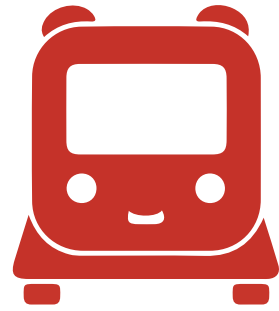


Robotik



Modul 6

Impressum

Herausgeber

Landesanstalt für Kommunikation Baden-Württemberg (LFK)
Anstalt des öffentlichen Rechts,
vertreten durch den Präsidenten Dr. Wolfgang Kreißig
Reinsburgstraße 27
70178 Stuttgart

**Gefördert durch das Ministerium für Kultus, Jugend und Sport
Baden-Württemberg**

Autor

Timo Lachenmaier

Redaktion

Laura Jaenicke, Landesanstalt für Kommunikation BW (LFK)
Dejan Simonović, Stadtmedienzentrum Stuttgart am LMZ BW

Design und Layout

Jana Falkner

Illustrationen

Ilan Backmann

Stuttgart | September, 2022

2. Auflage

Lizenz CC-BY-SA 4.0

Modul 6 Impressum

Die Handreichung steht unter <https://games-im-unterricht.de/toolkit> auch als PDF zur Verfügung.

Die Rechte der verwendeten Grafiken und Bilder liegen, soweit nicht anders vermerkt, beim Stadtmedienzentrum Stuttgart.

Das Werk enthält Screenshots aus dem verwendeten Programm Makeblock. Diese sind selbst erstellt und werden im Sinne eines Zitats zu Bildungszwecken genutzt.



Beschreibung	5
Einbindung in die Story	6
Rahmenbedingungen	7
Vorbereitung	10
Durchführung Modul 6 Robotik	11
6.1 Den Codey Rocky kennen lernen	11
6.2 Auf die Plätze, fertig, los	13
6.3 Bedingung und Verzweigung	16
6.3a Es werde Licht	16
6.3b Staubsaugerroboter	19
6.4 Laserduell	21
6.5 Wenn..., dann...	24
Arbeitsblätter	26

Hallo, ich bin
Platina Toolkid!







Beschreibung

Modul 6 | Robotik

Modul 6.1

Codey Rocky
kennenlernen

Modul 6.2

Auf die Plätze,
fertig, los

Modul 6.3a

Es werde Licht

Modul 6.3b

Staubsaug-
roboter

Modul 6.4

Laserduell

Modul 6.5

Wenn..., dann...

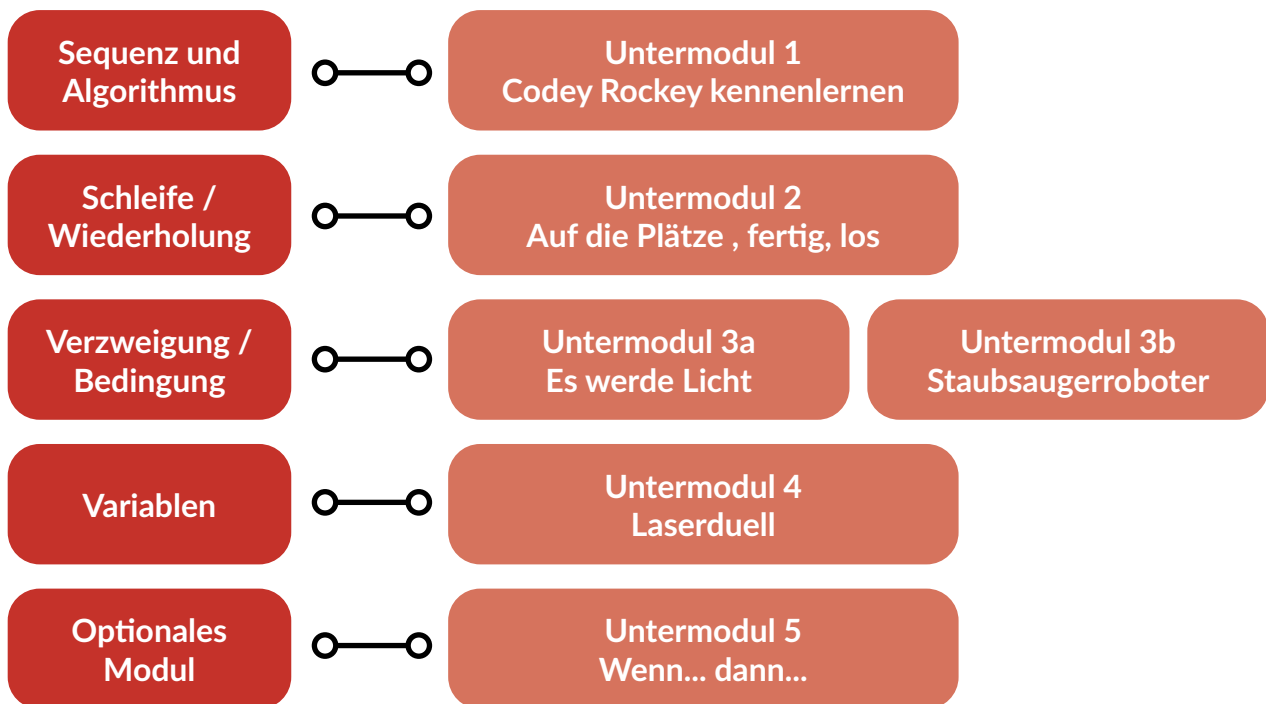
Das Modul 6 Robotik umfasst vier Untermodule und weitere Bestandteile, die als Bonusmodule optional behandelt werden können. Die einzelnen Einheiten sind auf den Bildungsplan Baden-Württemberg für den Aufbaukurs Informatik in Klasse 7 angepasst. Dabei werden alle Kompetenzen des Bereichs Algorithmen, besonders die algorithmischen Grundbausteine, und auch Kompetenzen aus anderen Bereichen wie Modellieren und Implementieren, abgedeckt.

Nach dem Kennenlernen der Bedienung der Roboter und der blockbasierten Programmiersprache navigieren die Schüler:innen die Roboter durch einen Parcours. Anschließend bringen sie den Robotern entweder bei, automatisch Hindernissen auszuweichen oder bei nachlassender Lichtstärke die Beleuchtung selbstständig einzuschalten. In der nächsten Stunde treten die Roboter dann in einer Arena gegeneinander zu einem Spiel an. Abschließend können die erworbenen Kenntnisse bei verschiedenen, differenzierten Aufgaben angewandt werden.

Codey Rocky



Anknüpfungspunkte an den Bildungsplan Informatik



Einbindung in die Story

Der Bad Bug ist schon wieder entwischt und versteckt sich erneut in einem Labyrinth, das sogar für Chip zu eng ist. Nachdem Platina die letzten Herausforderungen bewältigt hat, wird nun Chip selbst aktiv. Platina schenkt ihm dafür mit dem Codey Rocky einen eigenen Roboter, den Chip programmieren darf. Mithilfe des Roboters sollte es möglich sein, das Labyrinth zu durchfahren und den Bug zu fangen.

Platina hilft ihm sich bei der Bedienung des Roboters zurechtzufinden (Untermodul 6.1). Im Anschluss versucht Chip, das Labyrinth mit dem Roboter zu durchfahren (Untermodul 6.2) und den Bug zu fangen. Es dauert länger als geplant und die Dämmerung bricht herein. An einigen Stellen des Labyrinths ist es bereits dunkel und der Codey Rocky muss sein Licht einschalten (Untermodul 6.3a). Am nächsten Morgen betritt Platina die Werkstatt und findet

nach der Jagd auf den Bug ein heilloses Durcheinander vor. Gemeinsam wollen sie einen Staubsaugerroboter programmieren (Untermodul 6.3b). Nach dem Aufräumen möchten sie zur Belohnung gemeinsam ein Spiel spielen und lassen die Codey Rockys gegeneinander antreten (Untermodul 6.4).





Rahmenbedingungen



Zeit

4 bis 8 Unterrichtsstunden oder ein bis zwei Projekttag.

Die meisten Einheiten sind in 45 Minuten durchführbar. Die einzelnen Themen können aber auch auf zwei Stunden oder eine Doppelstunde ausgedehnt werden. Dabei geben die Differenzierungsmöglichkeiten den fortgeschrittenen Schüler:innen Anreiz zur vertiefenden Arbeit, während unerfahrene Schüler:innen von der längeren Zeit profitieren, indem sie ihre Algorithmen ausführlich testen und verbessern. Die Differenzierungen finden Sie auf den laminierten Kärtchen. Die Rückseite und die Nummern geben an, bei welchem Modul sie eingesetzt werden können. Der Schwierigkeitsgrad ist durch die verschiedenen Darstellungen von Chip gekennzeichnet.

Wenn Sie ein Untermodul auf zwei Stunden ausdehnen, können Sie einen Teil dieser Doppelstunde nutzen, um den fachwissenschaftlichen Bezug stärker herauszuarbeiten.

