

Digitale Technik und Coden / KidsDigiLab und TüftelLab Angebote

C1 Algorithmische Grundlagen – Informatik ohne Computer
C2 Einführung in Programmieren und Robotik – Sensoren und Roboter steuern
C3 Programmieren mit ScratchJr und kreative digitale Projekte

Unterrichtsreihe C: Digitale Technik und Coden / Kids DigiLab Angebote

C1 Algorithmische Grundlagen – Informatik ohne Computer

Unterrichtsthema	Algorithmische Grundlagen – Informatik ohne Computer
Jahrgangsstufen und Dauer	Jahrgangsstufen: 3–4 Dauer: 3–4 Unterrichtsstunden
Vorkenntnisse	Alltagserfahrungen mit digitalen Geräten; erste Erfahrungen mit Anleitungen oder Abläufen.
Inhalt	Die Lernenden setzen sich mit grundlegenden Konzepten algorithmischen Denkens auseinander. Sie formulieren und erproben Schritt-für-Schritt-Anleitungen, untersuchen eindeutige und mehrdeutige Anweisungen und steuern einfache Abläufe ohne Computer (z. B. Roboter steuern, Wege planen, Handlungsabfolgen entwickeln).
Didaktischer Kommentar	Die Einheit führt in informatisches Denken ein, ohne technische Geräte zu benötigen. Durch spielerische Aktivitäten erkennen die Lernenden, dass digitale Systeme auf klar strukturierten Abläufen beruhen. Fehler werden als Lernanlässe genutzt (Debugging).
Mögliche Kompetenzentwicklungen	• Abläufe in einzelne Schritte zerlegen und beschreiben • einfache Anleitungen formulieren und umsetzen • erkennen, dass Algorithmen eindeutige Handlungsanweisungen sind • Handlungsabläufe planen und erproben (z. B. Roboter steuern, Wege planen) • Fehler in Abläufen erkennen und Verbesserungen vorschlagen • Probleme strukturiert analysieren und Lösungswege entwickeln

	• eigene Lösungsstrategien erklären und reflektieren • erkennen, dass digitale Systeme auf klaren Regeln und Abläufen beruhen
Ideen und Links zur Unterrichtsreihe	<u>Material 1:</u> Im Kids DigiLab lernen Lernende, digitale Technologien durch experimentelles und handlungsorientiertes Arbeiten zu verstehen. Angebote wie Programmieren mit Karten oder das Steuern von Abläufen ohne Computer fördern algorithmisches Denken und Problemlösekompetenz. Passende Angebote zu den Workshops: <u>Weitere Hinweise unter:</u> https://technikmuseum.berlin/angebote/kids-digilab <u>Spezielle Angebote Kids DigiLab Berlin:</u> <u>Workshop 1:</u> Der Einstieg in das Thema informatische Grundkompetenzen erfolgt, indem Programme geschrieben werden, die Kunst erschaffen. Erst analog mit informatischen Konzepten wie dem algorithmischen Denken und dann durch Diskussion der Einbindungsmöglichkeiten in den pädagogischen Alltag. Anschließend werden testen verschiedene Tools getestet, mit denen Lernende geometrische Formen, Muster und originelle Eigenkreationen erstellen können. Der Workshop richtet sich an pädagogische Fach- und Lehrkräfte für Kinder der Primarstufe. Es sind keine Vorkenntnisse erforderlich. <u>Weitere Hinweise unter:</u> https://technikmuseum.berlin/angebot/kreatives-programmieren/ <u>Workshop 2:</u> In dieser Fortbildung erfolgt ein niederschwelliger Einstieg in das Thema informatische Grundkompetenzen. Was Digitale Bildung mit unserem Alltag zu tun hat wird reflektiert, es werden Ideen gesammelt, wie das Thema klischeefrei in der pädagogischen Praxis umgesetzt und wie ganz einfach einen Bezug zur Lebenswelt

hergestellt werden kann. Im Anschluss testen die Teilnehmenden ausgewählte Robotik-Spielzeuge und machen sich mit ihren Funktionen vertraut. Der Workshop richtet sich an pädagogische Fach- und Lehrkräfte. Es sind keine Vorkenntnisse erforderlich.

