden, dass die meisten Schüler LEGO Mindstorms nicht, oder zumindest nicht aus eigener Erfahrung kennen. Aufgrund der zu erwartenden begrenzten Anzahl von Bausätzen, werden die meisten Aufgaben in verschiedenen Formen von Gruppenarbeit durchzuführen sein. Die ideale Gruppenstärke liegt zwischen 2 und 3 Schülern pro Gruppe. Am Ende der Unterrichtseinheit sollen die Schüler ihren erstellten Roboter einem Publikum (große Pause, Elternabend, Tag der offenen Tür, etc.) vorstellen können.

## 2. Einführung in LEGO Mindstorms

### 2.1 Zielformulierung

Ziel dieser Phase soll sein, dass bei den Schülern die Begeisterung für das Unterrichtsthema geweckt wird. Des Weiteren sollen die Schüler nach dieser Phase den RCX-Baustein bedienen können und um die Polarität der Steckanschlüsse wissen. Die Schüler kennen danach auch die Funktionsweise der unterschiedlichen Sensoren (Licht-, bzw. Berührungssensor).

---

1 Auszug aus dem Bildungsplan für Gymnasien.  
[http://www.bildung-staerkt-menschen.de/service/downloads/Bildungsstandards/Gym/Gym\\_Inf\\_wb\\_bs.pdf](http://www.bildung-staerkt-menschen.de/service/downloads/Bildungsstandards/Gym/Gym_Inf_wb_bs.pdf)  
S.438. Stand: 07.06.2007

2 Der verwendete Bausatz trägt die Bausetnummer 9794.

## **2.2 Sozialform**

Das Zusammenbauen des Roboters kann in Gruppen erfolgen. Es wäre denkbar, dass ein Schüler für das Aussuchen und Herauslegen der korrekten Teile für die ganze Gruppe zuständig ist. Antrieb, Licht- und Stoßsensor können jeweils von einer Person in der Gruppe zusammengebaut werden, wobei man die entsprechenden Seiten aus der Anleitung kopieren könnte.

## **2.3 Nötige Vorbereitungen**

Der Lehrer muss sich vor der Unterrichtseinheit mit der Bedienung der Software, dem Erstellen und Übertragen der Programme vertraut machen. Nach meiner Erfahrung müssen dafür ca. 2 Stunden veranschlagt werden. Zum Ausprobieren der Roboter muss auf den Programmplätzen 1-4 des RCX, dem programmierbaren Baustein, jeweils ein kleines Demo-Programm installiert werden.

Programmplatz 1: Roboter fährt 3 Sekunden gerade aus, dreht sich dann nach links und fährt nochmals 3 Sekunden bevor er dann hält.

Programmplatz 2: Wenn Tastsensor1 gedrückt wird, macht er eine Kurve nach rechts, wenn Tastsensor 2 gedrückt wird, macht er Kurve nach links

Programmplatz 3: Der Lichtsensor ist aktiv. Roboter hält an, macht eine kleine Rechtsdrehung und fährt weiter, wenn eine schwarze Linie findet.

## **2.3 Durchführung**

Zu Beginn der Unterrichtseinheit bringt der Lehrer einen fertigen Roboter mit in den Unterricht. Anhand eines Programms (z.B. einem Roboter, der einer Linie folgen kann), demonstriert er die Möglichkeiten des Einsatzes. Die Schüler sollen nun selbst einen fahrbaren Roboter zusammenbauen.

Der LEGO-Roboter kann laut Anleitung zu verschiedenen Modellen zusammengebaut werden. Grundsätzlich sind der Kreativität hier keine Schranken gesetzt und ich denke es würde den Lernprozess behindern, wenn den Schülern der Bau eines bestimmten Modells vorgeschrieben wäre. Der Lehrer steht den Schülern beim Aufbau beratend zur Seite. Es sollte nur darauf geachtet werden, dass Licht- und Stoßsensoren bei allen Gruppen an die gleichen Stellen angebracht werden (Lichtsensor vorne unten, Stoßsensoren vorne links und rechts). Ich habe mich bei meinem Roboter für Grundaufbau für Fahrzeuge entschieden, wie er von Seite 16-21 in der beiliegenden Anleitung beschrieben ist. Es können dann verschiedene Antriebsarten, bzw. Arten von Rädern gewählt