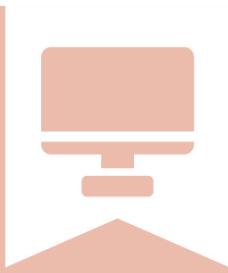
Das Modul bietet sich auch gut für Methoden- oder Projekttag an.



Raum

Klassenzimmer

Grundsätzlich gibt es keine speziellen Anforderungen an den Unterrichtsraum. Damit die Roboter sich bewegen können, wäre ausreichend Platz auf dem Boden bzw. auf Tischen wünschenswert. Sollten Sie keine mobilen Endgeräte nutzen können, dann gibt es die Möglichkeit, die Einheit in einem Computerraum durchzuführen.



Technik und Materialien

Technik im Koffer | Tafel & Kreide

Die Schüler:innen benötigen pro Team mindestens ein digitales Endgerät (Smartphone, Tablet, Computer). Man kann sowohl die mBlock App als auch den Browser nutzen.

mBlock App Browser



<https://mblock.makeblock.com/en-us/download/>

Wir empfehlen ein Tablet mit der mBlock App, da die Roboter per Bluetooth unkompliziert verbunden werden können und man nicht an einen Computerraum gebunden ist. Außerdem kann man so ohne einen Account Codes speichern und die Displaygröße ist groß genug.

Mehrere Teams können sich einen Codey Rocky teilen. Dadurch sind kleinere Teams möglich. Je größer das Team, desto eher übernimmt eine/ein Expert:in die Hauptarbeit.

Fragen Sie in Ihrem Kreismedienzentrum nach, ob Sie dort noch mehr Codey Rockys leihen können.

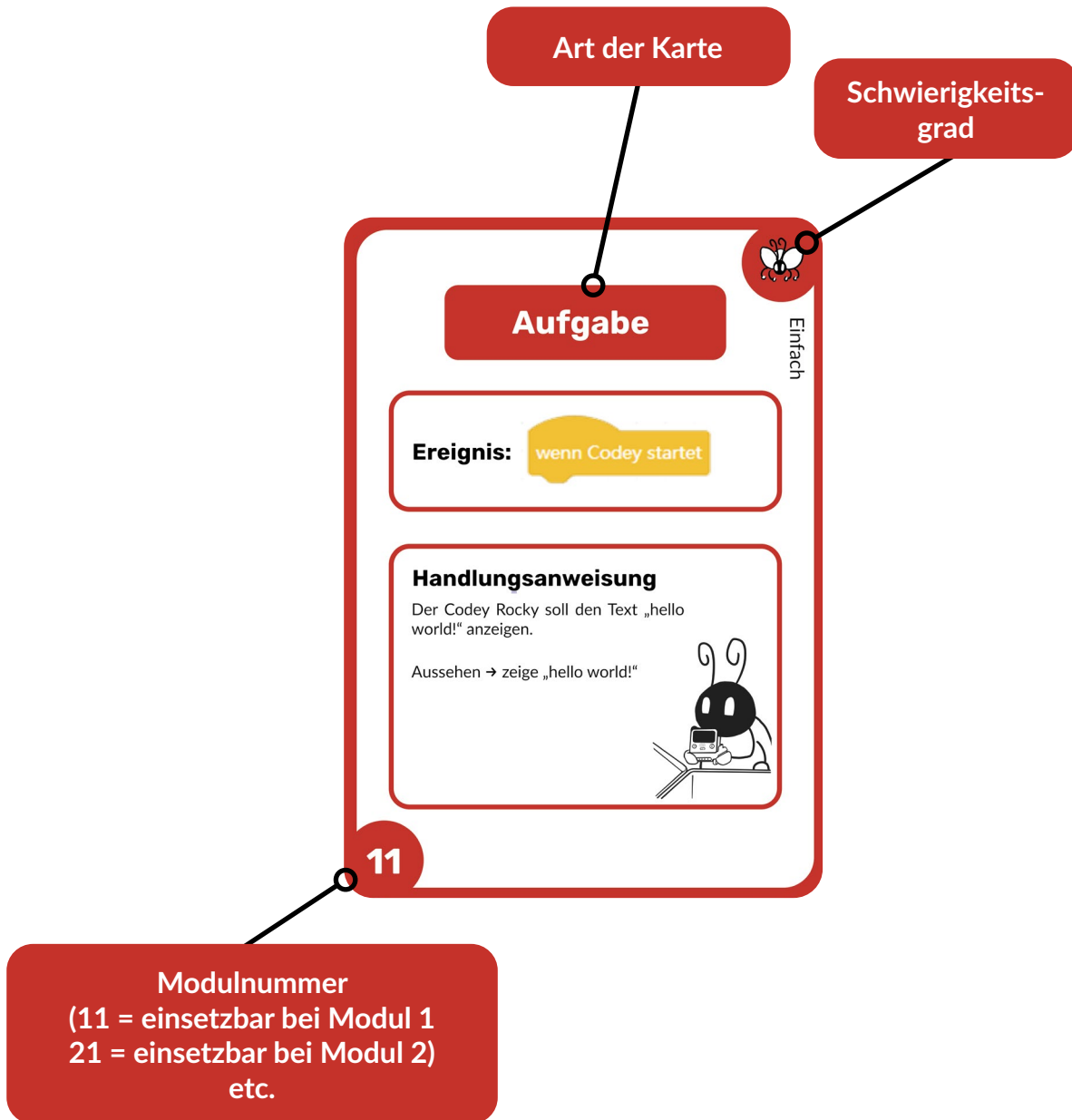
Vor jeder Stunde sollten Sie die Codeys und die Endgeräte laden.

Arten von Karten





Erklärung Karten



Schwierigkeitsstufen



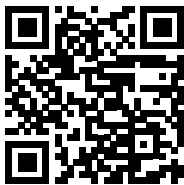
Vorbereitung

Für die Durchführung des Moduls Robotik brauchen Sie aus dem Koffer des Toolkits die Roboter. Sie bestehen aus zwei Teilen: einem Display mit Bedienelement, dem Codey, und einem fahrbaren Untersatz, dem Rocky.

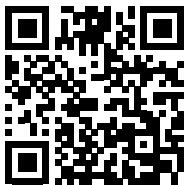
Sollen die Schüler:innen an Computern programmieren, benötigen Sie auch die Verbindungskabel, die Sie ebenfalls im Koffer finden.

Machen Sie sich zunächst mit Hilfe der vier Lernvideos mit den Grundlagen der mBlock App und dem Codey Rocky vertraut.

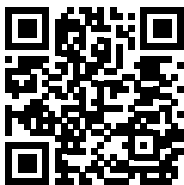
Lernvideos



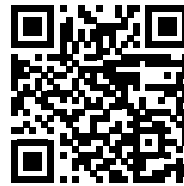
<https://vimeo.com/513196288/3d761a3ad8>



<https://vimeo.com/513196362/f6f41a35b2>



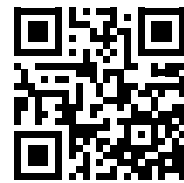
<https://vimeo.com/513196432/45c8bf9173>



<https://vimeo.com/513196509/2db3c760ef>

Dabei lernen Sie die Bedienung der App und die Kommunikation mit dem Roboter kennen. Installieren Sie dann die mBlock App auf den Endgeräten. Um selbst Sicherheit im Umgang mit der App zu gewinnen, ist es empfehlenswert, mehrere Aufgaben der ersten Unterrichtsstunde auszuwählen und diese selbst auszuprobieren.

Mehr über die App und die Codey Rockys erfahren Sie auf der Internetseite:



education.makeblock.com

Vor jeder Unterrichtsstunde müssen Sie die dort angegebenen Materialien ausdrucken. Im Koffer finden Sie außerdem mehrere laminierte Aufgabenkarten. Bei den einzelnen Unterrichtsstunden ist angegeben, welche der Kärtchen Sie für die jeweilige Stunde brauchen.



Durchführung Modul 6 Robotik

6.1 Den Codey Rocky kennen lernen

Die Schüler:innen lernen die Bedienung sowie die Input- und Output-Optionen des Codey Rockys kennen. Dabei erfahren sie, wie man auf dem Endgerät einen Code erstellt, diesen per Bluetooth auf den Roboter überträgt und ihn selbst direkt bedient.

Dabei sollen die Schüler:innen viele kleine Aufgaben bearbeiten, um einen möglichst guten Überblick über die Möglichkeiten zu erhalten. Zum Beispiel sollen sie auf dem Display „hello world“ anzeigen lassen, die Beleuchtung am Roboter einschalten oder den Roboter die Form eines Quadrats fahren lassen.

Die Aufgaben sind in drei Schwierigkeitsstufen (Level) eingeteilt: Einfach, mittel und schwierig. Zu jedem Level gibt es neun unterschiedliche Aufgaben. Bei den einfachen Aufgaben sind sowohl das Ereignis (z.B. „wenn Taste A gedrückt“) wie auch die Handlungsanweisungen klar vorgegeben. Es ist außerdem angegeben, in welchem Bereich man die Handlungsanweisung findet (z.B. „Aussehen“). Sie finden die Aufgaben im Koffer und als Vorlage (siehe Modul 6 - Aufgabenkarten).

