Offen lizenzierte Lehr- und Lernunterlagen für den Informatikunterricht (2)

Technische Universität Graz

Lizenzvereinbarung

Frei lizenzierte Lehr- und Lernunterlagen für den Informatikunterricht
von *Technischer Universität Graz* (Administration *Maria Grandl)* ist lizenziert unter einer
Creative Commons Namensnennung 4.0 International Lizenz.

Die verwendeten Abbildungen sind,
sofern keine entsprechende Kennzeichnung vorhanden ist,
frei von Urheberrechten oder wurden als solche identifiziert.

Mehr Informationen zu den Creative Common Lizenzen
finden Sie unter **https://creativecommons.org/**

Computational Thinking

Computational Thinking

Unter Computational Thinking, übersetzt informatisches Denken, wird nach Jeannette Wing, welche diesen Begriff maßgeblich prägte, ein 3-stufiger gedanklicher Prozess verstanden, bei dem es darum geht, ein Problem so zu betrachten und die Lösungsschritte so zu formulieren, dass die zur Lösung notwendigen Schritte von einem informationsverarbeitenden System ausgeführt werden können. Die Schüler/innen sollen lernen, ein Problem zu abstrahieren und in kleinere Elemente zu zerlegen, Muster zu erkennen, einen Algorithmus zu entwickeln und diesen zu testen und zu verbessern. Unter einem Algorithmus wird in diesem Sinne eine eindeutig festgelegte Abfolge von endlich vielen Anweisungen zur Lösung des Problems, das aus unterschiedlichen Lebensbereichen stammen kann, verstanden.

Sekundarstufe 1:

In Hinsicht auf den Lehrplan für die verbindliche Übung **Digitale Grundbildung** soll dieser Themenbereich die zum **Kompetenzfeld Computational Thinking** gehörigen **Kompetenzbereiche**

- **Mit Algorithmen arbeiten** und
- **Kreative Nutzung von Programmiersprachen**

umfassen.

Sekundarstufe 2:

In Hinsicht auf das **digi.komp12-Kompetenzmodell** soll dieser Themenbereich die zum **Kompetenzfeld Praktische Informatik** gehörigen **Kompetenzbereiche**

- **Konzepte der Informationsverarbeitung** und
- **Algorithmen, Datenstrukturen und Programmierung**

umfassen.

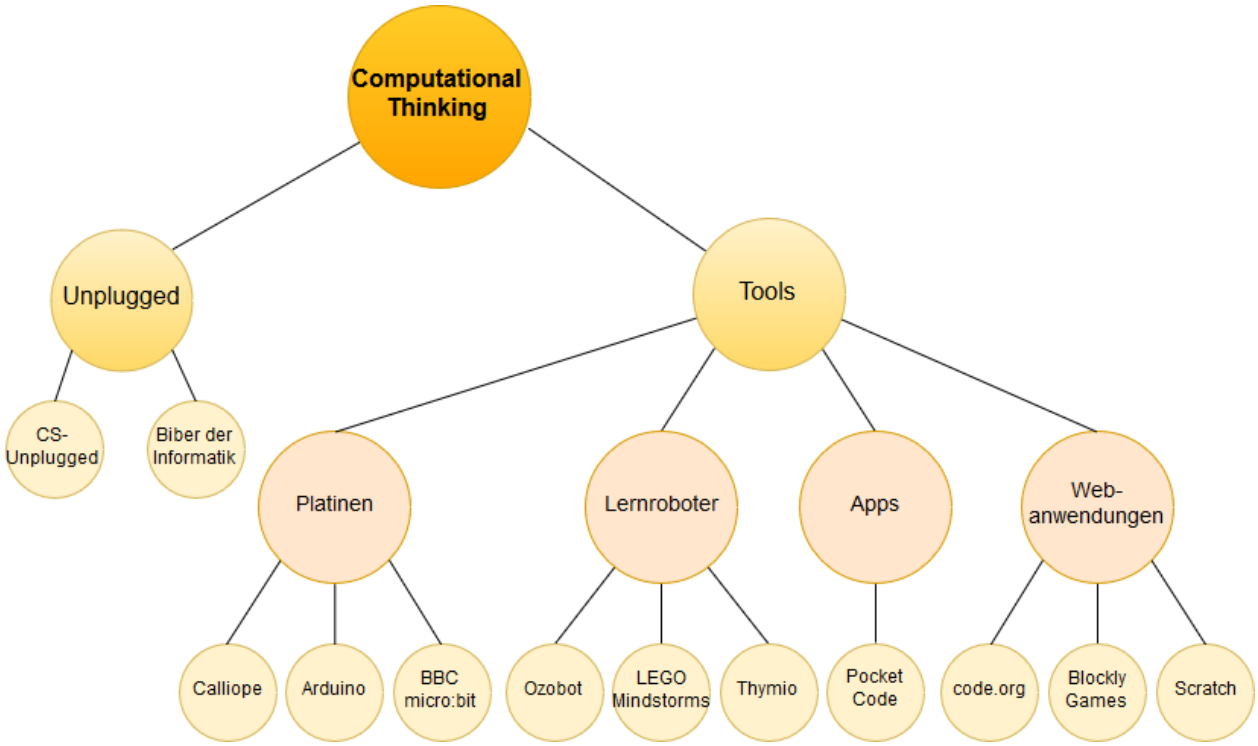
Aufbau

Auf den folgenden Seiten werden verschiedene Methoden und Hilfsmittel vorgestellt, welche die Schüler/innen auf unterschiedliche Art und Weise in das Thema Computational Thinking einführen. Dabei lassen sich diese in zwei Kategorien aufteilen: „Unplugged" und „Tools".

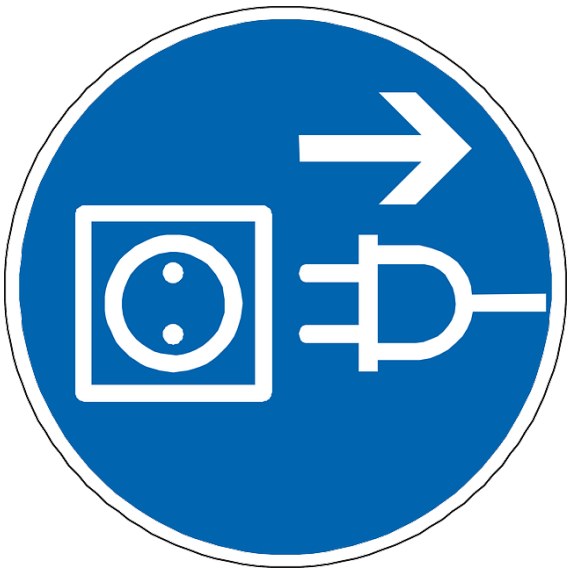
Unter „Unplugged" wird eine Herangehensweise ohne technische Hilfsmittel verstanden. Mit Stift und Papier werden Konzepte verständlich erklärt, ganz ohne Hilfe von Computern. Dazu stellen wir zum einen den Wettbewerb „Biber der Informatik" vor und zum anderen die Website „CS unplugged – Computer Science without a computer". Beide Initiativen stellen Aufgaben zur Verfügung, welche die Entwicklung von Problemlösekompetenzen unterstützen und wertvolle Konzepte der Informatik auf spielerische Weise zugänglich machen.

Bei der Kategorie „Tools" werden verschiedene Webanwendungen, Lernroboter, Platinen und Apps vorgestellt, die sich besonders gut dafür eignen, grundlegende Konzepte der Programmierung kennenzulernen und in verschiedenen Kontexten und Unterrichtsfächern anzuwenden.

Aufbau



Unplugged



Biber der Informatik

Der sogenannte *Biber der Informatik* ist ein Informatikwettbewerb zum Erlernen und Üben des informatischen Denkens, der von der österreichischen Computer Gesellschaft (OCG) im Jahr 2016 bereits zum zwölften Mal durchgeführt wird. Die Teilnahme ist von der 3. bis zur 13. Schulstufe möglich. Der Wettbewerb findet jährlich im November statt und kann in 5 Altersgruppen (3.-4., 5.-6., 7.-8., 9.-10. und 11.-13. Schulstufe) online durchgeführt werden. Beim Wettbewerb werden in der Sekundarstufe 2 jeweils 15 Aufgaben, welche in 3 Schwierigkeitsstufen unterteilt sind, gestellt. Diese müssen innerhalb von 40 Minuten bearbeitet werden.

Logo des Biber der Informatik
(**https://www.ocg.at/sites/ocg.at/files/styles/large/public/media/button-biber.jpg?itok=rgghNWJp**)

Die Ideen zu den Aufgaben stammen von Unterstützer(n)/innen der internationalen Initiative **Bebras** (International Challenge on Informatics and Computational Thinking), welche auch die nötige Infrastruktur zur Durchführung des Wettbewerbs bereitstellt. Die Aufgaben setzen keine informatischen Vorkenntnisse voraus und präsentieren vielfältige Problemstellungen, die durch das Anwenden von informatischen Konzepten gelöst werden müssen. Diese Konzepte müssen jedoch nicht bekannt sein, sondern werden im Rahmen der Aufgaben spielerisch erworben. Im Vordergrund steht der Erwerb und weniger die Überprüfung von Fähigkeiten im Zusammenhang mit Computational Thinking.
Auf der Website der OCG werden die Aufgaben für alle Altersgruppen gesammelt und durch wichtige Erklärungen ergänzt, sodass die informatischen Konzepte auch in das Bewusstsein der Schüler/innen geraten. Die Aufgabenhefte sind unter der Creative-Commons-Lizenz CC BY veröffentlicht.
Startseite: **http://www.ocg.at/de/biber-der-informatik**
Aufgabenhefte: **http://www.ocg.at/node/269**

Biber der Informatik

Exemplarische Aufgaben für die 9. Schulstufe

Aufgabenheft Biber der Informatik **2016**:
Cassy, die Schildkröte: Programmieren, Korrektheit, Anweisungsblock, Schleifen
Karten und Kegel: Programmieren, Variable, Zuweisung
Spielernummern: Algorithmus, Suchen, binäre Suche

Geheime Nachrichten: Kryptologie, Datenschutz
IP-Adressen: Netzwerke, IP, IP-Adressierung

Aufgabenheft Biber der Informatik **2015**:
Bühnenlicht: Modellierung, Farbinformation, RGB
Alea iacta: Programmieren, IF-THEN-ELSE, CASE
Schüsselfabrik: Algorithmus, Sortieren, Bubble-Sort
Fleißiger Biber: Modellierung, Turingmaschinen

Schnäppchen: Datensicherheit, Identität, Internet-Geschäfte
Weitergeben erlaubt?: Benutzerrecht, Creative Commons, Ethik

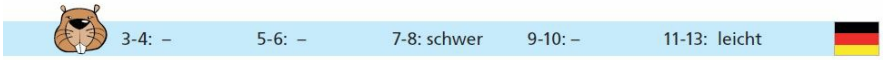
Biber der Informatik - Beispiel 1

Cassy, die Schildkröte

Beispiel für die 9. Schulstufe aus dem **Aufgabenheft Biber der Informatik 2016, Seite 19**:

Stichwörter:

- Programmieren
- Korrektheit
- Anweisungsblock
- Schleifen

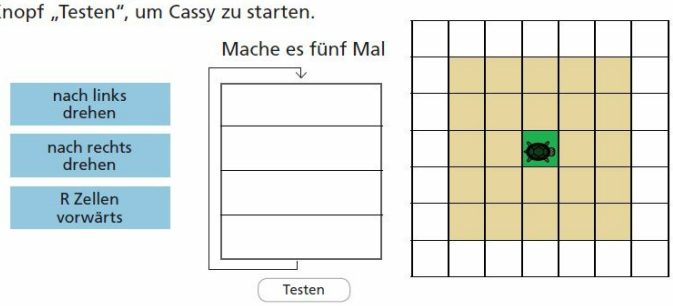


Cassy, die Schildkröte

Die Schildkröte Cassy lebt in Gitterland, auf einem Acker von fünf mal fünf Gitterzellen.
Sie isst für ihr Leben gern frische Salatpflanzen. Jeden Morgen wachsen neue Salatpflanzen.
Cassy weiß nicht, in welchen Gitterzellen sie sind, aber sie will alle essen!
Cassy startet jeden Tag in der Mitte des Ackers und will den ganzen Acker ablaufen.