werden. Durch Ausprobieren kam ich zu dem Ergebnis, dass sich Kettenantrieb und Antrieb auf vier Rädern nicht besonders gut eignen, wenn der Roboter eine Drehung auf der Stelle ausführen soll. Deswegen ist die Zwei-Räder-Variante, wie auf Seite 24 beschrieben, am besten geeignet. Die Anleitung zum Bau des Lichtsensors findet sich auf Seite 28, die „Fühler“ mit den Tastsensoren habe ich laut Anhang auf S. 87 gebaut. Hier ist ein wenig Fingerspitzengefühl nötig, die Fühler müssen so angebracht werden, dass die Tastsensoren ständig berührt, aber nicht gedrückt werden. Bei zu loser Befestigung „schwanken“ die Fühler zu sehr, so dass die Sensoren bei Fahrt gegen ein Hindernis nicht betätigt werden.

Wenn die Roboter zusammengebaut sind, können die Schüler den Roboter testen. Ziel ist hierbei, das Kennenlernen der Funktionen des RCX (Tasten *On/Off*, *Run* und *Prgm.*), sowie die Funktionsweise der Sensoren.

Aufgabe 1: Stellt euren Roboter auf den Boden des Klassenzimmers und drückt die rote *On/Off*-Taste. Drückt nun solange die graue *Prgm*-Taste, bis rechts eine 1 erscheint. Drückt nun die grüne *Run*-Taste und beschreibt, was passiert und was ihr auf dem Display seht. Wie könnt ihr den Roboter während der Fahrt anhalten? Wie könnt ihr erreichen, dass der Roboter rückwärts fährt ohne den Antrieb umzubauen?

Aufgabe 2: Wechselt nun zum Programm 2 des Roboters und drückt die grüne *Run*-Taste. Wie kann der Roboter um ein Hindernis fahren? Probiert es aus.

Aufgabe 3: Nehmt euch ein DIN A0 Blatt und malt mit Filzstift eine dicke schwarze Linie an den Rand, setzt den Roboter ans andere Ende. Wechselt zum Programm3 des Roboters und drückt die grüne *Run*-Taste und lasst euren Roboter daraufzufahren. Notiert eure Beobachtung. Nehmt ein neues Blatt und einen Stift. Wie kann euer Roboter darauf eine langgezogene Rechtskurve fahren?

## 3. Anweisungen & Algorithmen

### 3.1 Zielformulierung

Ziel dieser zweiten Phase soll sein, dass die Schüler die Arbeitsumgebung der graphischen RIS-Software kennen lernen und Programme auf den Roboter übertragen können. Sie verstehen, dass das Programm sequenziell abläuft und kennen den Algorithmenbegriff. Sie wissen wie sich das Ändern von Parametern in einem Demo-Programm auf das Verhalten des Roboters auswirkt, wissen was eine Anweisung ist und können eigene erstellte Anweisungsfolgen, bzw. Unterprogramme zur Problemlösung einsetzen.

### **3.2 Sozialform**

Zum Einstieg würde ich die lehrerzentrierte Unterrichtsform wählen, bei der der Lehrer das Anlegen des Benutzerkontos vorführt und bei Interesse das anschließende Video zeigt. Einstellungen in der Software können ebenfalls am Beamer vorzeigt werden. Anschließend kann jeder Schüler das Gezeigte am eigenen PC durchführen und die Aufgaben selbstständig bearbeiten. Das erstellte Programm kann zum Test von jedem Schüler auf den Gruppenroboter übertragen werden. Die anschließende Internet-Recherche soll auch von jedem Schüler selbstständig durchgeführt werden.