Alle Teams beginnen mit einfachen Aufgaben (Level 1). Erst wenn sie eine einfache Aufgabe erfolgreich erledigt haben, können sie eine Aufgabe aus Level 2 (mittel) wählen und dann im Anschluss analog eine schwierige Aufgabe auswählen. Sie steigen also in den Leveln auf. Wichtig ist, dass die Schülerinnen und Schüler auch noch andere Aufgaben aus demselben Level wählen können, bis sie sich sicher fühlen.

Die Schüler:innen protokollieren ihren Fortschritt auf dem Arbeitsmaterial „Codey Rocky kennen lernen“ (siehe S. 27) und/oder dem Codebook (siehe Modul 6 - Handbücher).

Die Schüler:innen lernen also vor allem die algorithmischen Grundbausteine Anweisung und Sequenz kennen und implementieren diese in einer visuellen Programmierungsumgebung.

Einbindung der Story

Platina schenkt Chip einen eigenen Roboter, den Chip programmieren darf. Platina hilft ihm dabei, sich bei der Bedienung des Roboters zurechtzufinden. Sie als Lehrperson übernehmen dabei zunächst die Rolle Platinas, die Chip die Grundlagen der Steuerung erklärt. Die Schüler:innen übernehmen dann die Rolle Chips, der verschiedene Möglichkeiten des Roboters ausprobiert.

Verlaufsplan | 6.1 Den Codey Rocky kennen lernen

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg	Verteilen der Codey Rockys, Teamfestlegung, Zuordnung der Teams zu festen Nummern, Verhaltensregeln mit den Robotern festlegen, Handbücher austeilen.	Handbücher
5 min	Hinführung	Einführung in die App Makeblock und die Bedienung des Codey Rockys: • Start, „Programmi...“ auswählen, über „+“ den „Codey“ auswählen und dann mit dem Haken rechts oben bestätigen • Beispiel-Baustein: • Ereignis Wenn Taste B gedrückt → Aussehen • Achten Sie darauf, dass Sie ein Beispiel wählen, bei dem man durch den Pfeil verschiedene Optionen auswählen kann. • Hochladen per Bluetooth über den Pfeil. • Am Roboter B drücken und so den Code starten.	Präsentationsfläche mit eigenem Codey
5 min	Arbeitsanweisung	Vorstellen der Aufgabenkärtchen und Arbeitsanweisung: • Jedes Team beginnt in Level 1 (einfach, hier gibt es mehr Hilfestellungen). • Nach erfolgreicher Erledigung wird die entsprechende Aufgabe auf dem Sicherungsblatt abgehakt. • Danach steigt das Team auf und kann Aufgaben der mittleren Schwierigkeitsstufe lösen. Die Schüler:innen können aber auch erst noch weitere Aufgaben aus Level 1 lösen. • Im Anschluss können schwierige Aufgaben gewählt werden.	Rocky kennen lernen (siehe S. 27 & 28) Aufgabenkarten



25 min	Erarbeitung Gruppenarbeit	Die Schüler:innen wählen selbst Aufgaben und erledigen diese in Teams.	Aufgabenkarten
5 min	Sicherung Einzelarbeit	Codey Rockys einsammeln Sicherung/HA Tabelle ausfüllen	

Anknüpfungspunkte Informatik | 6.1 Den Codey Rocky kennen lernen

Informatik	Modul 6.1
Sequenz	Die Blöcke werden nacheinander in der Reihenfolge abgearbeitet.
Input	Eingabe durch Drücken der Tasten auf dem Roboter.
Output	Ausgabe, beispielsweise durch das Display oder Sound.

6.2 Auf die Plätze, fertig, los

Die Schüler:innen erstellen einen sequenziellen Code, mit dem der Roboter einen selbst erstellten Parcours absolviert. Der Parcours wird von zwei Teams gemeinsam gestaltet. Dabei wählen die Teams selbst einen Startpunkt und ein Ziel und legen abwechselnd Hindernisse in den Weg. Das Team, das den Codey Rocky schneller erfolgreich durch den Parcours navigiert, hat gewonnen.

Im zweiten Schritt wird der Code so ausgebaut, dass mit Hilfe von Schleifen der Parcours mehrfach durchlaufen werden soll. Dabei werden unterschiedliche Schleifen auf ihre Tauglichkeit für die Problemstellung verglichen. Am Ende der Stunde erhalten die Schüler:innen das Arbeitsmaterial mit den Merksätzen zu allen algorithmischen Grundbausteinen. Dabei stehen drei Varianten in unterschiedlicher Schwierigkeit zur Verfügung (siehe S. 29 - 34).

Neben den algorithmischen Grundbausteinen Anweisung und Sequenz wird die Schleife eingeführt. Außerdem werden Programme gezielt getestet und bewertet.

Einbindung der Story

Chip versucht, das Labyrinth mit dem Roboter zu durchfahren und den Bad Bug zu fangen. Definieren Sie das Ziel als den Standort des Bugs und die Hindernisse, die die Schüler:innen in den Weg legen, als Labyrinth. Die Aufgabe ist dann, den Bug so schnell wie möglich zu finden und unschädlich zu machen. Sie könnten den Bug kurz vor Ende wieder an den Start fliegen lassen und sobald Codey Rocky dort auftaucht wieder zum Ende fliegen lassen. Nach fünf Wiederholungen ist der Bug so erschöpft, dass er nicht mehr fliegen und Chip ihn fangen kann.

Verlaufsplan | 6.2 Auf die Plätze, fertig, los

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg Arbeitsanweisung	Ziel, Parcours, Ablauf der Stunde vorstellen. Arbeitsanweisung 1: Zwei Teams gestalten gemeinsam einen einfachen Parcours. Bestimmt gemeinsam den Start und das Ziel (Entfernung maximal 2m). Im Anschluss darf jeder von Euch ein Hindernis in den Weg legen (z.B. Mäppchen oder Schulbücher). Wettbewerb 1 Welchem Codey Rocky gelingt es als erstes, den Parcours erfolgreich zu durchfahren? Wettbewerb 2 Welchem Codey Rocky gelingt es, den Parcours in der kürzesten Zeit zu durchfahren?	Präsentationsfläche Kreppband für Start- und Zielmarkierungen.
15 min	Erarbeitung I Gruppenarbeit	Wettbewerb 1+2 wird durchgeführt: 1) Welcher Codey Rocky schafft es als erstes, vom Start zum Ziel zu kommen? 2) Welcher Codey Rocky schafft es in der kürzesten Zeit? Schüler:innen programmieren den Code und testen. Optional Hinweis für Schüler:innen: Der Baustein „um xx° nach rechts drehen“ (bzw. nach links) erleichtert die Navigation enorm.	Parcours, Codey Rocky, Endgeräte
5 min	Sicherung I Gruppenarbeit	Teams, die bisher keinen erfolgreichen Durchlauf durch den Parcours erreicht haben, dürfen sich bei den anderen Teams Anregungen holen.	



5 min	Problemati- sierung Unterrichts- gespräch	Übergang zu Wettbewerb 3, Ar- beitsanweisung 2: Wettbewerb 3 • Der Codey Rocky soll den Par- cours zwei Mal durchlaufen (UG: Zurück zum Start, dann nochmal). Idee: Code verdop- peln. Umständlich, aber mög- lich. • Der Codey Rocky soll den Par- cours zwanzig Mal durchlau- fen (Problem). • Impuls: das muss doch einfa- cher gehen: • Lehrperson stellt Schleifen vor (Steuerung wiederholen).	Präsentations- fläche
10 min	Erarbeitung II Gruppen- arbeit	Wettbewerb 3 Welchem Codey Rocky gelingt es als erstes, den Parcours fünf Mal erfolgreich zu durchfahren (ohne Pause)? Hinweis: Da die Roboter nicht genau 90° einhalten, ist es beinahe unmöglich, durch Schleifen den Parcours zwei Mal zu durchfahren. Dies sollte im Anschluss themati- siert werden (Theorie / Praxis). Puffer: Arbeitsblatt Merksätze.	
5 min	Sicherung II Unterrichts- gespräch	Wiederhole „Anzahl“, Wiederhole bis, Wiederhole fortlaufend. HA: Arbeitsblatt Merksätze Aufga- be 1 (3 Schwierigkeitsstufen).	Merksätze ein- fach – mittel – schwierig (siehe S. 29 - 34)

Anknüpfungspunkte Informatik | 6.2 Auf die Plätze, fertig, los

Informatik	Modul 6.2
Schleifen	Der Parcours soll mehrfach durchfahren werden.
Anweisung / Sequenz	Der Roboter wird durch den Parcours geführt.
Optimierung	Der Roboter soll den Parcours schneller durchfahren.

6.3 Bedingung und Verzweigung

Hier können Sie zwischen 6.3a und 6.3b wählen. Bei 6.3a „Es werde Licht“ soll der Roboter bei geringer Helligkeit das Licht automatisch anschalten, bei 6.3b „Staubsaugerroboter“ weichen die Roboter automatisch Hindernissen aus.