Sage Cassy, wie sie sich bewegen soll!
Neben dem Gitterland ist ein Anweisungsblock mit vier Feldern. Fülle jedes Feld mit einer Bewegungs-Anweisung für Cassy, so dass sie insgesamt den ganzen Acker abläuft:

- Wähle durch Klicken die passenden Anweisungen aus.
- Eine Anweisung kann mehr als einmal verwendet werden.
- R ist ein Wiederholungszähler.
- Wenn der Anweisungsblock das erste Mal ausgeführt wird, enthält R den Wert R = 1 ; bei der zweiten Ausführung wird R = 2 ; usw.
- Cassy darf den Acker verlassen, aber nicht das Gitterland.
- Klicke auf den Knopf „Testen", um Cassy zu starten.



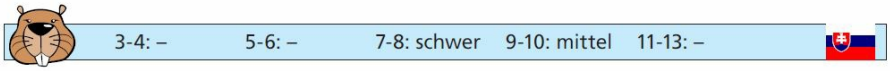
Biber der Informatik - Beispiel 2

Bühnenlicht

Beispiel für die 9. Schulstufe aus dem **Aufgabenheft Biber der Informatik 2015, Seite 16**:

Stichwörter:

- Modellierung
- Farbinformation
- RGB



Bühnenlicht

Drei Scheinwerfer beleuchten die Bühne. Einer strahlt rot, einer grün und einer blau.
Die Farbe des Bühnenlichts mischt sich aus den Farben der Scheinwerfer, die gerade eingeschaltet sind. Die Tabelle zeigt alle möglichen Farbmischungen:

Bühnenlicht	Schwarz	Blau	Grün	Cyan	Rot	Magenta	Gelb	Weiß
rotes Licht	aus	aus	aus	aus	an	an	an	an
grünes Licht	aus	aus	an	an	aus	aus	an	an
blaues Licht	aus	an	aus	an	aus	an	aus	an

Sobald die Vorstellung beginnt, wird jeder Scheinwerfer in einem eigenen Rhythmus ein- und ausgeschaltet:
Der rote Scheinwerfer strahlt im Rhythmus „zwei Minuten aus, zwei Minuten an".
Der grüne Scheinwerfer strahlt im Rhythmus „eine Minute aus, eine Minute an".
Der blaue Scheinwerfer strahlt im Rhythmus „vier Minuten an, vier Minuten aus".

Welche Farben hat das Bühnenlicht in den ersten vier Minuten der Vorstellung?
Schiebe die richtigen Farben über die Minuten:

CS - Unplugged

CS Unplugged - Computer Science without a computer

Die Website CS Unplugged - Computer Science without a computer bietet viele Ideen und Materialien für Aktivitäten, die einen spielerischen und spannenden Zugang zu informatischen Konzepten wie Algorithmen, Programmiersprachen und Graphen ermöglichen.

Logo der Seite CS Unplugged
(**https://classic.csunplugged.org/wp-content/uploads/2015/02/logo-330h-880w.png**)

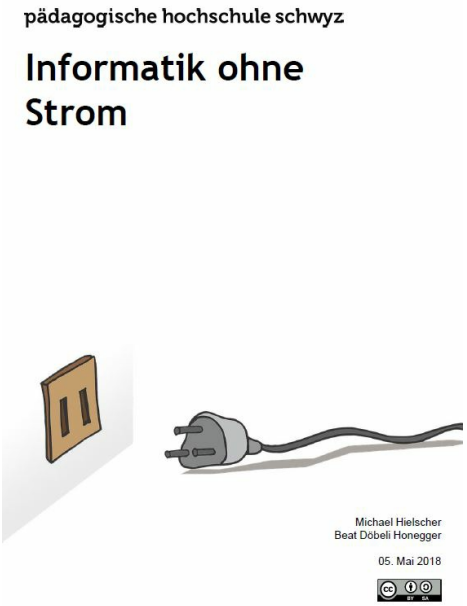
Diese sollen den Schüler(n)/innen vermitteln, dass die Informatik ein spannendes und intellektuell herausforderndes Fachgebiet darstellt, das sich nicht ausschließlich auf ein technisches Gerät, den Computer, bezieht. Die Aktivitäten schulen das informatische Denken und sind teilweise sogar für den Einsatz in der Primarstufe geeignet. Alle Materialien werden unter der Creative-Commons-Lizenz CC BY zur Verfügung gestellt.

Link: **http://csunplugged.org/**

Informatik ohne Strom

Informatik ohne Strom

"Informatik ohne Strom" lautet der Titel einer Broschüre mit 14 Beispielen, die teilweise aufeinander aufbauen. Die Unterrichtsideen eignen sich als kreativer Einstieg in ein neues Thema oder zur Unterstützung der Verständlichkeit von informatischen Konzepten. Die Beispiele beanspruchen zwischen 10 und 40 Minuten und der Vorbereitungsaufwand ist meist sehr gering. Die Broschüre ist unter der Creative-Commons-Lizenz CC BY SA frei zur Verfügung gestellt.

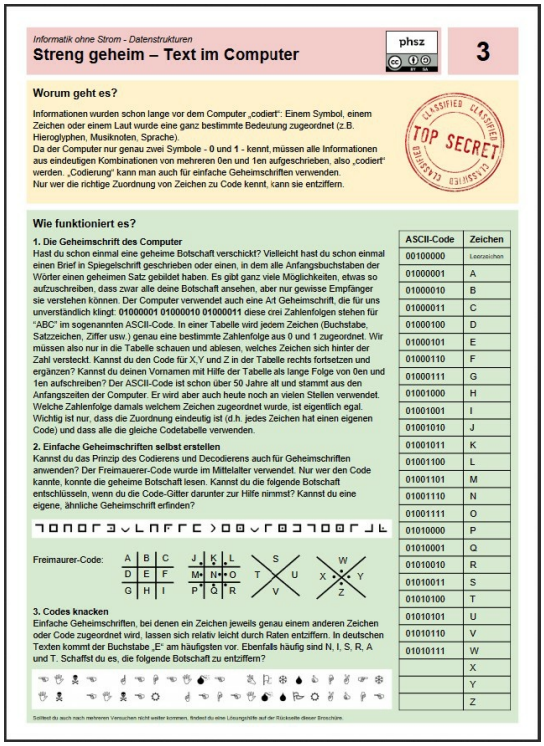


Links:
Website: **http://ilearnit.ch/de/stromlos.html**
Broschüre:
http://ilearnit.ch/download/InformatikohneStrom.pdf

Informatik ohne Strom - Beispiel 1

Streng geheim - Text im Computer

Bei dem Beispiel "Streng geheim" geht es um das Thema Codierung. Es werden der ASCII-Code und der Freimaurer-Code behandelt und dazu gibt es kurze Übungen. Diese Idee eignet sich gut als Einstieg in das Thema Kryptographie.



Informatik ohne Strom - Datenstrukturen
Streng geheim – Text im Computer phsz 3

Worum geht es?

Informationen wurden schon lange vor dem Computer „codiert": Einem Symbol, einem Zeichen oder einem Laut wurde eine ganz bestimmte Bedeurung zugeordnet (z.B. Hieroglyphen, Musiknoten, Sprache).
Da der Computer nur genau zwei Symbole - **0** und **1** - kennt, müssen alle Informationen aus eindeutigen Kombinationen von mehreren 0en und 1en aufgeschrieben, also „codiert" werden. „Codierung" kann man auch für einfache Geheimschriften verwenden.
Nur wer die richtige Zuordnung von Zeichen zu Code kennt, kann sie entziffern.

TOP SECRET

Wie funktioniert es?

1. Die Geheimschrift des Computer
Hast du schon einmal eine geheime Botschaft verschickt? Vielleicht hast du schon einmal einen Brief in Spiegelschrift geschrieben oder einen, in dem alle Anfangsbuchstaben der Wörter einen geheimen Satz gebildet haben. Es gibt ganz viele Möglichkeiten, etwas so aufzuschreiben, dass zwar alle deine Botschaft ansehen, aber nur gewisse Empfänger sie verstehen können. Der Computer verwendet auch eine Art Geheimschrift, die für uns unverständlich klingt: **01000001 01000010 01000011** diese drei Zahlenfolgen stehen für "ABC" im sogenannten ASCII-Code. In einer Tabelle wird jedem Zeichen (Buchstabe, Satzzeichen, Ziffer usw.) genau eine bestimmte Zahlenfolge aus 0 und 1 zugeordnet. Wir müssen also nur in die Tabelle schauen und ablesen, welches Zeichen sich hinter der Zahl versteckt. Kannst du den Code für X,Y und Z in der Tabelle rechts fortsetzen und ergänzen? Kannst du deinen Vornamen mit Hilfe der Tabelle als lange Folge von 0en und 1en aufschreiben? Der ASCII-Code ist schon über 50 Jahre alt und stammt aus den Anfangszeiten der Computer. Er wird aber auch heute noch an vielen Stellen verwendet. Welche Zahlenfolge damals welchem Zeichen zugeordnet wurde, ist eigentlich egal. Wichtig ist nur, dass die Zuordnung eindeutig ist (d.h. jedes Zeichen hat einen eigenen Code) und dass alle die gleiche Codetabelle verwenden.