### **3.3 Durchführung**

Nachdem sich jeder Schüler ein Benutzerkonto angelegt hat, gelangt man mit einem Klick auf „Programm“ und dem Ignorieren der Warnung zur „Freistil“-Programmierung. Dort kann unter „Einstellungen“ zunächst das praktische Randscrollen aktiviert werden. Nun sollte erklärt werden, dass sich im linken Teil unter „Kleine Blöcke“ die Anweisungen befinden. Die Problematik, dass sich unter „große Blöcke“ bereits fertige Routinen befinden, kann umgangen werden, indem man erklärt, dass diese nur bei den bestimmten Robotertypen Sinn machen und nicht verwendet werden sollten. Wie ein Programm auf den RCX übertragen wird, kann einmal z. B. mit der Anweisung „Kurzton“ demonstriert werden.

Aufgabe 4: Lade zunächst das Programm „Run1“. Was könnte dieses Programm tun? Notiere deine Überlegungen zuerst, bevor du sie überprüfst, indem du das Programm auf den Roboter überträgst. Lasse den Roboter anschließend für 5 Sekunden geradeaus fahren und den Ton1 ausgeben.

Aufgabe 5: Lade nun „Run2“. Was tut dieses Programm? Notiere deine Überlegungen zuerst, dann überprüfe deine Vermutung ebenfalls. Lasse dann deinen Roboter mit voller Leistung vorwärts, mit halber Leistung rückwärts und nochmals mit voller Leistung vorwärts fahren.

Anhand von Aufgabe 5 kann der Sinn von Unterprogrammen in einem kurzen Lehrervortrag erklärt und aufgezeigt werden, wie man anhand der „Eigenen Blöcke“ solche erstellt. Eine optimale Lösung wäre, wenn die Schüler die Anweisungsfolge „Geradeaus“ als Block erkennen und nochmals verwenden würden.

Aufgabe 6: Eine Kurve wird vom Roboter ausgeführt, indem sich das äußere Rad schneller dreht als das innere Rad. Lasse deinen Roboter zuerst piepsen und dann drei Sekunden geradeaus fahren und anschließend eine 90-Grad Rechtskurve machen. Danach soll der Roboter nochmals piepsen und für drei Sekunden geradeaus fahren, bevor er hält.

Zusatzaufgabe: Schreibe ein Programm, dass den Roboter zuerst wie beschrieben geradeausfahren, dann möglichst auf der Stelle wenden und wieder zum Ausgangsort zurückkehren lässt.

Zum Abschluss dieser Phase soll der Algorithmenbegriff eingeführt werden. Die Schüler bekommen Arbeitsblätter ausgeteilt, auf denen verschiedene richtige und falsche Algorithmen abgebildet sind (genaue, ungenaue Rezepte, Anleitungen usw.). Sie sollen den Begriff „Algorithmus“ im Internet suchen und die Abbildungen jeweils danach klassifizieren. Im Lehrer-Schüler-Gespräch werden die Gründe für die richtigen Lösungen erarbeitet und eine Definition wird an der Tafel festgehalten.

## **4. Flussdiagramme**

### **4.1 Zielformulierung**

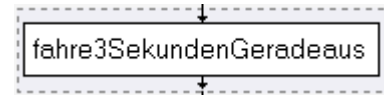
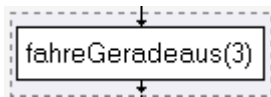
Die Schüler sollen in dieser theoretischen Phase das Lesen und Erstellen von Flussdiagrammen lernen. Durch die Flussdiagramme soll vom der intuitiven Bedienung der RIS-Software abstrahiert, und Anweisungen, Entscheidungen und Wiederholungen formalisiert werden.

### **4.2 Sozialform**

Lehrervortrag, Lehrer-Schüler-Gespräch, Einzelarbeit

### **4.3 Durchführung**

Zum Einstieg in diese Phase sollten das Konzept der Anweisungen, der Anweisungsfolgen und der Algorithmenbegriff nochmals wiederholt werden. Aufgrund der präzisen Ausdrucksweise bei Algorithmen kann die Notwendigkeit einer einheitlichen Schreibweise aufgezeigt werden. Als Beispiel kann Aufgabe6 verwendet werden. Der Lehrer sollte die Symbole, die im Flussdiagramm vorkommen erklären (noch ohne Schleifen und Entscheidungen). Bei Anweisungen im Pseudocode ist es wichtig zu erklären, dass die Argumente in Klammer hinter der Anweisung stehen, z.B. also :