Falls Sie wenig Zeit haben, empfehlen wir 6.3b, ansonsten können Sie auch gerne beide Einheiten durchführen.

6.3a Es werde Licht

Anschließend an die letzte Stunde wird erneut ein einfacher Parcours aufgebaut, diesmal aber von der Lehrkraft. Der Parcours sollte nicht zu komplex sein. Wichtiger Bestandteil sind zwei längere Geraden und wenige Hindernisse. An einer Stelle einer Geraden steht ein Tunnel. In diesem Tunnel soll das Licht eingeschaltet werden.

Im ersten Teil der Stunde programmieren die Schüler:innen einen Code, um den Parcours zu bewältigen. Vermutlich wird der Code eine Sequenz enthalten, mit der zur passenden Zeit das Licht eingeschaltet wird. Das Ziel soll jedoch sein, mit Hilfe einer Bedingung und einer Verzweigung das Licht automatisch einzuschalten. Deshalb verändert die Lehrkraft die Position des Tunnels nach einer Weile als stummen Impuls. Danach wird die „Wenn... dann ... sonst ...“ -Verzweigung eingeführt und die Sensorenmessung der Lichtstärke thematisiert. Im Anschluss erweitern die Schüler:innen ihren Code, testen und verbessern ihn.

Zur Differenzierung können Sie die Differenzierungskarten 40 – 45 nutzen. Sie liegen in ausgedruckter Form und als Kopiervorlage vor (siehe Modul 6 - Differenzierungskarten).

Die Schüler:innen lernen also, Sensoren zu steuern und die gemessenen Daten zu nutzen, um durch Bedingungen und Verzweigungen unterschiedliche Handlungsalternativen zu implementieren.

Einbindung der Story

Während Chip mit Codey Rocky den Bad Bug jagt, vergeht aufgrund der Schleifen Zeit und es wird an manchen Stellen des Parcours dunkel. Chip möchte also das Licht einschalten. Um den Bad Bug aber durch das Licht nicht ständig vorzuwarnen, soll das Licht nur an den Stellen eingeschaltet werden, wo es wirklich dunkel ist. Am Ende wird der Bad Bug endgültig gefangen.

Codey Rocky





Verlaufsplan | 6.3a Es werde Licht

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg	Ziel der Stunde vorstellen: Ein einfacher Parcours soll durchfahren werden, an einer Stelle soll in einem Tunnel das Licht eingeschaltet werden, Ablauf der Stunde vorstellen	Parcours, Tunnel (aufgeschnittener Karton)
10 min	Erarbeitung I Gruppenarbeit	Ziel: Im Tunnel soll das Licht eingeschaltet und danach ausgeschaltet werden. Schüler:innen programmieren den Code und testen. Die Schüler:innen programmieren vermutlich den Block, mit dem sie das Licht zur richtigen Zeit einschalten.	Endgeräte, Codey Rocky
5 min	Problematisierung Unterrichtsgespräch	Impuls: Was passiert, wenn ich den Tunnel verschiebe? Ziel: Selbst fahrendes Auto: Das Licht soll immer eingeschaltet werden, wenn es dunkel ist und immer ausgeschaltet werden, wenn es hell ist. Notwendigkeit für Bedingung (wenn...) und Verzweigung (dann..., sonst...) erläutern und als Flussdiagramm darstellen. Bedienung in Makeblock zeigen (Steuerung) und die Nutzung des Sensors (Fühlen) ergänzen. Hinweis: Der Schwellenwert für den Lichtsensor ist stark von der Umgebungshelligkeit abhängig. →Man braucht Sensoren und wenn... dann. Die Bedingung (wenn...) und die Verzweigung (dann..., sonst...) muss fortlaufend wiederholt werden.	Präsentationsmedium



20 min	Erarbeitung II Gruppenarbeit	Schüler:innen erweitern den Code und überprüfen die Ergebnisse direkt am Parcours. Differenzierungsmöglichkeiten: siehe unten.	Differenzierungskarten (40 – 45)
5 min	Sicherung Einzelarbeit	Codey Rockys einsammeln, Sicherung Arbeitsblatt Merksätze	Arbeitsmaterial Merksätze einfach – mittel – schwierig (siehe S. 29 – 34)

Anknüpfungspunkte Informatik | 6.3a Es werde Licht

Informatik	Modul 6.3a
Bedingungen	Wenn es dunkel ist, dann soll das Licht eingeschaltet werden.
Sensoren	Lichtstärke muss gemessen werden.
Schleife	Die Lichtstärke muss fortlaufend überprüft werden.
Werte	Der Schwellenwert für das Licht muss angepasst werden.

6.3b Staubsaugerroboter

Die Schüler:innen lernen, den IR-Sensor zu steuern, um wie ein Staubsaugerroboter Hindernisse zu erkennen und auszuweichen. Dabei werden auf Schleifen („Wiederhole immer“) und auf Verzweigungen („Wenn `sieht Hindernis`, dann...“) zurückgegriffen. Nach ersten Erfolgen soll der Roboter nicht nur ausweichen, sondern das Hindernis möglichst umfahren. Im Anschluss werden unterschiedliche Ergebnisse der Teams verglichen und bewertet. Dabei können die Schüler:innen Hilfekarten in Anspruch nehmen (siehe Modul 6 - Hilfekarten).

Zur Differenzierung können Sie die Differenzierungskarten 50 – 54 nutzen.

Sie liegen in ausgedruckter Form und als Kopiervorlage vor (siehe Modul 6 - Differenzierungskarten).

Die Schüler:innen lernen wie bei 6.3b, Sensoren zu steuern und die gemessenen Daten zu nutzen, um durch Bedingungen und Verzweigungen unterschiedliche Handlungsalternativen zu implementieren.

Einbindung der Story

Nach der Jagd auf den Bug ist in Platinas Werkstatt doch so einiges durcheinandergeraten. Überall liegen Blätter von den Hecken des Labyrinths. Platina schlägt vor, Codey Rocky als Staubsauger zu benutzen. Damit er nichts zerstört, soll er Hindernissen automatisch ausweichen.

Verlaufsplan | 6.3b Staubsaugerroboter

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg	Ziel, Ablauf der Stunde vorstellen	siehe S. 37
5 min	Problematisierung Unterrichtsgespräch	Der Klasse wird ein Codeabschnitt mit einer Bedingung und einer Verzweigung vorgestellt. Die Schüler:innen versuchen den Code zu interpretieren Hilfsimpuls: An der Tafel wird eine Strecke als Hindernis gezeichnet und mögliche Wege um das Hindernis herum. Schwierigkeit 1: Nur ausweichen und in eine andere Richtung fahren. Schwierigkeit 2: Hindernis umfahren. Notwendigkeit für Bedingung (wenn...) und Verzweigung (dann..., sonst...) erläutern und als Flussdiagramm darstellen. Bedienung in Makeblock zeigen (Steuerung) und die Nutzung des Sensors (Fühlen) ergänzen.	Parcours, Tunnel (aufgeschnittene Schachtel)



10 min	Erarbeitung I Gruppenarbeit	Die Schüler:innen versuchen, den Roboter ausweichen zu lassen. Die Gruppen können sich dabei austauschen.	Endgeräte Codey Rocky
5 min	Weiterführung Unterrichtsgespräch	Das Hindernis wird erweitert und komplexer gestaltet (sehr tiefes Hindernis).	Präsentationsmedium
15 min	Erarbeitung II Gruppenarbeit	Die Schüler:innen testen, wie der Code auf das komplexere Hindernis reagiert und verbessern ihren Code.	Differenzierungskarten (50 – 54)
5 min	Sicherung Einzelarbeit	Codey Rockys einsammeln, Sicherung Arbeitsblatt Merksätze.	Arbeitsmaterial Merksätze einfach – mittel – schwierig (siehe S. 29 - 34)

Anknüpfungspunkte Informatik | 6.3b

Informatik	Modul 6.3b
Bedingungen	Wenn ein Hindernis voraus ist, muss der Codey Rocky ausweichen.
Sensoren	Hindernisse müssen erkannt werden.
Schleife	Es muss ständig auf Hindernisse geprüft werden.
Werte	Je kürzer der Codey Rocky fährt, desto häufiger überprüft er, ob er inzwischen ein Hindernis erkennt.

6.4 Laserduell

Die Schüler:innen programmieren die Infrarotsensoren so, dass Nachrichten zwischen Codey Rockys ausgetauscht werden können. Jede Nachricht symbolisiert einen Treffer durch einen Laserstrahl. Eine Variable funktioniert als Zähler für Lebenspunkte, die durch die Laserstrahlen schrittweise reduziert werden.

Der Code ist komplexer als der bisherige Code. Zunächst wird im Unterrichtsgespräch mit Hilfe von Codeschnipseln ein Struktogramm erstellt (siehe S. 38 & 39). Dabei werden auch die Bedeutung und Umsetzung von Operatoren besprochen. Im Anschluss erhalten die Teams das Struktogramm (siehe S. 38) und wandeln es in einen Code um und testen diesen. Differenzierte Hilfekärtchen (siehe Modul 6 - Hilfekarten) unterstützen die Schüler:innen bei der Implementierung. Um genügend Zeit zu ermöglichen, wird die Einheit auf 90 Minuten ausgedehnt. Den Schluss der Stunde können Sie nutzen, um das erworbene Wissen mit Hilfe des Arbeitsmaterials (siehe S. 27-35) zu sichern.