ASCII-Code	Zeichen
00100000	Leerzeichen
01000001	A
01000010	B
01000011	C
01000100	D
01000101	E
01000110	F
01000111	G
01001000	H
01001001	I
01001010	J
01001011	K
01001100	L
01001101	M
01001110	N
01001111	O
01010000	P
01010001	Q
01010010	R
01010011	S
01010100	T
01010101	U
01010110	V
01010111	W
	X
	Y
	Z

2. Einfache Geheimschriften selbst erstellen
Kannst du das Prinzip des Codierens und Decodierens auch für Geheimschriften anwenden? Der Freimaurer-Code wurde im Mittelalter verwendet. Nur wer den Code kannte, konnte die geheime Botschaft lesen. Kannst du die folgende Botschaft entschlüsseln, wenn du die Code-Gitter darunter zur Hilfe nimmst? Kannst du eine eigene, ähnliche Geheimschrift erfinden?

Freimaurer-Code:

3. Codes knacken
Einfache Geheimschriften, bei denen ein Zeichen jeweils genau einem anderen Zeichen oder Code zugeordnet wird, lassen sich relativ leicht durch Raten entziffern. In deutschen Texten kommt der Buchstabe „E" am häufigsten vor. Ebenfalls häufig sind N, I, S, R, A und T. Schaffst du es, die folgende Botschaft zu entziffern?

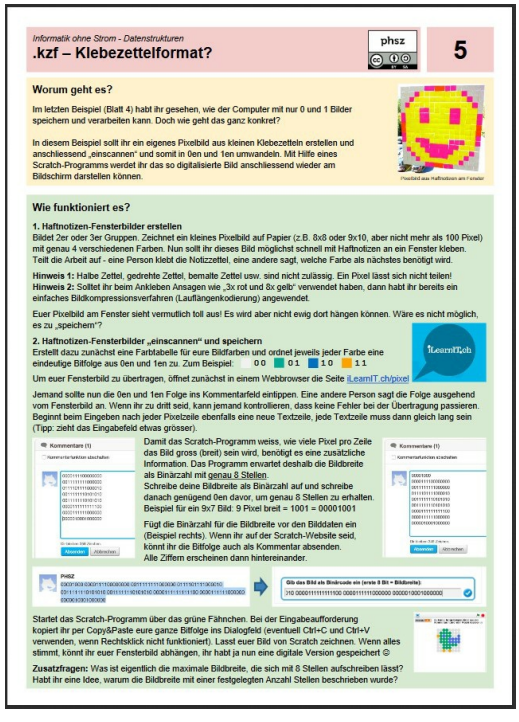
Sollte du auch nach mehreren Versuchen nicht weiter kommen, findest du eine Lösungshilfe auf der Rückseite dieser Broschüre.

Broschüre "Informatik ohne Strom", Seite 4

Informatik ohne Strom - Beispiel 2

.kzf - Klebezettelformat?

Das Beispiel ".kzf - Klebezettelformat?" behandelt das Thema Speicherformate und wie eine Datei abgespeichert wird, um sie eins zu eins wiedergeben zu können. Um es den Schüler(n)/innen zu veranschaulichen, wird das Klebezettelformat eingeführt, bei dem es darum geht, ein zweifarbiges Bild aus Klebezetteln zu kreieren, mit Nullen und Einsen händisch abzuspeichern und mit Hilfe von Scratch am Computer wiederzugeben.



Informatik ohne Strom - Datenstrukturen
.kzf – Klebezettelformat? phsz 5

Worum geht es?

Im letzten Beispiel (Blatt 4) habt ihr gesehen, wie der Computer mit nur 0 und 1 Bilder speichern und verarbeiten kann. Doch wie geht das ganz konkret?

In diesem Beispiel sollt ihr ein eigenes Pixelbild aus kleinen Klebezetteln erstellen und anschliessend „einscannen" und somit in 0en und 1en umwandeln. Mit Hilfe eines Scratch-Programms werdet ihr das so digitalisierte Bild anschliessend wieder am Bildschirm darstellen können.

Pixelbild aus Haftnotizen am Fenster

Wie funktioniert es?

1. Haftnotizen-Fensterbilder erstellen
Bildet 2er oder 3er Gruppen. Zeichnet ein kleines Pixelbild auf Papier (z.B. 8x8 oder 9x10, aber nicht mehr als 100 Pixel) mit genau 4 verschiedenen Farben. Nun sollt ihr dieses Bild möglichst schnell mit Haftnotizen an ein Fenster kleben. Teilt die Arbeit auf - eine Person klebt die Notizzettel, eine andere sagt, welche Farbe als nächstes benötigt wird.

Hinweis 1: Halbe Zettel, gedrehte Zettel, bemalte Zettel usw. sind nicht zulässig. Ein Pixel lässt sich nicht teilen!
Hinweis 2: Solltet ihr beim Ankleben Ansagen wie „3x rot und 8x gelb" verwendet haben, dann habt ihr bereits ein einfaches Bildkompressionsverfahren (Lauflängenkodierung) angewendet.

Euer Pixelbild am Fenster sieht vermutlich toll aus! Es wird aber nicht ewig dort hängen können. Wäre es nicht möglich, es zu „speichern"?

2. Haftnotizen-Fensterbilder „einscannen" und speichern
Erstellt dazu zunächst eine Farbtabelle für eure Bildfarben und ordnet jeweils jeder Farbe eine eindeutige Bitfolge aus 0en und 1en zu. Zum Beispiel: 00 01 10 11

Um euer Fensterbild zu übertragen, öffnet zunächst in einem Webbrowser die Seite iLearnIT.ch/pixel

Jemand sollte nun die 0en und 1en Folge ins Kommentarfeld eintippen. Eine andere Person sagt die Folge ausgehend vom Fensterbild an. Wenn ihr zu dritt seid, kann jemand kontrollieren, dass keine Fehler bei der Übertragung passieren. Beginnt beim Eingeben nach jeder Pixelzeile ebenfalls eine neue Textzeile, jede Textzeile muss dann gleich lang sein (Tipp: zieht das Eingabefeld etwas grösser).

Damit das Scratch-Programm weiss, wie viele Pixel pro Zeile das Bild gross (breit) sein wird, benötigt es eine zusätzliche Information. Das Programm erwartet deshalb die Bildbreite als Binärzahl mit genau 8 Stellen.
Schreibe deine Bildbreite als Binärzahl auf und schreibe danach genügend 0en davor, um genau 8 Stellen zu erhalten.
Beispiel für ein 9x7 Bild: 9 Pixel breit = 1001 = 00001001

Fügt die Binärzahl für die Bildbreite vor den Bilddaten ein (Beispiel rechts). Wenn ihr auf der Scratch-Website seid, könnt ihr die Bitfolge auch als Kommentar absenden.
Alle Ziffern erscheinen dann hintereinander.

Startet das Scratch-Programm über das grüne Fähnchen. Bei der Eingabeaufforderung kopiert ihr per Copy&Paste eure ganze Bitfolge ins Dialogfeld (eventuell Ctrl+C und Ctrl+V verwenden, wenn Rechtsklick nicht funktioniert). Lasst euer Bild von Scratch zeichnen. Wenn alles stimmt, könnt ihr euer Fensterbild abhängen, ihr habt ja nun eine digitale Version gespeichert ☺

Zusatzfragen: Was ist eigentlich die maximale Bildbreite, die sich mit 8 Stellen aufschreiben lässt? Habt ihr eine Idee, warum die Bildbreite mit einer festgelegten Anzahl Stellen beschrieben wurde?

Broschüre "Informatik ohne Strom", Seite 6

Weitere Links

Experimente ohne Computer

Auf dieser Seite finden sich 13 Unplugged-Beispiele - unter anderem zu den Themen Binärzahlen, Sortieralgorithmen und Deadlock. Diese meist kurzen Experimente (zwischen 10 und 30 Minuten) bieten sich vor allem als Einstieg in ein neues Kapitel an.

Link:
https://www.swisseduc.ch/informatik/theoretische_informatik/paper_compute

Informatik ohne Stecker

Diese Website bietet 5 Module zu den Themen Daten, Algorithmen, Kontrollfluss, Netzwerke und Sicherheit an. Es handelt sich dabei um durchgeplante Unterrichtseinheiten, welche sich über mehrere Stunden erstrecken. Es wird auf Materialien von verschiedenen Webseiten (u.a. csunplugged.org) zurückgegriffen, welche alle ins Deutsche übersetzt wurden.

Link:**https://troeger.eu/unplugged**

Tools

Webanwendungen

Webanwendungen

Die Erstellung eines Programmes fordert und fördert Computational Thinking. Das trifft insbesondere dann zu, wenn die Schüler/innen im Entwicklungsprozess ihre individuellen Ideen und Präferenzen einbringen können.

Auf den folgenden Seiten werden drei verschiedene Webanwendungen vorgestellt, welche sich sehr gut als Einstieg in die Programmierung eignen. Alle drei Webanwendungen beruhen auf der Programmierung mit grafischen Programmbausteinen, die aneinander gereiht werden, womit sich schnell Erfolge bei den Schüler(n)/innen einstellt und die Motivation steigert.

Scratch

Scratch ermöglicht einen einfachen und interessanten Einstieg in die Programmierung. Mithilfe von Befehlsbausteinen, welche unterschiedliche Kategorien wie Steuerung, Ereignisse oder Bewegung zugeordnet sind und sich je nach Kategorie farblich voneinander unterscheiden, können einfache Projekte schnell umgesetzt werden. Erfolgserlebnisse sind daher gleich zu Beginn möglich. Die fertigen Programme können anschließend veröffentlicht und geteilt werden.
Mit Scratch lassen sich einfache Programme schnell und intuitiv erstellen, aber auch komplexere Programme sind mit der Software möglich und so werden auch bereits fortgeschrittene Scratcher herausgefordert.

Offizielle Website: **https://scratch.mit.edu/**

Logo der Programmiersprache Scratch:
https://en.scratch-wiki.info/wiki/File:Logo.png

Scratch

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/98awWpkx9UM

Scratch Junior

Die App ScratchJr, für Kinder im Alter von 5-7 Jahren, ist eine altersgerechte Version von Scratch. Kinder können Figuren im Maleditor bearbeiten, Stimmen und Geräusche hinzufügen, eigene Fotos hochladen und mit Hilfe der grafischen Programmblöcke zum Hüpfen, Tanzen und Singen bringen. Die App ist für die Betriebssysteme Android und iOS verfügbar.