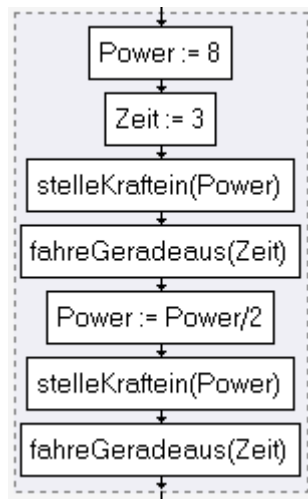


anstelle von

Nun kann im Lehrer-Schüler-Gespräch Aufgabe6 in ein Flussdiagramm umgesetzt werden.

Danach sollten Variablen eingeführt werden. Eine Variable kann den Schülern beispielsweise als „Zahlencontainer“ vorgestellt werden, der am Anfang leer ist und zu dem man etwas addieren, subtrahieren, multiplizieren oder dividieren kann, so dass der „Container“ hinterher diesen Wert besitzt. Eine Variable wird somit beschränkt als „Zähler“ eingeführt, was in diesem Fall auch zweckmäßig ist.

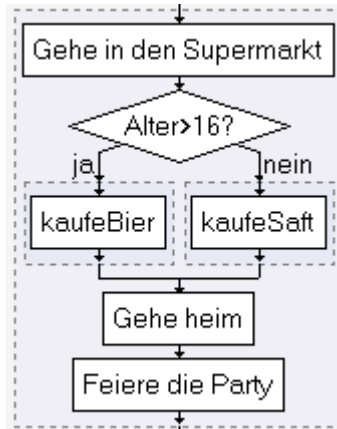
Aufgabe7: Beschreibe die Wirkung dieses Algorithmus:



Aufgabe 7: Erstelle ein Flussdiagramm, das anhand des aktuellen Datums und deines Geburtsdatums dein Alter *errechnet*. Dein Alter soll dann die Fahrzeit für den Roboters in Sekunden sein.

*Tipp:* Verwende die Variablen AktTag, AktMonat, AktJahr, GebTag, GebMonat, GebJahr.

Nun können Entscheidungen eingeführt werden. Das Konzept der Entscheidungen zu verstehen, dürfte für die Schüler keine allzuschwere Aufgabe sein, da Entscheidungen im täglichen Leben überall vorkommen. Dies kann der Lehrer anhand des folgenden Beispiels für einen Partyeinkauf deutlich machen:



Die Schüler sollen das Entscheidungsprinzip mit Hilfe von Variablen nun auf den Roboter übertragen.

Aufgabe 8: Zeichne ein Flussdiagramm, das deinen Roboter nach 3 Sekunden Fahrt eine Rechtsdrehung machen lässt wenn du jünger als 17 bist. Andernfalls soll er nach Links abdrehen.

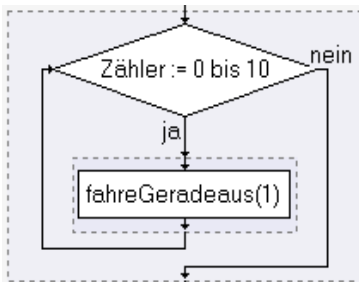
Um die einfache Zählschleife einzuführen, kann der Lehrer 10 Kärtchen mit in den Unterricht bringen. Auf jeder Karte ist ein Roboter abgebildet. Nun kann er die Schüler an der Tafel versuchen lassen, eine Rechnung mit Ergebnis „10 Roboter“ darzustellen. Dies können die Schüler einerseits durch die Addition erreichen, andererseits könnten sie eine Multiplikation verwenden und so 9 Roboter „einsparen“. Anhand des Ergebnisses kann der Lehrer zum Thema Wiederholungen überleiten. Mit der Anweisung „Fahre 1 Sekunde vorwärts“ soll im Lehrer-Schüler-Gespräch geklärt werden, wie es möglich ist den Roboter 10 Sekunden lang vorwärts fahren zu lassen.

Aufgabe 9: Zeichne ein Flussdiagramm, das deinen Roboter im Rechteck fahren lässt. *Tipp:* Du kannst dein Ergebnis aus Aufgabe 6 verwenden.