Weitere Hinweise unter:

<https://technikmuseum.berlin/angebot/denken-und-spielen-wie-ein-computer/>

Material 2:

Die Broschüre zeigt, wie Lernende im Grundschulalter grundlegende Konzepte der Informatik spielerisch und handlungsorientiert entdecken können – sowohl mit als auch ohne digitale Geräte. Im Mittelpunkt stehen zentrale informatische Ideen wie Algorithmen, Daten, Codierung, digitale Kommunikation und das Funktionsprinzip von Computern. Sie verdeutlicht, dass informatische Bildung vor allem bedeutet, Probleme zu strukturieren, Abläufe zu verstehen und Lösungen zu entwickeln, und nicht primär das Arbeiten am Computer. Lernende lernen, Informatik in Alltagssituationen zu erkennen, etwa beim Sortieren, bei Schritt-für-Schritt-Anleitungen oder beim Entschlüsseln von Informationen. Die Broschüre enthält Hintergrundwissen für pädagogische Fachkräfte sowie konkrete Praxisimpulse, Experimente und Beispiele, die forschendes Lernen unterstützen und zeigen, wie informatisches Denken systematisch aufgebaut werden kann. Ziel ist es, Neugier auf digitale Technologien zu fördern und Kinder zu befähigen, ihre digital geprägte Umwelt reflektiert zu verstehen und mitzugestalten.

Weitere Hinweise unter:

https://campus.stiftung-kinder-forschen.de/pluginfile.php/1945/engage_article/content/94/broschuere_informatik_entdecken%20%281%29.pdf

https://campus.stiftung-kinder-forschen.de/pluginfile.php/1945/engage_article/content/210/praxiskarten_informatik_entdecken.pdf

	https://campus.stiftung-kinder-forschen.de/pluginfile.php/1945/engage_article/content/106/kinderkarten_informatik_entdecken.pdf
Weitere Links zum Thema	<u>Material 3:</u> Wie funktionieren Roboter? Die Lernenden schlüpfen in die Rolle des Programmierenden und in die des Roboters und steuern sich gegenseitig. <u>Weitere Hinweise unter:</u> https://www.stiftung-kinder-forschen.de/praxisanregungen/experimente-fuer-kinder/experiment/einmal-roboter-sein/ <u>Material 4:</u> Gemeinsam tanzen macht Spaß. Doch wie funktionieren Choreografien? Woher weiß jeder, wann er sich wie bewegen soll? <u>Weitere Hinweise unter:</u> https://www.stiftung-kinder-forschen.de/praxisanregungen/experimente-fuer-kinder/experiment/programmier-deinen-tanz/

Unterrichtsreihe C: Digitale Technik und Coden / Kids DigiLab Angebote
C2 Einführung in Programmieren und Robotik – Sensoren und Roboter steuern