Benutzeroberfläche von ScratchJr
(**https://www.scratchjr.org/images/tips/character-list.png**)

Offizielle Website: **https://www.scratchjr.org/**
Download für Android: **https://play.google.com/store/apps/details?id=org.scratchjr.android**
Download für iOS: **https://itunes.apple.com/us/app/scratchjr/id895485086?ls=1&mt=8**

Scratch

Scratch 3.0

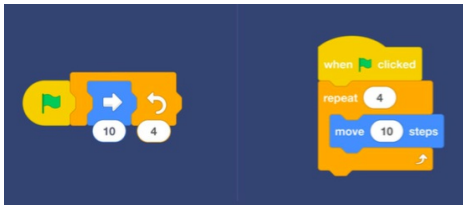
Am 2. Jänner 2019 wurde Scratch 3.0 veröffentlicht, was einige Änderungen mit sich brachte. Unter anderem:

- vertikales Block-Layout
- manche Blöcke haben andere Farben
- Blöcke befinden sich in einer anderen Reihenfolge
- Standardeingaben wurden geändert
- viele Blöcke wurden mit zusätzlichen Funktionen erweitert
- manche Blöcke sind weggefallen
- Internet Explorer wird nicht mehr unterstützt
- BBC micro:bit und LEGO Mindstorms EV3 Erweiterung

Alle Änderungen sind im Scratch-Wiki nachzulesen.

Bei den Unterrichtsmaterialien ist jeweils angegeben, um welche Scratch Version es sich handelt.

Link zum Wiki:
https://scratch-dach.info/wiki/Scratch_3.0



Vgl. horizontales/vertikales Blocklayout: https://scratch-

Links und Materialien

Offizielle Website

Die offizielle Website bietet einen eigenen Bereich für Lehrkräfte, welcher eine Vielzahl an Tutorials und Projektideen beinhaltet.

Link:**https://scratch.mit.edu/**

Einführung in 10 Sequenzen

Die Website swisseduc.ch bietet Unterrichtssequenzen zum Einstieg in die Programmiersprache Scratch. Es gibt 9 Grundübungen, die jeweils zwischen 15 und 60 Minuten dauern, bei denen grundlegende Konzepte und Abläufe erarbeitet werden. Die erlernten Konzepte sollen in einem eigenen Abschlussprojekt, für welches 90 Minuten Zeit eingeplant ist, angewandt werden.
Auf dieser Seite finden sich auch noch drei weitere, etwas komplexere Projektideen: Labyrinth, Fang mich und Breakout. Jeweils mit Vorlage und Musterlösung.

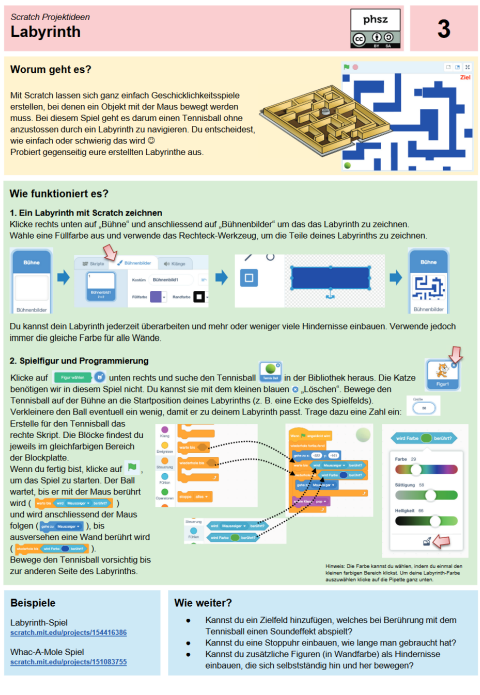
in den Unterlagen werden die Scratch Versionen 1.0 und 2.0 verwendet!

Link:

Links und Materialien

Broschüre "Projektideen"

Die nachfolgende Broschüre beinhaltet sieben verschiedene Projektideen. Die jeweilige Kernidee wird durch Skript-Blöcke dargestellt, somit sind erste Erfolge in der Umsetzung garantiert. Es wird aber genug Freiraum gelassen, um es zu einem ganz individuellen Projekt weiterführen zu können und seine eigenen Ideen einfließen zu lassen. Die Beispiele eignen sich gut, um grundlegende Konzepte, welche bereits erarbeitet wurden, zu festigen.


Scratch Projektideen
Labyrinth
phsz
3
Worum geht es?
Mit Scratch lassen sich ganz einfach Geschicklichkeitsspiele erstellen, bei denen ein Objekt mit der Maus bewegt werden muss. Bei diesem Spiel geht es darum einen Tennisball ohne anzustossen durch ein Labyrinth zu navigieren. Du entscheidest, wie einfach oder schwierig das wird ☺
Probiert gegenseitig eure erstellten Labyrinthe aus.
Wie funktioniert es?
1. Ein Labyrinth mit Scratch zeichnen
Klicke rechts unten auf „Bühne" und anschliessend auf „Bühnenbilder" um das das Labyrinth zu zeichnen.
Wähle eine Füllfarbe aus und verwende das Rechteck-Werkzeug, um die Teile deines Labyrinths zu zeichnen.
Du kannst dein Labyrinth jederzeit überarbeiten und mehr oder weniger viele Hindernisse einbauen. Verwende jedoch immer die gleiche Farbe für alle Wände.
2. Spielfigur und Programmierung
Klicke auf unten rechts und suche den Tennisball in der Bibliothek heraus. Die Katze benötigen wir in diesem Spiel nicht. Du kannst sie mit dem kleinen blauen „Löschen". Bewege den Tennisball auf der Bühne an die Startposition deines Labyrinths (z. B. eine Ecke des Spielfelds).
Verkleinere den Ball eventuell ein wenig, damit er zu deinem Labyrinth passt. Trage dazu eine Zahl ein:
Erstelle für den Tennisball das rechte Skript. Die Blöcke findest du jeweils im gleichfarbigen Bereich der Blockpalette.
Wenn du fertig bist, klicke auf um das Spiel zu starten. Der Ball wartet, bis er mit der Maus berührt wird ( ) und wird anschliessend der Maus folgen ( ), bis ausversehen eine Wand berührt wird ( )
Bewege den Tennisball vorsichtig bis zur anderen Seite des Labyrinths.
Hinweis: Die Farbe kannst du wählen, indem du einmal den kleinen farbigen Bereich klickst. Um deine Labyrinth-Farbe auszuwählen klicke auf die Palette ganz unten.
Beispiele
Labyrinth-Spiel
scratch.mit.edu/projects/154416386
Whac-A-Mole Spiel
scratch.mit.edu/projects/151083755
Wie weiter?
- Kannst du ein Zielfeld hinzufügen, welches bei Berührung mit dem Tennisball einen Soundeffekt abspielt?
- Kannst du eine Stoppuhr einbauen, wie lange man gebraucht hat?
- Kannst du zusätzliche Figuren (in Wandfarbe) als Hindernisse einbauen, die sich selbstständig hin und her bewegen?

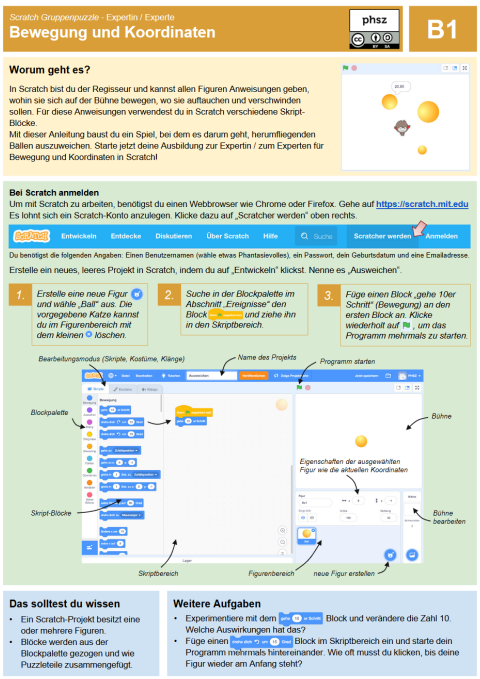
in den Unterlagen wird die Scratch Version 3.0 verwendet!

Link: **http://ilearnit.ch/download/ScratchProjektideen.pdf**

Links und Materialien

Broschüre "Gruppenpuzzle"

Die zweite Broschüre befasst sich mit den vier Aspekten Gestaltung, Bewegung, Sensoren und Rechnen & Variablen. Es wird sich in vier Gruppen Wissen über jeweils einen Aspekt angeeignet und danach werden vierer Gruppen gebildet. In jeder Gruppe gibt es einen Experten pro Aspekt und gemeinsam wird ein Spiel entwickelt. Es handelt sich dabei nicht um eine detailgetreue Schritt-für-Schritt-Anleitung, sondern die Schüler/innen sollen in zehn Schritten zum Nachdenken und zur Lösungsfindung angeregt werden. Das Gruppenprojekt ist für mindestens 2 x 90 Minuten konzipiert.



in den Unterlagen wird die Scratch Version 3.0 verwendet!
Link:
http://ilearnit.ch/download/ScratchGruppenpuzzle.pdf

code.org

Die Website code.org bietet Schüler(n)/innen die Möglichkeit, erste eigene Programme zu erstellen, ohne dafür Vorkentnisse haben zu müssen. Die Oberfläche ist, wie Scratch auch, mit Blockbausteinen aufgebaut und mit Drag and Drop einfach zu bedienen. Als Lehrperson kann man ein Profil erstellen und die Schüler/innen verwalten, ihnen Aufgaben und Kurse zuweisen und den Lernfortschritt überprüfen.

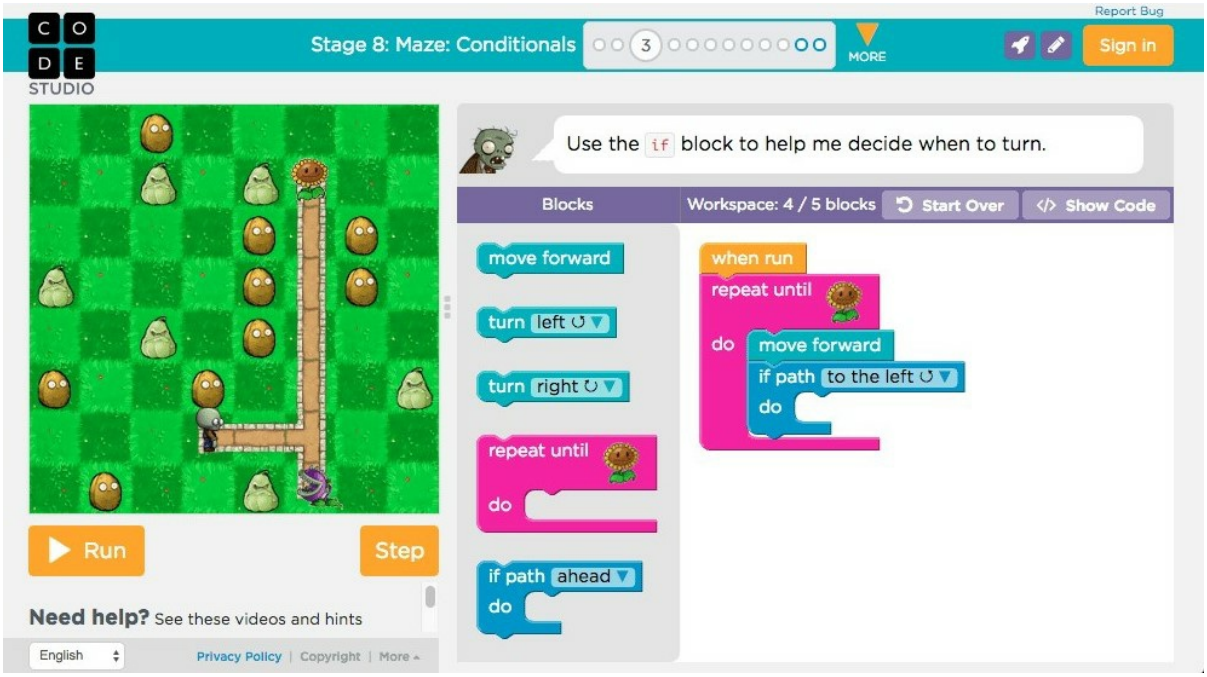
Die Kurse sind vielfältig: Von zeichnerischen Aufgaben über Spiele und Geschichten programmieren bis hin zu Rätsel lösen ist für jede Alters- und Leistungsstufe etwas dabei. Weiters gibt es auch Unplugged Aktivitäten und Materialien.