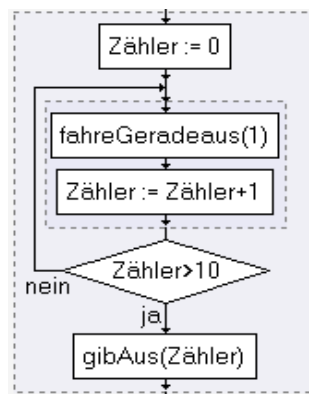
Danach können die Symbole für kopf- und fußgesteuerte Schleifen vorgestellt werden. Diese Schleifen unterscheiden sich von der Zählschleife in der Hinsicht, dass eine Bedingung an ihre Ausführung geknüpft ist. Mit dieser Information kann der Lehrer die Schüler die folgenden Aufgaben lösen lassen:

Aufgabe 10: Erkläre der Unterschied zwischen den verschiedenen Schleifenarten. Wie ist der Zählerstand jeweils bei der Ausgabe bei b) und c)? Zusatzaufgabe: Wie ist das Ergebnis bei d)?

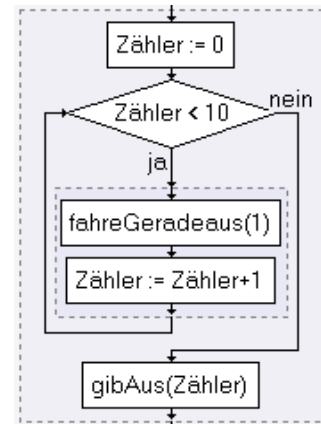
a)



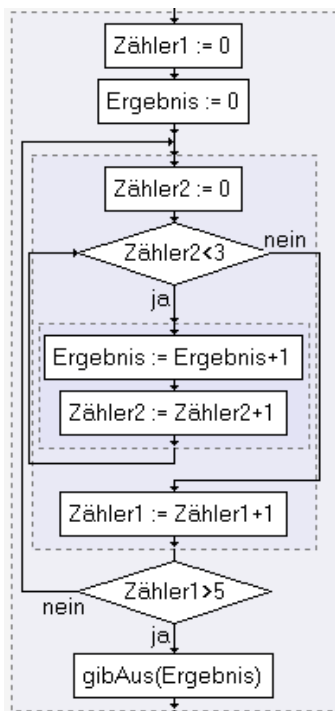
b)



c)



d)



Der Lehrer sollte nun die wichtigsten Unterschiede zwischen den Schleifenarten an einem Schaubild an der Tafel zusammenfassen.

## 5. Variablen, Entscheidungen und Wiederholungen

### 5.1 Zielformulierung

In dieser Phase sollen die Schüler ihr theoretisches Wissen, das sie über Variablen, Wiederholungen und Entscheidungen kennen gelernt haben festigen. Sie können die bereits gemachten Flussdiagramme äquivalent im Quellcode darstellen, sowie eigenen Quellcode in Flussdiagrammen darstellen.

### 5.2 Sozialform

Anfangs Einzelarbeit. Zur Bearbeitung der weiteren Aufgaben sollten sich die Schüler in ihren Gruppen zusammenfinden und diese gemeinsam am PC lösen.

### 5.3 Durchführung

Nachdem die Schüler ein theoretisch fundiertes Wissen über die Kontrollstrukturen haben, sollen sie dieses Wissen an den Lego-Robotern testen. Bevor die Schüler mit dem Erstellen des Quellcodes beginnen, sollte der Lehrer ihnen zeigen, wie man in RIS Kommentare schreibt. Die folgenden Aufgaben sollen nun von den Schülern allein am PC gelöst, und die Lösung jeweils am Gruppenroboter getestet werden, wobei die wichtigsten Stellen im Programm kommentiert werden müssen.