Im Raum muss eine „Arena“ für die Codey Rockys geschaffen werden. Dafür bietet sich ein Quadrat mit einer Seitenlänge von 1,5m an. Es kann bspw. durch Tische abgesichert werden.

Zur Differenzierung können Sie die Differenzierungskarten 60 – 63 nutzen. Sie liegen in ausgedruckter Form und als Kopiervorlage vor (siehe Modul 6 - Differenzierungskarten).

Weiterhin können Sie den Teams vorgeben, sich nicht um 90° zu drehen, sondern einen Wert zwischen 60° und 120° vorgeben und den Teams unterschiedliche Drehrichtungen zuordnen, damit die Roboter chaotischer fahren.

Mit dem Abschluss dieser Einheit wurden alle algorithmischen Grundbausteine behandelt. Die Einheit 6.5 ist optional, ist aber zur Wiederholung und Vertiefung sinnvoll.

Einbindung der Story

Nach der erfolgreichen Jagd wollen Platina und Chip endlich die Zeit nutzen, um gemeinsam bzw. gegeneinander zu spielen. Weisen Sie die Schüler:innen den Teams „Platina“ und „Chip“ zu, anstatt Team blau und Team rot.

Alternativen

- Die Codey Rockys fahren in einem Gebiet und weichen sich gegenseitig aus. Jedes Mal, wenn sie ausweichen, erhöht sich eine Zählervariable um eins. Die Variable wird ständig angezeigt. Der Roboter mit der höchsten Punktzahl gewinnt.
- Der Codey Rocky misst die Lautstärke im Klassenzimmer. Jedes Mal, wenn die Lautstärke einen Schwellenwert (z.B. 5) übersteigt, verliert der Codey Rocky einen Lebenspunkt. Die Roboter könnten dann auch in anderen Stunden zur Lautstärkekontrolle eingesetzt werden.



Verlaufsplan | 6.4 Laserduell

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg	Ziel, Ablauf der Stunde vorstellen	
5 min	Problematik- sierung Unterrichts- gespräch	Spiel Laserduell vorstellen: Die Klasse wird in zwei Gruppen eingeteilt: Rot und blau bzw. Platin und Chip. Wenn man den A-Knopf drückt, dann fährt der Code Rocky los und weicht Hindernissen automatisch aus. Trifft er auf den Roboter eines anderen Teams, dann soll dieser Roboter zwei Lebenspunkte (zwei Herzen) verlieren. Sobald der Codey Rocky startet, fährt er solange bis ein Hindernis kommt. Wenn er keinen Lebenspunkt mehr hat, soll er ein trauriges Gesicht zeigen, einen traurigen Sound abspielen und alle Aktivitäten einstellen.	Endgeräte, Codey Rocky
10 min	Erarbeitung I Unterrichts- gespräch	Die fehlenden Bausteine des Struktogramms werden gemeinsam in die richtige Reihenfolge gebracht. Lebenspunkte als veränderlichen, variablen Wert erläutern. Bedienung der Variablen in mBlock zeigen, mathematische Operatoren (<, >, = in mBlock vorführen).	Lehrmaterial (siehe S. 38 & 39)



25 min	Erarbeitung II Gruppenarbeit	Die Schüler:innen implementieren ihren Code. Zur Unterstützung erhält jedes Team ein passendes Struktogramm. Außerdem können die Schüler:innen auf die gestuften Hilfekärtchen zurückgreifen. Sind sie zufrieden, suchen sie sich (z.B. An einer Halterstelle) ein gegnerisches Team und testen den Code gemeinsam. Sind alle Gruppen fertig, kann auf einer großen Fläche mit allen gespielt werden.	Kopiervorlagen (Arbeitsblätter, siehe S. 35) Hilfekarten
5 min	Sicherung I Unterrichtsgespräch	Die Codes werden auf die Wirksamkeit überprüft, Möglichkeiten zum Cheaten durch Code werden besprochen.	
25 min	Differenzierung Gruppenarbeit	Die Schüler:innen fixen ihren Code oder bauen ihn aus. Danach suchen sie sich wieder andere Gruppen und testen ihre Algorithmen in der Arena.	Differenzierungskarten (60 – 63)
5 min	Erarbeitung III Unterrichtsgespräch	Bedeutung von gut benannten Variablen erläutern.	
10 min	Sicherung II Einzelarbeit	Die Merksätze werden fertiggestellt, Roboter einsammeln etc.	Arbeitsmaterial Merksätze einfach – mittel – schwierig (siehe S. 29 - 34)

Anknüpfungspunkte Informatik | 6.4 Laserduell

Informatik	Modul 6.4
Grundbausteine	Mit Hilfe einer Schleife und einer Bedingung weichen die Roboter fortlaufend Hindernissen aus.
Variable	Variablen werden als Wertespeicher benutzt (Anzahl der Leben).
Operatoren	Wenn die Variable Leben 0 (oder kleiner) ist, dann hört der Codey Rocky auf zu fahren).

6.5 Wenn..., dann...

Kurzbeschreibung

Der Stundenverlauf wird zunächst anhand eines Struktogramms vorgestellt (siehe S. 40): Die Teams können Aufgaben in unterschiedlichen Schwierigkeitsgraden wählen, beispielsweise soll die Lautstärke oder der Batterieladestand grafisch dargestellt werden. Die Schüler:innen nutzen für ihre Codes dazu unterschiedliche Sensoren, um durch Verzweigungen verschiedene Handlungen auszuführen.

Die passenden Differenzierungskarten für die Aufgaben haben die Nummern 70 – 78 und liegen in ausgedruckter Form und als Kopiervorlage vor (siehe Modul 6 - Differenzierungskarten).

Bei den Aufgaben werden alle zuvor erlernten algorithmischen Grundbausteine genutzt, wiederholt und vertieft. Methodisch sollen die Schüler:innen üben, selbst Struktogramme zu erstellen. Außerdem sollen die Teams ihre Ideen und Lösungswege den anderen Teams präsentieren.

Sie können die Einheit auch ausdehnen, Plakate mit den Struktogrammen entwerfen lassen und die Präsentation als Galeriegang durchführen. Dafür sollten Sie eine Doppelstunde ansetzen.

Anknüpfungspunkte Informatik | 6.5 Wenn..., dann...

Informatik	Modul 6.5
Grundbausteine	Neben Bedingungen werden auch Schleifen benötigt.
Sensoren	Bei den Aufgaben werden verschiedene Sensoren und Output-Optionen genutzt.
Operatoren	Wenn die Variable Leben 0 (oder kleiner) ist, dann hört der Codey Rocky auf zu fahren.
Variablen	Sensorenergebnisse werden gespeichert, Variablen dienen als Wertespeicher, Zufallsvariablen werden optional eingeführt.



Verlaufsplan | 6.5 Wenn..., dann...

Dauer	Phase und Unterrichtsform	Inhalt	Material und Technik
5 min	Einstieg	Ziel, Ablauf der Stunde anhand eines Struktogrammes vorstellen. Gehen Sie auf die Wiederholungen und Bedingungen ein.	Lehrmaterial (siehe S. 40)
20 min	Erarbeitung Gruppenarbeit	Die Schüler:innen wählen selbst Aufgaben in verschiedenen Schwierigkeitsstufen. Bei jeder Aufgabe sollen sie im Vorfeld ein Struktogramm entwerfen. Wir empfehlen, die Einheit auf zwei Unterrichtsstunden auszudehnen. Geben Sie Ihren Schüler:innen dann hier gerne bis zu 60 Minuten Zeit.	Differenzierungskarten (70 – 78)
10 min	Präsentation I	Die Teams wählen ihren jeweiligen Lieblingsalgorithmus aus. Ein (oder zwei) Schüler:innen jedes Teams bleiben bei dem eigenem Codey Rocky, die anderen besuchen die anderen Teams: Zunächst wird den Besucher:innen das Struktogramm bzw. der Code gezeigt. Die Besucher:innen erklären den Entwickler:innen, was der Roboter ihrer Meinung nach machen wird. Danach wird die These gemeinsam überprüft.	Erstellte Struktogramme
10 min	Präsentation II	Die Schüler:innen eines Teams wechseln die Aufgaben. Bei größeren Teams und mehr Zeit können sie mehrere Durchgänge planen.	Erstellte Struktogramme

Arbeitsblätter



6.1 Codey Rocky kennen lernen

Heute geht es darum, die Bedienung des Roboters kennen zu lernen.

Dazu könnt Ihr Euch bei Eurer Lehrerin/Eurem Lehrer verschiedene Aufgaben holen, die in drei Schwierigkeitsstufen eingeteilt sind. Wenn Ihr eine Aufgabe erledigt habt, dann gebt sie wieder ab und holt Euch die nächste.