Einige Kurse sind nur auf Englisch verfügbar!

Offizielle Website: **https://code.org/**

Logo der Website Code.org:
https://code.org/images/logo.png

code.org



Oberfläche eines Tutorials (https://code.org/images/proficiency/puzzle-screenshot.png)

Links und Materialien

Hour of Code

Hour of Code ist eine Website, welche spielerisch einstündige Einführungen in die Welt der Programmierung anbietet. Es gibt eine Vielzahl an verschiedenen Tutorials mit den unterschiedlichsten Themen, die durchgespielt werden können. Es werden auch kontinuierlich neue Tutorials zu aktuellen Kinofilmen oder Spielen erstellt, was die Motivation der Schüler/innnen erhöht.

Ein Tutorial beginnt immer mit dem Kennenlernen der einzelnen Funktionen, welche Schritt für Schritt in vielen Sequenzen erlernt werden. Man kann erst zur nächsten Sequenz weitergehen, wenn man die aktuelle Aufgabe einer Sequenz geschafft hat. Am Ende hat man einen Pool an verschiedenen Funktionen zur Verfügung, aus dem man schöpfen kann, um sein eigenes Spiel oder Rätsel zu erschaffen.
Durch den einfachen, aufbauenden Zugang zur Entwicklungsumgebung bietet eine Hour of Code den idealen Einstieg in die Programmierung mit Scratch.

Zu einigen Tutorials gibt es auch ergänzendes Unterrichtsmaterial und Unterrichtsplanungen auf Englisch!

Offizielle Website: **https://hourofcode.com/de**

Logo der Seite Hour of Code
(**https://hourofcode.com/us/de/images/hour-of-code-logo.png**)

Blockly Games

Google Blockly ist eine grafische Programmiersprache, ähnlich wie Scratch, bei der Programmbausteine aneinander gereiht werden.
Basierend auf dieser Programmiersprache bietet Blockly Games acht verschiedene Spiele an, bei denen man Schritt für Schritt eine Aufgabe löst und dabei immer mehr Bausteine kennen lernt. Hinter den Bausteinen steckt JavaScript, diesen Code kann man sich nach jedem erfolgreich absolvierten Level auch ansehen.
Auf der Website von Google Blockly gibt es Anregungen für weitere Projektideen und man kann zudem weitere, eigene Programme erstellen. Diesen Code kann man sich in verschiedenen Programmiersprachen (u.a. PHP, Phyton) ansehen.

Website nur auf Englisch verfügbar!

Blockly Games: **https://developers.google.com/blockly/**
Google Blockly: **https://blockly-games.appspot.com/?lang=de**

Logo der Seite Blockly Games:
https://blockly-games.appspot.com/index/title-beta.png

Apps

Apps

Heutzutage besitzt bereits ein Großteil der Schüler/innen ein eigenes Smartphone und/oder Tablet und diese Entwicklung kann positiv im Informatikunterricht genutzt werden. Mit der App "Pocket Code", die auf den nächsten Seiten vorgestellt wird, kann jede/r ganz einfach zum Programmierer werden und eigene Anwendungen erstellen.
Da das Smartphone nicht mehr aus dem Alltag wegzudenken ist, kann durch die App das informatische Denken überall weiterentwickelt werden, egal ob beim Warten auf den Schulbus, in der Mittagspause oder daheim am Sofa.

Pocket Code

Pocket Code ist eine App, mit der man am Smartphone kleine Anwendungen, Spiele, Animationen und vieles mehr programmieren kann.

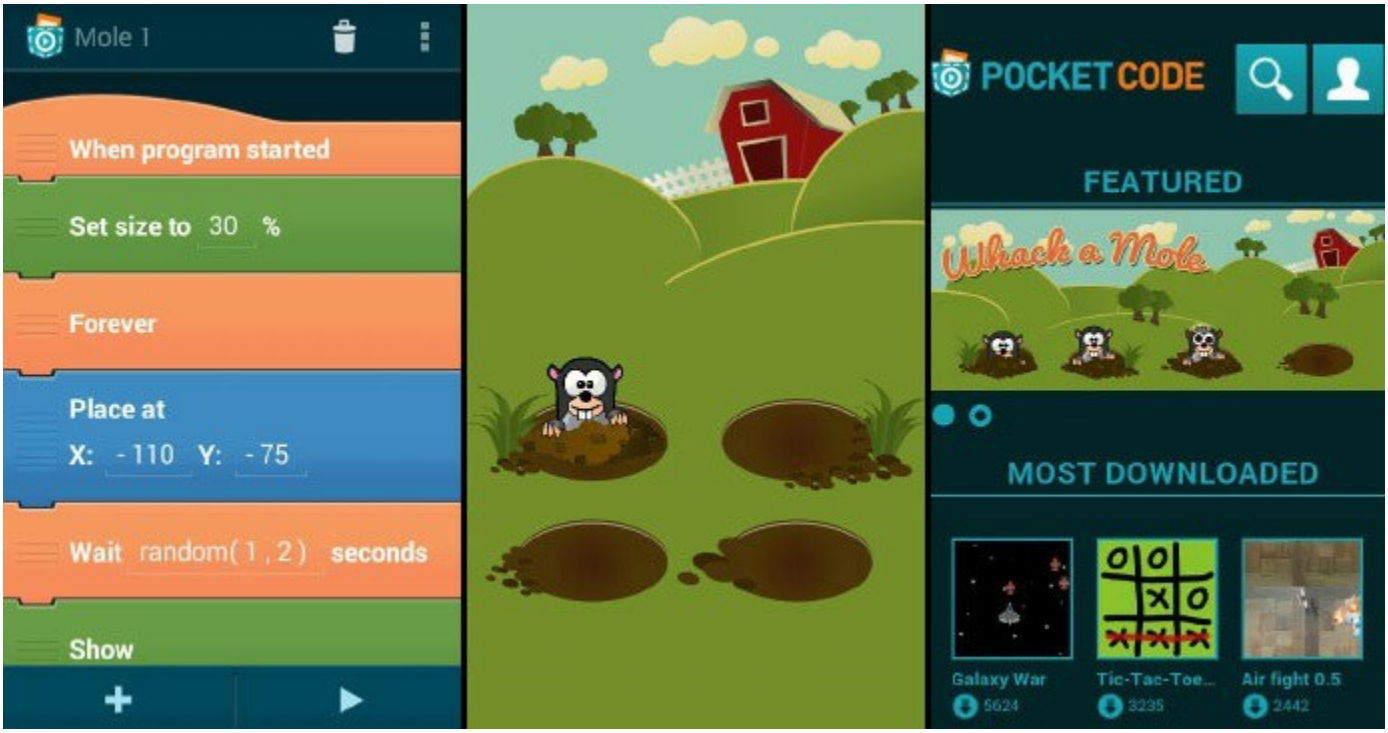
Logo der App Pocket Code

Mit grafischen Programmierbausteinen können einfach und ohne viel Vorwissen Anwendungen und Spiele erstellt werden, wodurch sich schnell Erfolgserlebnisse einstellen. Alles, was dazu benötigt wird, ist ein Smartphone bzw. Tablet mit Android oder iOS (Betriebssystem). Für die Programme können die Sensoren des Smartphones/Tablets wie GPS, Neigungssensor, Gesichtserkennung, etc. genutzt werden. Die fertigen Programme können hochgeladen und mit anderen Usern geteilt werden. Um selbst Grafiken zu gestalten, kann man zusätzlich noch die App *Pocket Paint* herunterladen und die dort erstellten Bilder in Pocket Code verwenden.

Offizielle Website: **https://pocketcode.org**

Download *Pocket Code*: **https://www.catrobat.org/#pocketcode**
Download *Pocket Paint* auf Android: **https://play.google.com/store/apps/details?id=org.catrobat.paintroid**

Pocket Code



Oberfläche der App Pocket Code (https://wwwhatsnew.com/wp-content/uploads/2016/03/pocket-code.jpg)

Pocket Code

Hinweis zur Verwendung auf dem Betriebssystem iOS

Die iOS-Version bietet derzeit noch nicht alle Funktionen, welche auch auf Android zur Verfügung stehen (Stand April 2019). So fehlen noch Szenen, der Querformat-Modus, Berechnen im Formeleditor und einige Bausteine (Sagen, Denken, Bedingte Anweisungen, Fragen, Klonen, nach vorne gehen, Malstift und Stempel, Gehe zu, Physikalisches).
Pocket Paint ist dafür schon integriert und muss nicht extra installiert werden.

Durch diese Einschränkungen ist Pocket Code auf iOS noch nicht für alle Aufgaben des folgenden Lehr- und Lernpaketes geeignet.

Links und Materialien

Lehrmaterial für einen fächerintegrativen Einsatz
Auf dieser Seite gibt es durchgeplante Unterrichtsstunden für den Einsatz von Pocket Code im Physikunterricht in der 8. Schulstufe zum Thema "1. und 2. Newtonsches Gesetz" (4-6 Unterrichtsstunden) bzw. Vorlagen, wie man ein Quiz, Puzzle oder interaktives Buch mit Hilfe von Pocket Code (5-6 Unterrichtsstunden) in der 7.-9. Schulstufe gestaltet.

Link: **https://learninglab.tugraz.at/informatischegrundbildung/pocket-code/**
Link PDF: **https://learninglab.tugraz.at/informatischegrundbildung/wp-content/uploads/2017/09/PocketCodeDigitaleGrundbildung.pdf**

MOOC/Moodle Kurs
Der 5-wöchige Online Kurs bietet Videos, Anleitungen und Übungen für die ersten Schritte mit der App. Die Inhalte des MOOCs sind auch als Moodle-Kurs verfügbar, welcher sich in ein anderes (schuleigenes) Moodle-System laden lässt.