Unterrichtsthema	Einführung in Programmieren und Robotik – Sensoren und Roboter steuern
Jahrgangsstufen und Dauer	Jahrgangsstufen: 3-4 Dauer: 6-8 Unterrichtsstunden
Vorkenntnisse	Grundverständnis algorithmischer Abläufe aus C1; erste Erfahrungen mit Schritt-für-Schritt-Anleitungen und Problemlöseprozessen.
Inhalt	Die Lernenden lernen grundlegende Konzepte des Programmierens und der digitalen Steuerung kennen. Sie erstellen einfache Programme mit visuellen Programmierertools und erkunden, wie Roboter über Sensoren Informationen aufnehmen und auf Befehle reagieren. Anhand praktischer Aufgaben – etwa dem Programmieren von Robotern oder dem Entwickeln einfacher Steuerprogramme mit dem Calliope mini – planen, testen und verbessern die Lernenden digitale Abläufe. Sie setzen sich mit der Frage auseinander, wie Computer und Roboter Anweisungen verarbeiten und wie Programme strukturiert sind.
Didaktischer Kommentar	Die Einheit verbindet informatisches Lernen mit handlungsorientierter Technikbildung. Durch das eigene Programmieren und das Arbeiten mit Robotern erleben die Lernenden unmittelbar die Wirkung von Algorithmen und digitalen Steuerungen. Fehler werden als Lernanlass genutzt, um Programme zu optimieren und ein Verständnis für iterative Entwicklungsprozesse zu entwickeln. Die Einbindung außerschulischer Lernorte wie dem Kids DigiLab ermöglicht authentische Erfahrungen mit digitaler Technik und stärkt die Motivation sowie die Anschlussfähigkeit an informatische Bildung.
Mögliche Kompetenzentwicklungen	• einfache Programme mit visuellen Programmierumgebungen erstellen • erkennen, dass digitale Systeme auf Algorithmen und Befehlen beruhen • Sensoren als Informationsquellen für technische Systeme verstehen • Programme planen, testen und verbessern

	• Ursache-Wirkungs-Zusammenhänge in digitalen Systemen erklären • technische Probleme analysieren und Lösungsstrategien entwickeln • digitale Werkzeuge zielgerichtet nutzen • im Team kooperativ an Programmieraufgaben arbeiten • eigene Ideen mit digitalen Medien umsetzen und reflektieren
Ideen und Links zur Unterrichtsreihe	<u>Material 1:</u> Im Kids DigiLab lernen Lernende, digitale Technologien durch experimentelles und handlungsorientiertes Arbeiten zu verstehen. Angebote wie Programmieren mit Karten oder das Steuern von Abläufen ohne Computer fördern algorithmisches Denken und Problemlösekompetenz. Passende Angebote zu den Workshops: <u>Weitere Hinweise unter:</u> https://technikmuseum.berlin/angebote/kids-digilab <u>Spezielle Angebote Kids DigiLab Berlin:</u> <u>Workshop 1:</u> Sensoren finden sich fast überall. Digitale Thermometer geben mithilfe von Sensoren die Temperatur an. Und manchmal lösen Sensoren sogar Befehle aus: Erfasst ein Bewegungsmelder ein Signal, geht das Licht an. Aus der Automatisierung von Prozessen sind Sensoren nicht wegzudenken. Im Workshop wird reflektiert, wie unsere Umwelt über unsere Sinne wahrgenommen wird und wie empfangene Signale und deren enthaltene Informationen bestimmte Handlungen auslösen. Im Anschluss beschäftigen erfolgt die ausgiebige Beschäftigung mit dem kleinen Roboter Ozobot, der seine Anweisungen über Sensoren erhält. <u>Weitere Hinweise unter:</u> https://technikmuseum.berlin/angebot/sinne-signale-und-sensoren/ <u>Workshop 2:</u>

In diesem Workshop wird ein Ausflug in die Welt der informatischen Grundkonzepte unternommen. Zunächst erfolgt das gegenseitige Steuern durch den Raum mittels einfacher Pfeil-Befehle, um das Thema Algorithmen spielerisch zu erfahren. Da beim Programmieren naturgemäß Fehler auftreten, bilden das Testen und Verbessern zentrale Bestandteile. Analog dazu wird ein Programm aus Pfeil-Befehlen erstellt, mit dem eine Roboter-Figur über das Spielfeld gesteuert wird. Bei Abweichungen vom erwarteten Ergebnis werden die Programme getestet und optimiert. Im Anschluss werden die erworbenen Kenntnisse übertragen, um den Roboter Dash zu programmieren.