Jedes Team muss zuerst eine leichte Aufgabe () erfolgreich absolvieren, bevor es die nächste Stufe(mittel) beginnen kann. Hat das Team auch eine mittlere Aufgabe () gemeistert, könnt Ihr auch schwierige Aufgaben wählen ()

Ihr könnt aber auch so lange Aufgaben mit dem selben Schwierigkeitsgrad auswählen, bis Ihr Euch für die nächste Stufe sicher fühlt.

Wenn Ihr nicht sicher seid, ob Eure Lösung richtig ist, dann fragt Eure:n Lehrer:in.

Hier findet Ihr alle Aufgaben im Überblick:

Einfach

Aufgabe	Fertig
11 – Text	
12 – Aussehen	
13 – Sound	
14 – Bewegung	
15 – Bewegung	
16 – Bewegung	
17 – Aussehen	
18 – Aussehen	
19 – Aussehen	

Mittel

Aufgabe	Fertig
21 – gemischt	
22 – gemischt	
23 – Bewegung	
24 – Bewegung	
25 – Aussehen	
26 – gemischt	
27 – gemischt	
28 – gemischt	
29 – Bewegung	

Schwierig

Aufgabe	Fertig
31 – gemischt	
32 – Bewegung	
33 – Bewegung	
34 – Sensoren	
35 – Sensoren	
36 – Sensoren	
37 – Sound	
38 – Bewegung	
39 – Sound	

Hausaufgabe: Ordne die folgenden Bausteine in der Tabelle passend zu:

Sich drehen – Wenn Codey startet – Lautstärke – in einer Farbe leuchten – Sound ausgeben – Wenn Taste B gedrückt – Wenn Codey geschüttelt wird – Lichtintensität – vorwärtsbewegen – Text anzeigen

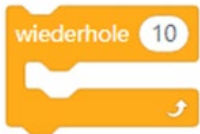
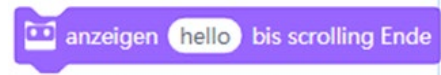
Ereignis	Input	Anweisung



Name:

Datum:

Ordne den Bausteinen die jeweils passende Kategorie zu:



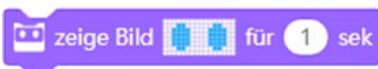
Aussehen



Beleuchtung



Aktion



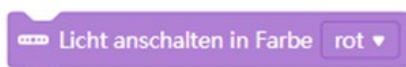
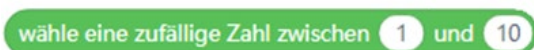
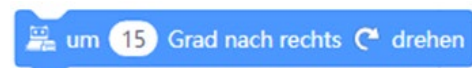
Infrarot



Steuerung



Operator





Name:

Datum:

Merksätze „algorithmische Grundbausteine“ (einfach)

Algorithmen

- 1** Fülle die Lücken mit den folgenden Begriffen: Aufgabe, Anweisungen, Problem, Algorithmus

Ein _____ ist eine Reihe von (Handlungs-) _____, die Schritt für Schritt ausgeführt werden, um ein _____ zu lösen oder eine _____ zu bewältigen.

Schleifen/ Wiederholung

Bestimmte Teile eines Algorithmus wiederholen sich häufig. Um Arbeit zu sparen, kann man Schleifen benutzen.

- 2** Formuliere aus den Satzteilen Sätze und notiere sie unten.
- a. alle wiederholt werden. | fasst man | In einer Schleife | Anweisungen | zusammen, die dann
 - b. Man kann beispielsweise | die Anzahl | der Wiederholungen | angeben oder | die Wiederholung | endlos | durchführen lassen.
 - c. so lange | Meistens | werden die Anweisungen | eine Bedingung an; | diese Bedingung | erfüllt ist, | ausgeführt. | gibt man

Struktogramm

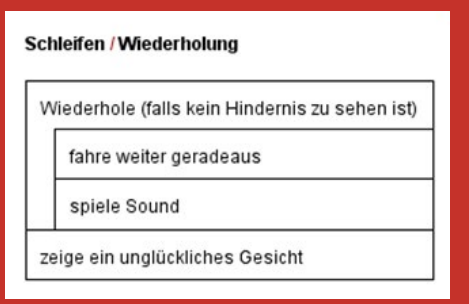


Abb 1: Struktogramm 1(Nassi-Shneiderman)
Die Bedingung für die Wiederholung hier ist „falls kein Hindernis zu sehen ist“.
Es werden zwei Bausteine wiederholt: „Fahre weiter geradeaus“ und „spiele Sound.“



Name:

Datum:

Bedingung und Verzweigung

Mit Hilfe von Verzweigungen wird eine bestimmte Codesequenz nur dann ausgeführt, wenn eine Bedingung erfüllt ist. Die Bedingung (oder Voraussetzung) wird beispielsweise mit „wenn...“ oder „if...“ eingeleitet. Die Verzweigung wird durch „dann...“ / „then...“ angezeigt und optional durch „sonst...“ oder „else ...“ fortgeführt.

- 3 Unterstreiche jeweils die Bedingung blau.
- 4 Unterstreiche die Handlungsanweisung nach der Verzweigung grün.
 - a. Wenn die Bedingung wahr ist, dann wird der Teil nach „dann...“ / „then...“ ausgeführt.
 - b. Wenn die Bedingung falsch ist, dann wird der Teil nach „sonst“ / „else...“ ausgeführt.
- 5 Bilde aus den Sätzen (4a und 4b) einen vollständigen Satz. Dieser Satz soll mit einer Bedingung anfangen und dann eine Verzweigung mit „dann“ und „sonst“ enthalten.

- 6 Unterstreiche bei dem Satz oben die Bedingung blau, die Anweisung, wenn die Bedingung wahr ist, grün und die Anweisung, wenn die Bedingung falsch ist, rot.

Strukto-
gramm

Abb. 2: Struktogramm 2 (Nassi-Shneiderman)

Bedingung:

Wenn das Jahr ein Schaltjahr ist...

Verzweigung:

wahr: dann ist die Anzahl der Tage 366

falsch: sonst ist die Anzahl der Tage 365.

- 7 Unterstreiche in denselben Farben alle Bedingungen und Verzweigungen auf dem ersten Arbeitsblatt (6.1 Den Codey Rocky kennen lernen).

Variablen

- 8 Fülle die Lücken unten aus. Nutze dafür die folgenden Begriffe: und, speichern, bezeichnen.

Wichtige Bausteine von Codes sind Variablen (auch Bezeichner genannt).

In Variablen kann man Daten (Zahlen, Texte, ...) _____. Diese Daten können sich ändern, sie sind also variabel.

Um diese Daten wiederfinden und benutzen zu können, sollte man sie stets gut _____.

Variablen haben also einen Namen (eine Bezeichnung) _____ einen Inhalt. Man kann sich als eine Art Kiste vorstellen: Außen steht der Name, innen ist der gespeicherte Wert.



Name: _____

Datum: _____

Merksätze „algorithmische Grundbausteine“ (mittel)

Algorithmen

- 1** Fülle die Lücken. Hinweis: An einer Stelle muss „Algorithmus“ stehen.

Ein _____ ist eine Reihe von (Handlungs-) _____, die Schritt für Schritt ausgeführt werden, um ein _____ zu lösen oder eine _____ zu bewältigen.

Schleifen/ Wiederholung

Bestimmte Teile eines Algorithmus wiederholen sich häufig. Um Arbeit zu sparen, kann man Schleifen benutzen.

- 2** Formuliere aus den Satzfragmenten Sätze und notiere sie unten.
Hinweis: Es fehlen jeweils Worte.

- a. wiederholt. | fasst | Schleife | Anweisungen | zusammen,
- b. kann | Anzahl | Wiederholungen | angeben oder | Wiederholung |
endlos | durchführen.
- c. so lange | Meistens | Anweisungen | Bedingung; | Bedingung | erfüllt, |
ausgeführt. | gibt

Struktogramm

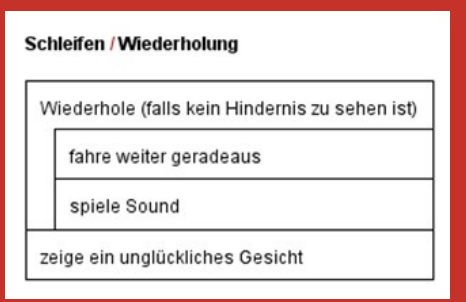


Abb 1: Struktogramm 1 (Nassi-Shneiderman)
Die Bedingung für die Wiederholung hier ist „falls kein Hindernis zu sehen ist“. Es werden zwei Bausteine wiederholt: „Fahre weiter geradeaus“ und „spiele Sound.“



Name: _____

Datum: _____

Bedingung und Verzweigung

Mit Hilfe von Verzweigungen wird eine bestimmte Codesequenz nur dann ausgeführt, wenn eine Bedingung erfüllt ist.

- a. Wenn die Bedingung wahr ist, dann wird der Teil nach „dann...“ / „then...“ ausgeführt.
- b. Wenn die Bedingung falsch ist, dann wird der Teil nach „sonst“ / „else...“ ausgeführt.