Link zum
MOOC: **https://imoox.at/mooc/local/courseintro/views/startpage.php?id=29**
Link zum Moodle-Kurs: **https://tc.tugraz.at/main/course/view.php?id=1415**

Links und Materialien

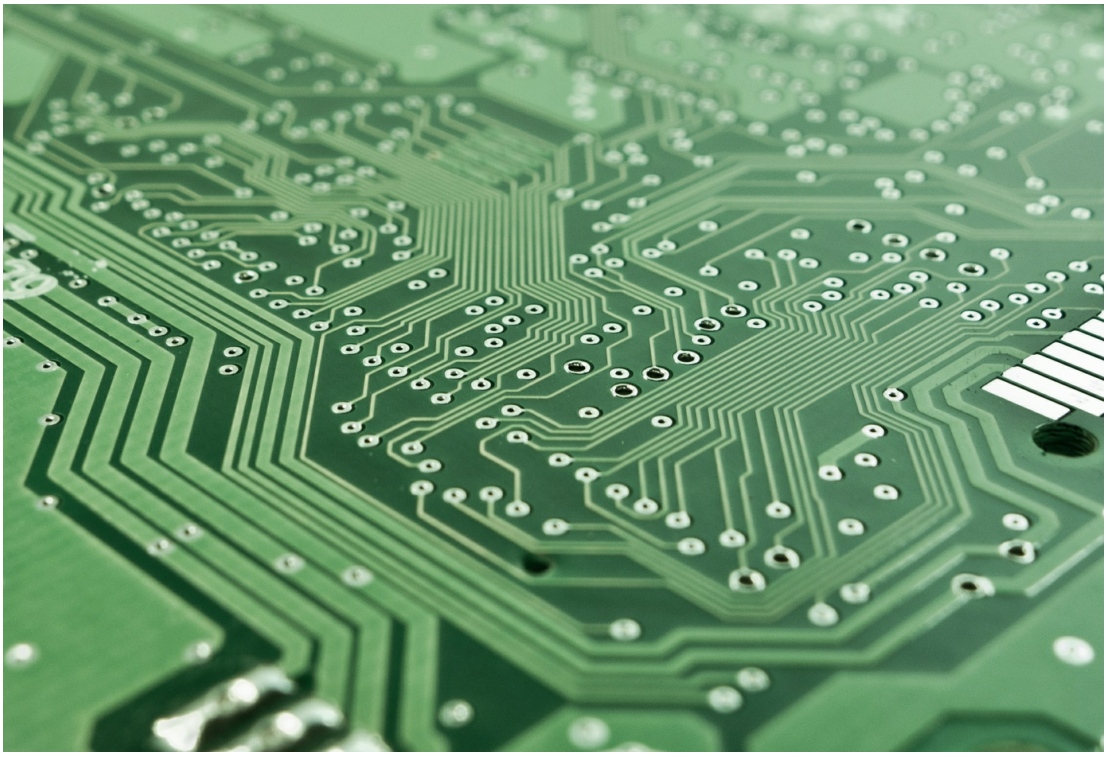
MOOC

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/6wvTiAoCbtE

Platinen

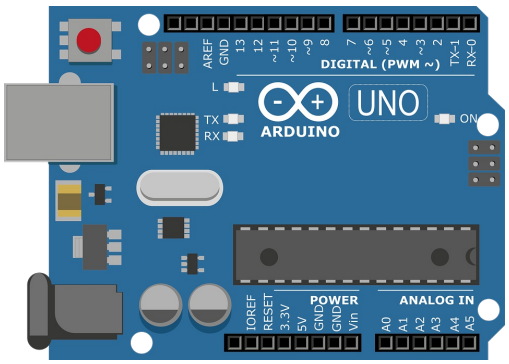


Platinen

Das Programmieren von Platinen ermöglicht die Umsetzung eines breiten Spektrums an eigenen Ideen und Projekten. Eine Software zu entwickeln, welche zur Steuerung von Hardware verwendet wird, übt dabei besonderen Reiz auf die Schüler/innen aus. Kleine Erfolgserlebnisse zu Beginn steigern schnell die Motivation, größere und komplexere Aufgaben zu lösen.

Es werden drei verschiedene Platinen vorgestellt, welche sich sehr gut für den Einsatz im Unterricht eignen, da sie zum Teil auch speziell für die Verwendung in der Schule entwickelt wurden. Der Code kann einfach und mit wenig Aufwand von einem Computer, Tablet oder Smartphone auf die Platine übertragen werden. Trotz der Einfachheit der vorgestellten Platinen bieten sie viele Umsetzungsmöglichkeiten verschiedenster Aufgaben und lassen dabei der Kreativität freien Lauf.

Arduino



Arduino ist ein Microcontroller, der über mehrere analoge und digitale Ein- und Ausgänge verfügt. Über einen USB-Anschluss wird er mit dem Computer verbunden, sodass Programmcode ganz einfach überspielt werden kann. Auf der offiziellen Website gibt es kostenlos die Arduino IDE (Integrierte Entwicklungsumgebung) zum Herunterladen, welche auf den Programmiersprachen C bzw. C++ basiert.

Durch mehrere Stiftleisten lassen sich LEDs, Sensoren, Motoren, Bildschirme, etc. anschließen und durch den Code steuern, wodurch der Arduino vielseitig einsetzbar ist.

Neben dem Klassiker Arduino Uno gibt es noch einige andere Ausführungen wie den Arduino Nano, Arduino Mega oder Arduino Micro, welche sich in Größe und Ausstattung unterscheiden, aber alle demselben Prinzip folgen.

Offizielle Website: **https://www.arduino.cc/**

Steckbrief

Arduino Uno	
Empfohlene Schulstufe	7.-12. Schulstufe
Kategorie	Platine (53.4 mm × 68.6 mm)
Umsetzungsmöglichkeiten	Der Arduino ermöglicht die Steuerung verschiedenster Geräte. Mit ihm kann man einfache LEDs zum Leuchten bringen, aber auch die Steuerung von elektronischen Bauteilen und selbstgebauten Robotern verwirklichen.
Programmiersprache/ -umgebung	Arduino IDE (C, C++) zum Download und als Webeditor Link: **https://www.arduino.cc/en/Main/Software**
Notwendige Vorkenntnisse	Keine
Ausstattung	14 Input/Output Pins (6 PWM-Ausgänge), 6 analoge Inputs, 1 USB Anschluss, 16 Mhz Quarz, ICSP Header, Reset Knopf
Zusätzliches Zubehör	USB-Kabel, LEDs, Sensoren, Bildschirm, Motoren,…
Besonderheiten	Es gibt kaum Grenzen bei den Umsetzungsmöglichkeiten.
Preis	Ca. 20 € (Stand: Februar 2019) Nachbau von der Firma Elegoo: ab 10 €
Offizielle Website	**https://www.arduino.cc**

Links und Materialien

Arduino-Workshops
Auf dieser Seite finden sich ein Kurzworkshop mit drei Projektideen und ein etwas ausführlicherer Workshop mit sechs Ideen und zugehöriger Präsentation, Leitfaden und einem Ablaufplan. Es werden unter anderem die Themen Befehle, if-Abfragen und Variablen behandelt.
Für die Workshops werden zusätzliche Bauteile (Breadboard, USB-Kabel, Steckkabel, Widerstände, LEDs, Taster, etc.) benötigt, welche es als Startersets günstig zu erwerben gibt.

Link: **https://starthardware.org/arduino-unterrichtsmaterial/**

Arduino im Physikunterricht
Unter dem nachfolgenden Link findet man eine fächerübergreifende Unterrichtssequenz zum Thema Beschleunigung für die 10. Schulstufe. Die Unterrichtssequenz ist für eine Doppeleinheit konzipiert und ist als Stationenbetrieb mit 8 Stationen aufgebaut.

Link: **https://learninglab.tugraz.at/informatischegrundbildung/der-arduino-im-physikunterricht/**

BBC micro:bit

Der BBC micro:bit ist eine speziell für die Schulbildung entwickelte Platine, ausgestattet unter anderem mit einer LED-Matrix, zwei programmierbaren Knöpfen, verschiedenen Sensoren und Bluetooth, um mit anderen Geräten kommunizieren zu können.
Der BBC micro:bit kann mit den Sprachen Blocks, JavaScript und Python im eigenen MakeCode Editor programmiert werden.

Offizielle Website: **https://microbit.org/de/**

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden. Link zum Inhalt: https://www.youtube.com/embed/Wuza5WV	Inhalt kann in diesem Ausgabeformat nicht angezeigt werden. Link zum Inhalt: https://www.youtube.com/embed/zkleA4_C

Steckbrief

BBC micro:bit	
Schulstufe	5.-7. Schulstufe
Kategorie	Platine (43mm x 52mm)
Umsetzungsmöglichkeiten	Anzeigen von Text durch LED-Matrix, Verschicken von Nachrichten an andere micro:bits per Bluetooth,...
Programmiersprache/ -umgebung	Blocks, JavaScript (MakeCode Editor) Python (MicroPython Editor) Link: **https://microbit.org/de/code/** Swift Playgrounds (App) Link: **https://microbit.org/de/guide/swift-playgrounds/#get-swift**
Notwendige Vorkenntnisse	Keine
Ausstattung	5x5-LED-Matrix, 2 programmierbare Buttons, Reset Button, Micro-USB-Port, 32-Bit-Prozessor, 16-kB-RAM, Magnetometer, Beschleunigungssensor, 20 Input/Output-Pins (6 analoge), Bluetooth
Zusätzliches Zubehör	Krokodilklemmen, Batteriebox, Micro-USB-Kabel
Besonderheiten	Verbindung mit Smartphone möglich
Preis	20€ (Stand: März 2019)
Offizielle Website	**https://microbit.at** **https://microbit.org/de/**

Links und Materialien

Beispielsammlung

Auf der offiziellen Website findet sich eine große Sammlung an Projektideen mit Anleitungen und Materialien für Lehrer/innen.

Ressourcen für Lehrer/innen:
https://microbit.org/de/teach/
Projekte (Tutorials, Spiele, Musik, ...):
https://makecode.microbit.org/projects

Dokumentation

Weiters gibt es auf der BBC-micro:bit-Website auch Dokumentationen für die Programmiersprachen JavaScript und Python.