Weitere Hinweise unter:

<https://technikmuseum.berlin/angebot/rettet-codie/>

Workshop 3:

Ein Roboter zur Reinigung des Zimmers stellt eine nützliche Erfindung dar. Modelle zum Staubsaugen oder Rasenmähen sind bereits auf dem Markt etabliert. Im Workshop wird gemeinsam mit den Lernenden erarbeitet, in welchen Bereichen des Alltags Roboter sinnvoll eingesetzt werden können, gefolgt von der Entwicklung eigener Ideen für Robotermodelle. Zudem wird reflektiert, wie ein Roboter Verhaltensanweisungen erhält. Zu diesem Zweck agieren die Lernenden selbst als Roboter und steuern sich gegenseitig mittels Programmierung durch den Raum. Abschließend erfolgt die gemeinsame Erforschung und Programmierung des Robotik-Spielzeugs Blue-Bot.

Weitere Hinweise unter:

<https://technikmuseum.berlin/angebot/roboter-programmieren-grundschule/>

Material 2:

Die Lernaufgabe führt Lernende der Jahrgangsstufen 3-4 in die Arbeit mit dem Mikrocontroller Calliope mini ein, indem sie einen „Schatzdetektor“ programmieren, der mithilfe drahtloser Funkkommunikation ein verstecktes Objekt aufspürt. Dabei übernehmen zwei Geräte die Rollen von Sender und Empfänger und tauschen Signale aus, deren Stärke Hinweise auf die Entfernung zum „Schatz“ gibt.

	https://bildungsserver.berlin-brandenburg.de/unterricht/faecher/mathematik-naturwissenschaften/mint/i-mint-akademie/i-mint-akademie-grundschule/sachunterricht/calliocademy
Weitere Links zum Thema	Material 3: Die Lernaufgabe lässt Lernende mit dem Calliope mini einen Funk-Schatzdetektor programmieren und vermittelt dabei grundlegende Konzepte der Informatik wie Sensorik, Algorithmen und Problemlösen durch praktisches Experimentieren. https://bildungsserver.berlin-brandenburg.de/fileadmin/bbb/unterricht/faecher/naturwissenschaften/mint/iMINT-Akademie/iMINT-Grundschule/Sachunterricht/Calliope/cadmey/Calliocademy_Unterrichtsmaterialien/Schatzsuche/iMINT_Informatik_Lernaufgabe_Schatzsuche_final.pdf

Unterrichtsreihe C: Digitale Technik und Coden

C3 Programmieren mit ScratchJr und kreative digitale Projekte

Unterrichtsthema	Programmieren mit ScratchJr und kreative digitale Projekte
Jahrgangsstufen und Dauer	Jahrgangsstufen: 3-4 Dauer: 4-6 Unterrichtsstunden
Vorkenntnisse	Grundverständnis von Algorithmen und Programmabläufen aus C1 sowie erste Erfahrungen mit visuellen Programmierertools oder Robotik aus C2.
Inhalt	Die Lernenden vertiefen ihre Kenntnisse im Programmieren mit ScratchJr und entwickeln eigene digitale Projekte wie Geschichten, Animationen oder einfache Spiele. Sie planen Programmabläufe, setzen diese um und optimieren ihre Programme durch Testen und Debugging. Ergänzend nutzen sie Angebote aus dem TüftelLab, um weitere digitale Werkzeuge kennenzulernen und Maker-Projekte zu erproben. Dabei setzen sie sich mit der Frage auseinander, wie digitale Anwendungen gestaltet werden und wie kreative Ideen technisch umgesetzt werden können.
Didaktischer Kommentar	Die Einheit stärkt die Gestaltungskompetenz im digitalen Bereich, indem Lernende eigene Ideen entwickeln und in Programmierprojekten umsetzen. Programmieren wird als kreativer und problemlösender Prozess erfahrbar, der Planung, Kooperation und Reflexion erfordert. Durch die Verbindung mit Maker-Angeboten wird informatische Bildung als aktiver Gestaltungsprozess verstanden, der technisches Verständnis und kreative Ausdrucksmöglichkeiten verbindet. Fehler werden bewusst als Teil des Lernprozesses genutzt.
Mögliche Kompetenzentwicklungen	• eigene Programme planen und entwickeln • Sequenzen und einfache Programmstrukturen anwenden • Programme testen, Fehler erkennen und verbessern (Debugging) • digitale Werkzeuge kreativ zur Gestaltung nutzen • eigene Ideen in digitale Projekte umsetzen