3 Fülle die Lücken unten aus.

Die Bedingung (oder Voraussetzung) wird beispielsweise mit „.....“ oder „if...“ eingeleitet. Die Verzweigung wird durch „.....“ oder „then...“ angezeigt und optional durch „sonst...“ oder „else...“ fortgeführt.

4 Unterstreiche bei a. und b. jeweils die Bedingung blau.

5 Unterstreiche bei a. und b. die Handlungsanweisung nach der Verzweigung grün.

6 Bilde aus den Sätzen (4a und 4b) einen vollständigen Satz. Dieser Satz soll mit einer Bedingung. anfangen und dann eine Verzweigung mit „dann“ und „sonst“ enthalten.

7 Unterstreiche bei dem Satz oben (6.) die Bedingung blau, die Anweisung, wenn die Bedingung wahr ist, grün und die Anweisung, wenn die Bedingung falsch ist, rot.

8 Unterstreiche in denselben Farben alle Bedingungen und Verzweigungen auf dem ersten Arbeitsblatt (6.1 Den Codey Rocky kennen lernen).

Strukto-
gramm



Abb. 2: Struktogramm 2 (Nassi-Shneiderman)

Bedingung:

Wenn das Jahr ein Schaltjahr ist...

Verzweigung:

wahr: dann ist die Anzahl der Tage 366

falsch: sonst ist die Anzahl der Tage 365.

Variablen

9 Fülle die Lücken unten aus.

Wichtige Bausteine von Codes sind Variablen (auch Bezeichner genannt).

In Variablen kann man Daten (Zahlen, Texte, ...) _____. Diese Daten können sich ändern, sie sind also variabel.

Um diese Daten wiederfinden und benutzen zu können, sollte man sie stets gut _____.

Variablen haben also einen Namen (eine Bezeichnung) _____ einen Inhalt. Man kann sich als eine Art Kiste vorstellen: Außen steht der Name, innen ist der gespeicherte Wert.



Name:

Datum:

Merksätze „algorithmische Grundbausteine“ (schwer)

Algorithmen

- 1** Erkläre den Begriff „Algorithmus“. Nutze dafür folgende Begriffe:
Schritt für Schritt, Problem, Anweisungen, Aufgabe.

Schleifen/ Wiederholung

Bestimmte Teile eines Algorithmus wiederholen sich häufig. Um Arbeit zu sparen, kann man Schleifen benutzen.

- 2** Formuliere aus den Satzfragmenten Sätze und notiere sie unten.
Hinweis: Es fehlen jeweils Worte.
- a. wiederholt. | fasst | Schleife | Anweisungen | zusammen,
b. kann | Anzahl | Wiederholungen | angeben oder | Wiederholung |
endlos | durchführen.
c. so lange | Meistens | Anweisungen | Bedingung; | Bedingung | erfüllt, |
ausgeführt. | gibt

Struktogramm

Schleifen / Wiederholung

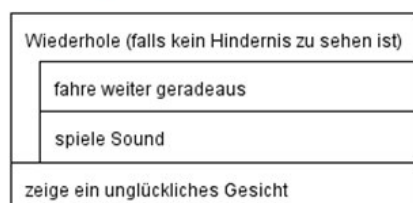


Abb 1: Struktogramm 1(Nassi-Shneiderman)
Die Bedingung für die Wiederholung hier ist „falls kein Hindernis zu sehen ist“.
Es werden zwei Bausteine wiederholt: „Fahre weiter geradeaus“ und „spiele Sound.“



Name:

Datum:

Bedingung und Verzweigung

Mit Hilfe von Verzweigungen wird eine bestimmte Codesequenz nur dann ausgeführt, wenn eine Bedingung erfüllt ist.

- a. Wenn die Bedingung wahr ist, dann wird der Teil nach „dann...“ / „then...“ ausgeführt.
- b. Wenn die Bedingung falsch ist, dann wird der Teil nach „sonst“ / „else...“ ausgeführt.

3 Fülle die Lücken unten aus.

Die Bedingung (oder Voraussetzung) wird beispielsweise mit „.....“ oder „if...“ eingeleitet. Die Verzweigung wird durch „.....“ oder „then...“ angezeigt und optional durch „sonst...“ oder „else...“ fortgeführt.

4 Unterstreiche bei a. und b. jeweils die Bedingung blau.

5 Unterstreiche bei a. und b. die Handlungsanweisung nach der Verzweigung grün.

6 Bilde aus den Sätzen (4a und 4b) einen vollständigen Satz. Dieser Satz soll mit einer Bedingung. anfangen und dann eine Verzweigung mit „dann“ und „sonst“ enthalten.

Wenn die Bedingung wahr ist, dann wird der Teil nach „dann“ ausgeführt, sonst wird der Teil nach „sonst“ ausgeführt.

7 Unterstreiche bei dem Satz oben (6.) die Bedingung blau, die Anweisung, wenn die Bedingung wahr ist, grün und die Anweisung, wenn die Bedingung falsch ist, rot.

8 Unterstreiche in denselben Farben alle Bedingungen und Verzweigungen auf dem ersten Arbeitsblatt (6.1 Den Codey Rocky kennen lernen).

Strukto-
gramm

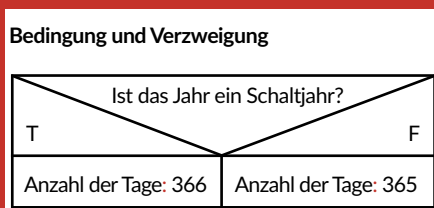


Abb. 2: Struktogramm 2 (Nassi-Shneiderman)

Bedingung:

Wenn das Jahr ein Schaltjahr ist...

Verzweigung:

wahr: dann ist die Anzahl der Tage 366

falsch: sonst ist die Anzahl der Tage 365.

Variablen

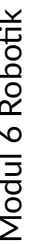
9 Fülle die Lücken unten aus.

Wichtige Bausteine von Codes sind Variablen (auch Bezeichner genannt).

In Variablen kann man Daten (Zahlen, Texte, ...) _____. Diese Daten können sich ändern, sie sind also variabel.

Um diese Daten wiederfinden und benutzen zu können, sollte man sie stets gut _____.

Variablen haben also einen Namen (eine Bezeichnung) _____ einen Inhalt. Man kann sich als eine Art Kiste vorstellen. Außen steht der Name, innen ist der gespeicherte Wert.



Setze Variable "Leben" auf 10	
Zeige "Team blau"	
Wenn Taste A gedrückt T F	
Wenn "Leben" = 0 T F mache ein trauriges Gesicht spiele ein trauriges Geräusch bis zum Ende stoppe alle Bewegungen stoppe alle Skripte Ø	
Wenn empfangen IR Nachricht "rot" T F verändere Variable "Leben" um - 2 zeige Variable "Leben" spiele ein wütendes Geräusch bis zum Ende Ø	
Wenn Hindernisse voraus T F sende IR Nachricht "blau" drehe 90° nach rechts fahre vorwärts	

Lehrmaterialien

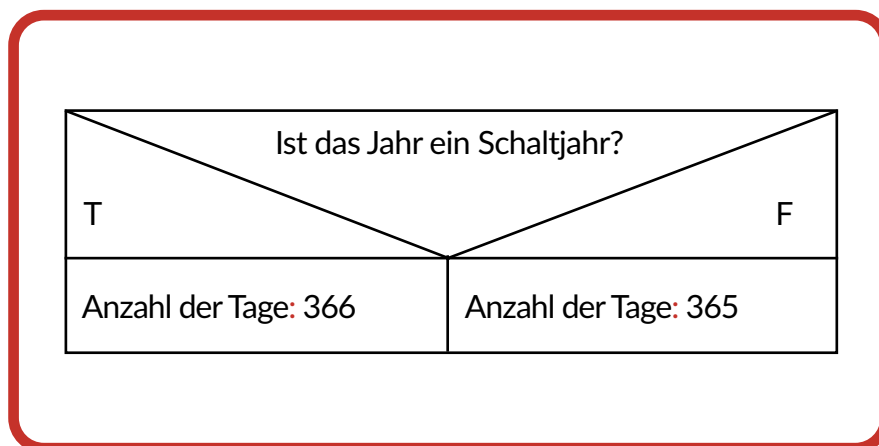
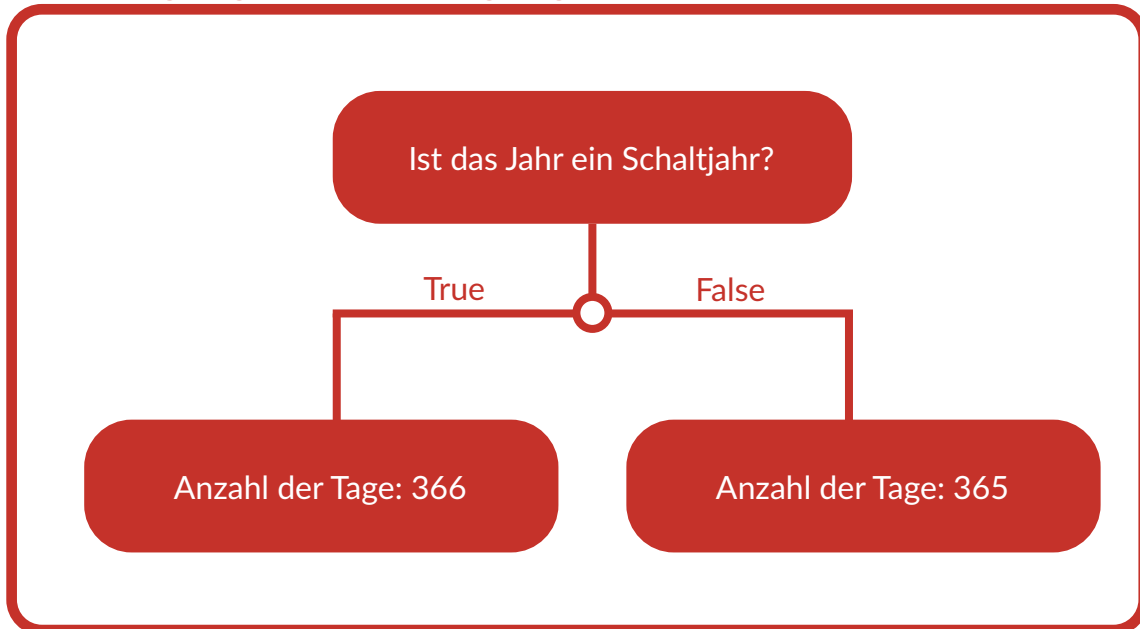