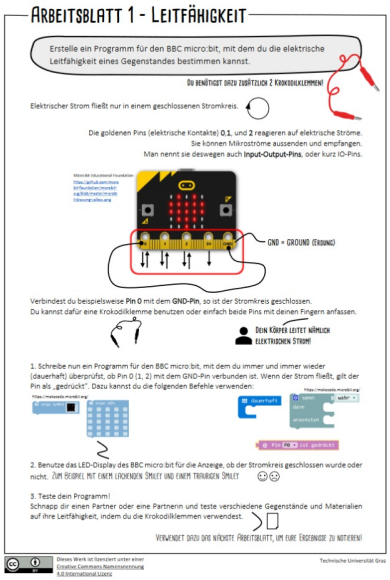
Dokumentationen nur auf Englisch verfügbar!

Java Script: **https://makecode.microbit.org/javascript**

Links und Materialien

Werkstattbeispiele

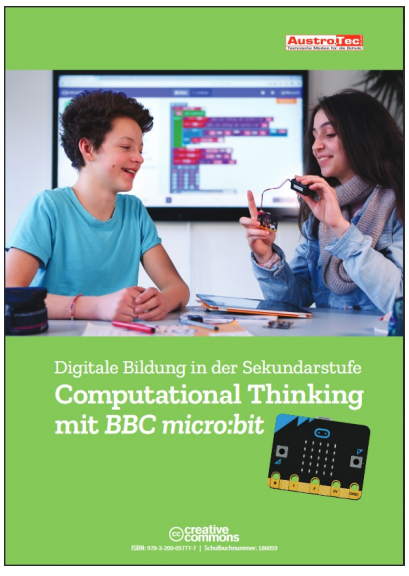
Auf dieser Seite finden sich drei Beispiele, wie der BBC micro:bit fächerübergreifend (Physik, Musik, Deutsch) angewandt werden kann. Die durch-geplanten Beispiele eignen sich für die 5.-6. (Musik, Deutsch) bzw. für die 7.-8. Schulstufe (Physik) und sind für 1-3 Unterrichtseinheiten konzipiert. Behandelt werden die Themen Elektrizität/Stromkreis/Leitfähigkeit, Tonleiter/ aktives Hören/Rhythmus und Sprach- und Schreib-förderung.



Links und Materialien

Schulbuch "Digitale Bildung in der Sekundarstufe – Computational Thinking mit BBC micro:bit"

Dieses Schulbuch beinhaltet 22 fächerübergreifende Projektideen, bei denen der BBC micro:bit zum Einsatz kommt. Die Beispiele können in einer bis drei Schulstunden verwirklicht werden und sind in drei Schwierigkeitsstufen eingeteilt. Die Aufgaben, welche oft alltägliche Probleme darstellen, können auf verschiedene Wege gelöst werden. Tipps und Hilfestellungen zu den Lösungen finden sich auf der dazugehörigen WIKI-Seite.
Neben den Beispielen beinhaltet das Schulbuch auch noch ein Einführungskapitel "Einach programmieren!", um einen einfachen und gelungenen Einstieg in das Arbeiten mit einem BBC micro:bit zu garantieren.



Link Schulbuch (PDF):
https://microbit.eeducation.at/images/f/f2/Buch-microbit_20180729.pdf
Link WIKI-Seite: **https://microbit.eeducation.at/index.php/Hauptseite**

Links und Materialien

MOOC
Der 5-wöchige Online-Kurs bietet Videos, Übungen und weitere Materialien rund um den BBC micro:bit, basierend auf dem Schulbuch ***Computational Thinking mit BBC micro:bit - Digitale Bildung in der Sekundarstufe.***
Link: **https://imoox.at/mooc/local/courseintro/views/startpage.php?id=54**

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/FyBDk36pavE

Calliope mini

Der Calliope mini ist eine gut ausgestattete, sechseckige Platine, speziell für den Einsatz in der Schule entwickelt. Sie besitzt neben zahlreichen Pins und Anschlüssen eine Matrix aus 25 LEDs, mit der Texte und Symbole ausgegeben werden können. Weiters können Töne und Melodien mithilfe des integrierten Piezo-Lautsprechers wiedergegeben werden.
Programmiert werden kann die Platine unter anderem über den Calliope Mini Editor, welcher mit einer grafischen Programmiersprache und JavaScript ausgestattet ist.

Offizielle Website: **https://calliope.cc/**

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/k2Hf2Ua

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/FDznaxY

Steckbrief

Calliope mini	
Schulstufe	5.-9. Schulstufe
Kategorie	Platine (sechseckig 85mm in der Diagonale)
Umsetzungsmöglichkeiten	Text auf der LED Matrix ausgeben, Lied spielen lassen, Roboter programmieren,...
Programmiersprache/ -umgebung	Programmier-Blöcke (ähnlich wie Scratch) oder JavaScript Calliope mini Editor, MAKECODE, Open Roberta Lab Link: **https://www.calliope.cc/los-geht-s/editor** Swift Playgrounds (App) Link: **https://calliope.cc/playground**
Notwendige Vorkenntnisse	Keine
Ausstattung	Bluetooth, 5x5 LED Matrix, Beschleunigungssensor, Gyroskop, Magnetometer, MEMS Mikrofon, DC Motortreiber, Piezo-Lautsprecher, Programmierbare RGB LED, 2 programmierbare Taster, Serielle Schnittstelle, PWM Ausgabe, 4 Bananenstecker/Krokodilklemmen-Anschlüsse, 4 analoge Eingänge, 8-11 Ein-/Ausgangsanschlüsse, USB Micro B Anschluss, JST Batterieanschluss, 2 Grove Steckverbinder
Zusätzliches Zubehör	Krokodilklemmen, Sensoren, Motoren
Besonderheiten	Programmierung auch am Tablet oder Smartphone möglich (Übertragung über Bluetooth), Kommunikation mit anderen Calliope minis
Preis	40€ (Starterset mit USB-Kabel und Batteriehalter) (Stand: Februar 2019)
Offizielle Website	**https://calliope.cc**

Links und Materialien

Offizielle Website

Auf der offiziellen Website findet sich eine große Sammlung an Anleitungen zu Projekten und Experimenten, kategorisiert nach Schulstufe, Zubehör und Schwierigkeitslevel.

Offizielle Website: **https://calliope.cc/**

Überblick und Beispiele

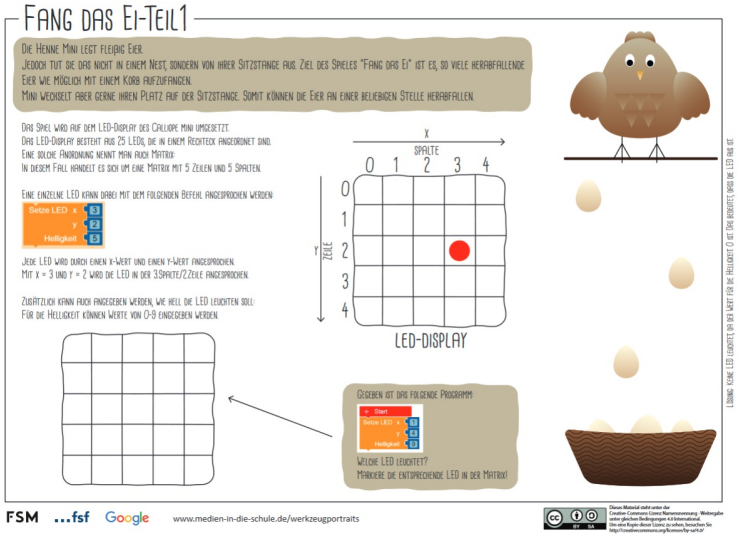
Die untenstehende Seite bietet einen guten Überblick über die Ausstattung, Fähigkeiten und elektronischen Erweiterungsmöglichkeiten, die der Calliope mini hat. Auf der Seite finden sich auch Tutorials, wie man sein erstes Programm erstellt bzw. auf die Platine überträgt.
Weiters finden sich auch zwei Projektideen mit unterschiedlicher Schwierigkeitsstufe und Zeitaufwand: Für Einsteiger gibt es eine Anleitung, wie man in 20 Minuten sein eigenes Klavier macht, und für Fortgeschrittene eine Anleitung, wie man in 3 Stunden seinen eigenen Malroboter bastelt und programmiert.

Link: **https://www.tuduu.org/lernen/calliope**
Anleitung Klavier: **https://tuduu.org/projekt/calliope-klavier**
Anleitung Malroboter: **https://tuduu.org/projekt/automatischer-malroboter**

Links und Materialien

Werkstattbeispiele

Unter dem nachstehenden Link finden sich fünf fächerübergreifende Projektideen, welche sich im Zeitrahmen von zwei bis fünf Unterrichtseinheiten mit Hilfe des Calliope mini umsetzen lassen. Die Unterrichtseinheiten sind alle durchgeplant und enthalten auch einige Tipps, Abwandlungen und Ergänzungen für ein gelungenes Projekt.



Link: **https://learninglab.tugraz.at/informatischegrundbildung/oer-schulbuch/calliope-mini/**

Links und Materialien

App Camps

Die Website App Camps stellt Unterrichtsmaterialien für die ersten Schritte mit dem Calliope mini bereit. Die Unterlagen sind eingeteilt in Kategorien für die Schulstufen 3-6 und 6-9. Die Kurse werden als Sitzungen mit Videos und Materialien aufbereitet und beinhalten auch viele Tipps und Empfehlungen zur Vorbereitung.
Jede Sitzung hat vier bis acht Unterrichtseinheiten und es werden mit Hilfe von Lernkarten eine oder mehrere Anwendungen programmiert. In einem Abschlussvideo werden wichtige Konzepte der Informatik von einem/r Softwareentwickler/in erklärt.
Link: **https://appcamps.de/unterrichtsmaterial/calliope-mini/**

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/LUxLUtCvZ

Links und Materialien

MOOC

In dem 7-wöchigen Online Kurs lernt man den Calliope mini mit Hilfe von Videos, Übungen und weiteren Materialien Schritt für Schritt kennen. Am Ende des Kurses sind die Teilnehmer/innen in der Lage, selbstständig Programme für den Calliope mini zu schreiben und eigene Ideen und Projekte zu verwirklichen.

Link: **https://imoox.at/mooc/local/courseintro/views/startpage.php?id=66**

Calliope mini erfolgreich in der Schule einsetzen

Ein weiterer 4-wöchiger MOOC, der sich speziell an Lehrer/innen richtet und zeigt, wie man den Calliope mini in der Schule einsetzt und im Unterricht einführt.

Link: **https://open.sap.com/courses/calli1**

Roboter

Roboter

Bei der Arbeit mit Robotern können die Schüler/innen den erstellten Programmcode direkt am Roboter ausprobieren. So wird der Zusammenhang zwischen Software und Hardware nachvollziehbar und greifbar für die Schüler/innen. Dies wirkt sich besonders positiv auf Motivation und Ehrgeiz aus. Doch die Arbeit mit Robotern fördert nicht nur das informatische Denken, sondern eignet sich auch besonders gut für Gruppenarbeiten, wodurch Kommunikation und soziale Kompetenzen gefördert werden.