Aufgabe 11: Euer Roboter ist mit zwei Stoßsensoren ausgestattet. Entwerfe folgenden Algorithmus: Wenn ihm bei der Fahrt am linken Sensor ein Hindernis im Weg steht, dann soll er anhalten, kurz zurücksetzen, nach rechts ausweichen und nach 2 Sekunden stehen bleiben. *Tipp:* Verwende für das geradeaus fahren eine „Endlosschleife“ (Entscheidungen, Schleifen)

Aufgabe 12: Euer Roboter hat auch einen Lichtsensor, welcher ab einem gewissen Lichtwert ausgelöst wird. Finde einen Algorithmus, der euer Roboter geradeaus fahren lässt. Wenn er die Linie überquert soll er zum Zeichen dass er sie gefunden hat zweimal piepsen. (Entscheidungen, Schleifen)

Aufgabe 13: Der Roboter soll nach rechts ausweichen, wenn er mit dem linken Sensor gegen das Hindernis gestoßen ist. Wenn er mit dem rechten Sensor dagegen gestoßen ist, soll er nach links ausweichen. (Entscheidungen, Schleifen)

Aufgabe 14: Setze die Aufgaben 11-13 in ein entsprechendes Flussdiagramm um.

Die Schüler sollen sich nun in Gruppen von 2-3 zusammenfinden um die folgenden Aufgaben zu lösen. Der Lehrer kann den Schülern bei Problemen Hilfestellung geben. Am Ende sollen die Gruppen den Quellcode ihrer Lösungen mit Kommentaren versehen am Beamer und das entsprechende Flussdiagramm an der Tafel präsentieren und für jede Aufgabe begründen, warum sie sich für diese Art Kontrollstruktur entschieden haben.

Aufgabe 15: Lasst den Roboter mehrmals im Rechteck fahren. Auf dem Display soll jeweils angezeigt werden, wieviele Kurven bereits gefahren wurden.

Aufgabe 16: Baut eine Teststrecke mit mehreren Hindernissen auf. Verwendet euer Ergebnis aus Aufgabe 1 um den Roboter genau dreimal einem Hindernis ausweichen zu lassen.

Aufgabe 17: Nehmt mehrere DIN A0-Blätter und zeichnet einen großen Kreis darauf. Programmiert euren Roboter so, dass er nach dem Einschalten ständig fährt aber den Kreis nicht verlässt.

Aufgabe 18: Entwerft einen Algorithmus, der, wenn man den Roboter auf die Tischmitte setzt und anschält, losfährt ohne vom Tisch herunterzufallen.

Aufgabe 19: Euer Roboter soll nach dem Einschalten erkennen können, ob er an der Tischkante steht. Steht an an der Kante, so soll er zuerst zurückfahren und wenden bevor er weiterfährt

Aufgabe 20: Erweitere Aufgabe 19 so, dass auf dem Display des Roboters angegeben wird, wie oft der Roboter bereits einem Abgrund ausgewichen ist

Aufgabe 21: Setze das folgende Verhalten in ein entsprechendes Flussdiagramm um: Der Roboter soll ein Hindernis von unbekannter Größe umfahren können. Hinter dem Hindernis befindet sich eine Ziellinie auf der der Roboter zum Stillstand kommen soll.

## **6. Abschlussprojekt**

### **6.1 Zielformulierung**

Die Schüler sollen zum Abschluss der Lerneinheit ihre Kenntnisse im Umgang mit Anweisungen, Anweisungsfolgen, Entscheidungen, Wiederholungen, Variablen und Flussdiagrammen festigen.

### **6.2 Sozialform**

Gruppen- bzw. Projektarbeit

### **6.3 Durchführung**

Mit dem Abschlussprojekt sollen die Schüler ihr Wissen festigen. Das Abschlussprojekt kann eine zeitintensive Aufgabe für die einzelnen Gruppen sein. Es wäre denkbar, dass die Schüler ihr Abschlussprojekt am Tag der offenen Tür der Schule präsentieren und somit gleichzeitig gute „Werbung“ für die Informatik-AG machen.

Abschlussprojekt: Entwerft einen Algorithmus, welcher euren Roboter auf einer Linie entlang fahren lässt. Sobald ihm auf dieser Linie ein Hindernis im Weg steht, soll er das Hindernis umfahren und anschließend auf der Linie weiterfahren. Skizziert sein Verhalten zuerst in einem Flussdiagramm bevor ihr mit der Programmierung beginnt!