Name:

Datum:

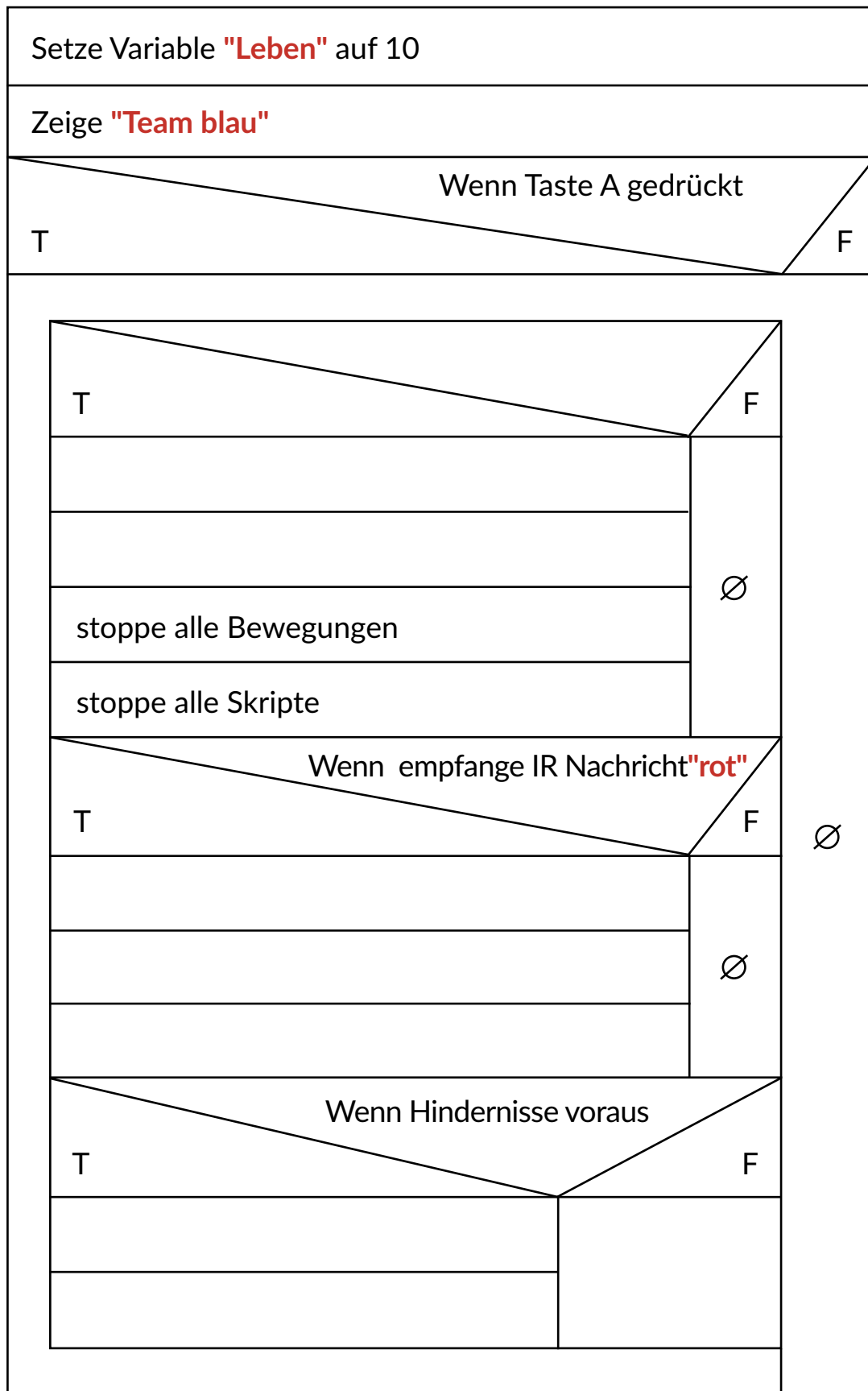
6.3 Einstiegsgrafik

Bedingung und Verzweigung





6.4 Laserduell Einstieg





6.4 Laserduell Einstieg

Drehe 90° nach rechts

spiele ein trauriges Geräusch bis zum Ende

Wenn "**Leben**" = 0

Spiele ein wütendes Geräusch bis zum Ende

Zeige Variable "**Leben**"

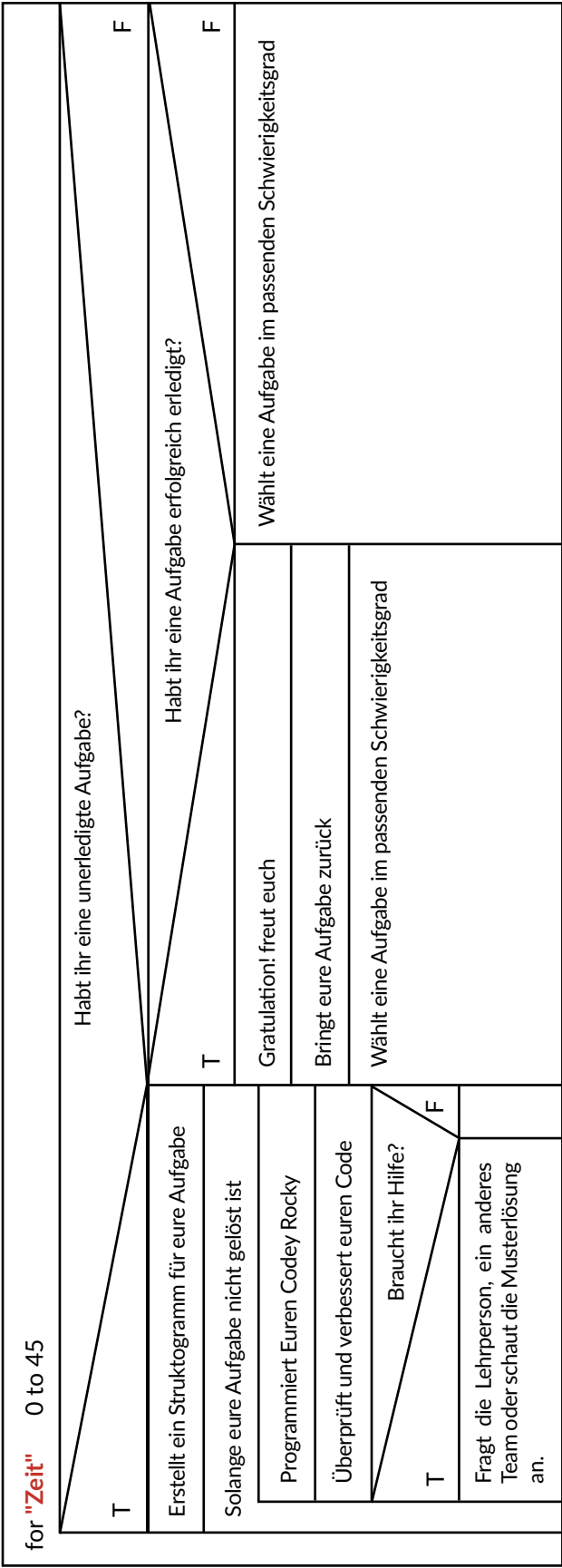
Sende IR Nachricht "**blau**"

Drehe 90° nach rechts

make ein trauriges Gesicht

fahre vorwärts

6.5 Wenn..., dann ... Einstiegsgrafik



Ein Projekt der

LFK • Die Medienanstalt für
Baden-Württemberg

Gefördert durch das



Baden-Württemberg

MINISTERIUM FÜR KULTUS, JUGEND UND SPORT



In Kooperation mit