	• Zusammenhänge zwischen Eingaben, Befehlen und Ergebnissen erklären • Arbeitsprozesse reflektieren und Ergebnisse präsentieren • kooperativ an digitalen Projekten arbeiten
Ideen und Links zur Unterrichtsreihe	<u>Material 1:</u> Die Einheit führt Lernende spielerisch in die Grundlagen des Programmierens mit der visuellen Programmiersprache ScratchJr ein. In einem analogen Einstieg lernen Lernende zentrale Begriffe wie Programmieren, Befehl und Algorithmus kennen, bevor sie die Oberfläche von ScratchJr erkunden und erste Animationen oder kleine Projekte erstellen. Durch Lernkarten, Video-Tutorials und praktische Aufgaben sammeln die Kinder erste Erfahrungen mit Programmabläufen und entwickeln ein Verständnis dafür, wie digitale Anwendungen durch Schritt-für-Schritt-Anweisungen gesteuert werden. Ziel ist es, grundlegende Programmierkompetenzen sowie logisches und kreatives Denken zu fördern. <u>Weitere Hinweise unter:</u> https://coding-for-tomorrow.de/konzepte/einfuehrung-in-scratch-jr/#1.Einheit <u>Material 2:</u> Die Seite stellt ScratchJr als intuitive, visuelle Programmiersprache für Lernende ab etwa fünf Jahren vor, die einen niedrigschwelligen Einstieg in das Programmieren ermöglicht. Durch farblich codierte Programmierblöcke können Lernende einfache Programme erstellen, Abläufe strukturieren und erste Erfahrungen mit Algorithmen sammeln, ohne sich mit komplexer Syntax auseinandersetzen zu müssen. ScratchJr wird insbesondere für kreative Projekte wie Geschichten, Präsentationen oder kleine Spiele empfohlen und fördert neben Programmierverständnis auch kreativen Ausdruck und Storytelling. Die Plattform bietet Tutorials, Projektideen und Materialien, um den Einstieg in informatisches Denken spielerisch zu gestalten. <u>Weitere Hinweise unter:</u>

	https://digital.tueftellab.de/mod/page/view.php?id=144%2F <u>Material 3:</u> Die Seite bietet eine Übersicht über digitale Lernangebote des Tüftellab im Bereich Programmierung und Making. Lehrkräfte und Lernende finden hier vielfältige Unterrichtskonzepte, Tutorials und Lernmaterialien zu Themen wie Scratch, ScratchJr, Makey Makey, Calliope mini, Robotik und App-Entwicklung. Die Kurse sind nach Schwierigkeitsgrad, Altersstufe und Werkzeugen strukturiert und ermöglichen einen niedrigschwelligen Einstieg ebenso wie vertiefende Projekte. Ziel der Plattform ist es, Maker Education und informatische Bildung praxisnah umzusetzen, indem Lernende digitale Technologien durch eigenes Tüfteln, Programmieren und Experimentieren verstehen und kreative Lösungen entwickeln. <u>Weitere Hinweise unter:</u> https://digital.tueftellab.de/course/index.php?filter_categories%5B0%5D=2%2F
Weitere Links zum Thema	<u>Material 4:</u> Weitere Materialien und Links vom Klett Verlag <u>Weitere Hinweise unter:</u> https://grundschul-blog.de/coding-grundschule-scratchjr-programmieren/