Auf den folgenden Seiten werden der Ozobot, LEGO Mindstorms und der Thymio vorgestellt. Der Ozobot, ein kleiner Roboter der Linien nachfährt und programmierbare Farbcodes erkennt, eignet sich für erste Begegnungen mit Robotern und um Programmiererfahrungen zu sammeln. Mit LEGO Mindstorms hat man die Möglichkeit selbst einen Roboter zu bauen und zu programmieren und bietet einen guten Einstieg in die Welt der Robotik. Thymio ist ein kleiner fahrbarer Roboter, ausgestattet mit den verschiedensten Sensoren, wodurch er viele Einsatzmöglichkeiten bietet.

Ozobot

Der Ozobot ist ein kleiner, akkubetriebener Roboter, dessen Herzstück 5 Farb-Sensoren an der Unterseite bilden. Mithilfe der Sensoren fährt er schwarzen Linien auf Papier oder am Bildschirm entlang. Durch verschiedene Farbcodes können Funktionen wie Tempo ändern, Richtung wechseln, drehen etc. programmiert werden. Durch die unkomplizierte Bedienung, für die kein Computer erforderlich ist, und die schnell eintretenden Erfolgserlebnisse eignet sich der Ozobot besonders gut für den Einstieg in die Programmierung in der 5./6. Schulstufe.
Für etwas Fortgeschrittene lässt sich der Ozobot nicht nur durch farbige Linien steuern, sondern auch mit Programmcode. Mit dem Programmiertool Ozoblockly lässt sich schnell und einfach durch Codeblöcke ein Code generieren, welcher auf den Ozobot über die Farbsensoren übertragen werden kann.

Offizielle Website: **https://ozobot.com/**
Offizielle deutsche Website: **https://ozobot-deutschland.de/**
Ozoblockly: **https://ozoblockly.com/**

Steckbrief

Ozobot Evo	
Schulstufe	5.-8. Schulstufe
Kategorie	Roboter (ca. 3 cm Durchmesser)
Umsetzungsmöglichkeiten	Verfolgen einer Farblinie, Erkennen und Umsetzen von Farbcodes (Tempo ändern, Richtung wechseln, drehen,...), Programmieren von Bewegungsabläufen
Programmiersprache/ -umgebung	Farblinien zeichnen mit Filzstift oder am Bildschirm Ozoblockly Link: **https://ozoblockly.com/**
Notwendige Vorkenntnisse	Keine
Ausstattung	7 LEDs, 4 Distanzsensoren, Bluetooth, Lautsprecher
Zusätzliches Zubehör	breite Filzstifte, Papier, eventuell Smartphone/Tablet
Besonderheiten	Auf verschiedenen Niveaus anwendbar. Auch zur Verwendung als Messwerkzeug durch die Präzision der Motoren möglich.
Preis	75€ (Stand: März 2019)
Offizielle Website	**https://ozobot.com/** **https://ozobot-deutschland.de/**

Links und Materialien

Unterrichtsbeispiele

Unter dem nachstehenden Link finden Sie offen lizenzierten Lehr- und Lernunterlagen zum ***Einsatz des Ozobot im Unterricht***, die im Zuge des Projekts *You can code – YC²* entwickelt wurden. Dabei wurden 7 Unterrichtseinheiten, inklusive Erklärungsvideos, einer Farbcode-Übersicht und einem Info-Blatt mit hilfreichen Tipps, entwickelt.

Links und Materialien

Unterrichtsmaterial Offizielle Website

Auf der offiziellen deutschen Website finden sich weitere durchgeplante Lektionen und Aktivitäten für den Einsatz im Unterricht.

Link: **https://ozobot-deutschland.de/unterrichtsmaterial/**

Broschüre "Ozobot Projektideen"

In dieser Broschüre finden sich 10 Projektideen, die sich mit einem Ozobot in 10-50 Minuten umsetzen lassen. Die Übungen decken vom einfachen Linien-Abfahren bis hin zu Programmieren mit visueller Programmiersprache alle Schwierigkeitsstufen ab.

Link: **http://ilearnit.ch/download/OzobotProjektideen.pdf**

LEGO Mindstorms

Herzstück von LEGO Mindstorms ist ein programmierbarer Minicomputer, welcher in einen LEGO-Baustein eingebettet ist, an dem Motoren und Sensoren angeschlossen werden können. Somit kann ein selbstgebauter Roboter gesteuert werden. Die Schüler/innen müssen sich zum einen Gedanken über die Konstruktion der Hardware machen und zum anderen die Software entsprechend programmieren, um bestimmte Aufgabenstellungen lösen zu können. Die mitgelieferte Software beinhaltet Aufgaben für verschiedene Altersgruppen und bietet Anleitungen und Tipps zum Modellbau und zur Programmierung. Programmiert wird der Roboter mit der eigenen, symbolbasierten LEGO MINDSTORMS EV3-Software am PC oder Tablet.

Offizielle Website: **https://www.lego.com/de-de/mindstorms**

Steckbrief

LEGO Mindstorms - Education EV3 Set	
Schulstufe	5.-12. Schulstufe
Kategorie	Roboter
Umsetzungsmöglichkeiten	Roboter bauen und programmieren, die u.a. vielfältigen Einsatz in MINT-Fächern finden (Experimente durchführen, Messdaten erfassen, Mechanikphänomene beobachten,...)
Programmiersprache/ -umgebung	Symbolbasierte LEGO MINDSTORMS EV3-Software Link: **https://www.lego.com/de-de/mindstorms/downloads/download-software**
Notwendige Vorkenntnisse	Keine
Ausstattung	Programmierbarer Stein mit verschiedenen Anschlüssen (PC, USB, SD-Karte), 4 Sensoren (Licht/Farbe, Gyro, 2 x Berührung, Ultraschall), Infrarot-Fernsteuerung, 2 große Motoren, 1 mittlerer Motor, Lautsprecher, Akku, Verbindungskabel, zahlreiche LEGO-Technic-Bausteine
Zusätzliches Zubehör	Weitere LEGO-Bausteine, weitere Sensoren
Besonderheiten	Sowohl Software als auch Hardware werden selbst entworfen
Preis	280 € (Stand: März 2019)
Offizielle Website	**https://www.lego.com/de-de/mindstorms**

Links und Materialien

Unterrichtsmaterial Offizielle Website

Auf der offiziellen Website werden Unterrichtsmaterialien zum Download angeboten. Die Materialien reichen von einzelnen Experimenten bis hin zu durchgeplanten, aufeinander aufbauenden Unterrichtseinheiten für bis zu 30 Schulstunden.

Link: **https://education.lego.com/de-de/downloads/mindstorms-ev3/curriculum**

Anregungen für den Einsatz im Unterricht

In dieser PDF-Datei finden sich sechs aufeinander aufbauende Unterrichtseinheiten, in denen die Schüler/innen Schritt für Schritt den LEGO-Mindstorms-Roboter und seine Funktionen kennenlernen. Jede Einheit hat einen anderen Schwerpunkt, dabei werden zuerst die Motoren und anschließend die verschiedenen Sensoren behandelt.

Link: **https://www.lmz-bw.de/fileadmin/user_upload/Downloads/Handouts/laporte-hoffmann-lego-mindstorms.pdf**

Thymio

Thymio ist ein kleiner, mit vielen Sensoren ausgestatteter Roboter, der speziell für pädagogische Zwecke entwickelt wurde, um die Welt der Robotik zu entdecken. Durch die Möglichkeit, zwischen visueller, blockbasierter und textbasierter Programmierung auszuwählen, ist der Thymio für jedes Alter geeignet.

Offizielle Website: **https://www.thymio.org/de:thymio**

Inhalt kann in diesem Ausgabeformat nicht angezeigt werden.

Link zum Inhalt:

https://www.youtube.com/embed/RTiZDiglC

Inhalt kann in diesem Ausgabeformat nicht

Steckbrief

Thymio II	
Schulstufe	1.-12. Schulstufe
Kategorie	Roboter (110 mm x 112 mm x 53 mm)
Umsetzungsmöglichkeiten	Handbewegungen folgen, Hindernissen ausweichen, Linien verfolgen, zeichnen,...
Programmiersprache/ -umgebung	Visuelle Programmierumgebung, Google Blockly, textuelle Programmierung Programmierumgebung: Aseba Link: **https://www.thymio.org/de:start**
Notwendige Vorkenntnisse	Keine
Ausstattung	2 Räder, 39 LED, 11 Sensoren (Abstand, Temperatur, 3-achsige Beschleunigung, 5 x Abstand, 2 x Boden, Infrarot) 5 berührungsintensive Knöpfe, Li-Po Batterie-Laderanzeige, Mikrofon, Lautsprecher, USB-Anschluss, Speicherkarten-Slot, Haken für einen Anhänger, Reset-Knopf
Zusätzliches Zubehör	Wireless DIY Kit, Infrarot-Fernsteuerung
Besonderheiten	mit LEGO kompatibel, praktische Stift-Halterung
Preis	140€ (Stand: März 2019)
Offizielle Website	**https://www.thymio.org/de:thymio**

Links und Materialien

Unterrichtsmaterial Offizielle Website

Auf der offiziellen Website finden sich Unterrichtsmaterialien mit didaktischen Hinweisen, welche in drei Zyklen aufgebaut sind. Der erste Zyklus ist unplugged (nur Thymio ohne Computer) für Kinder ab 4 Jahren, bei der die Unterrichtseinheit aus zwölf Aktivitäten besteht. Der zweite Zyklus ist für Kinder ab 8 Jahren und es wird begonnen mit der VPL (Visual Programming Language) zu arbeiten. Für Kinder ab 12 Jahren ist der dritte Zyklus geeignet und es wird mit Blockly programmiert. In diesem Zyklus gibt es einen Workshop bestehend aus 10 Einheiten welcher sowohl mit VPL, Blockly aber auch mit Text-Programmierung gelöst werden kann.

Link: **https://www.thymio.org/de:thymioschoolprojects**

Links und Materialien

THOOL-Projekt

Das Thool-Projekt bietet viele verschiedene Ideen mit Videos und Arbeitsmaterialien im Ausmaß von einer bis zu drei Unterrichtseinheiten an.

die Videos, die bei jeder Aktivität dabei sind, sind alle auf Französisch ohne Untertitel!
Link: **https://www.thymio.org/de:thoolproject-formation**

Thymio Workshop

Auf dieser Website findet sich eine Sammlung an Arbeitsblättern, Anleitungen, Videos, Informationen und viele weitere Unterlagen rund um den Roboter Thymio. Eingeteilt sind alle Dokumente in folgende fünf Kategorien:

- Grundlagen
- Programmieren
- Karten & Aufgaben
- Behelfe & Links
- Thymio erweitert

Link: **https://padlet.com/eis/thymioWS**