

SCHOOL-SCOUT.DE

Unterrichtsmaterialien in digitaler und in gedruckter Form

Auszug aus:

Einfach Programmieren: Raum und Form

Das komplette Material finden Sie hier:

[School-Scout.de](https://www.school-scout.de)



Inhaltsübersicht

Vorwort 4
 Übersicht der Lernumgebungen 5
 Fachdidaktische Einordnung 6
 Vorstellung der programmierbaren Materialien 11

Lernumgebungen

Ebene Figuren zeichnen

Methodisch-didaktische Anmerkungen 14
 Ebene Figuren zeichnen (Dash®) 19
 Programmvorlage (Dash®) 21
 Figurenkarten (Dash® und Scratch) 22
 Ebene Figuren zeichnen (mTiny) 23
 Programmvorlage (mTiny) 25
 Ebene Figuren zeichnen (Scratch) 26
 Programmvorlage (Scratch) 28
 Wortspeicher 29

Bandornamente

Methodisch-didaktische Anmerkungen 30
 Modul 1 (mTiny): Methodisch-didaktische Anmerkungen 32
 Bandornamente fortsetzen 34
 Bandornamente beschreiben 35
 Bandornamente zeichnen mit mTiny 36
 Bandornamente: Vorlagen 37
 Bandornamente: Ideenwerkstatt 38
 Modul 2 (Scratch): Methodisch-didaktische Anmerkungen 39
 Einfache Bandornamente zeichnen 41
 Komplexe Bandornamente zeichnen 42
 Bandornamente: Vorlagen 43
 Bandornamente entwerfen und untersuchen 44
 Modul 3 (Scratch): Methodisch-didaktische Anmerkungen 45
 Winkelmuster zeichnen 47
 Lösungshinweise 48

Koordinatensystem „Corolina-Freizeitpark“

Methodisch-didaktische Anmerkungen 50
 Wo ist was im Corolina-Freizeitpark? 54
 Nanos Wege durch den Freizeitpark 55
 Code für Nanos Wege durch den Freizeitpark 56
 Nanos Wege durch den Freizeitpark programmieren 57
 Parkplan: Corolina-Freizeitpark 60
 Nanos Wortschatzkarten 61
 Lösungshinweise 62

Netze und Wege in Eckenstadt

Methodisch-didaktische Anmerkungen 65
 Plan von Eckenstadt (Dash® und mTiny) 71
 Evas Schulweg 72
 Evas und Leos Schulweg 73
 RoboterKinder programmieren 74
 Aktionskarten (mTiny) 75
 Aktionskarten blanko (mTiny) 76
 Koordinatenkarten (mTiny) 77
 Dash® in Eckenstadt: Vorbereitung 78
 Dash® in Eckenstadt: Los geht's! 79
 Dash® in Eckenstadt: Ideenwerkstatt 80
 mTiny in Eckenstadt: Vorbereitung 81
 Planquadrante (mTiny) 82
 mTiny in Eckenstadt: Los geht's! 83
 mTiny in Eckenstadt: Ideenwerkstatt 84
 Ozobot® in Eckenstadt: Vorbereitung 85
 Farbcodes (Ozobot®) 86
 Plan von Eckenstadt (Ozobot®) 87
 Plan von Eckenstadt blanko (Ozobot®) 88
 Aktionskarten (Ozobot®) 90
 Ozobot® in Eckenstadt: Los geht's! 91
 Ozobot® in Eckenstadt: Schulwege 92
 Ozobot® in Eckenstadt: Ideenwerkstatt 93
 Plan des Eckenstädter Zoos (Ozobot®) 94

Anhang

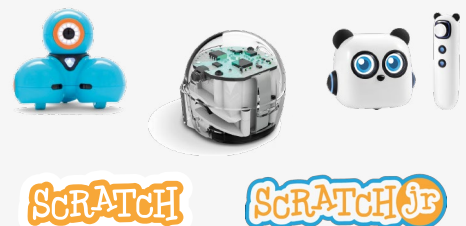
mTiny-Führerschein 95

© scolix Ozobot® © 2022 Ozo Edu, Inc.; Dash® © makevondr.com; Logo Scratch © Scratch Team, CC BY-SA 2.0, verfügbar unter: <https://commons.wikimedia.org/wiki/File:Scratchlogo.svg>; Logo ScratchJr © Lrex, CC BY-SA 4.0, verfügbar unter: https://de.scratch-wiki.info/wiki/Datei:ScratchJrLogoping_mTiny © education.makeblock.com



Download

- ScratchJr-Projekt „CorolinaFreizeitpark“ (.sjr)
- Farbcodes (Ozobot®) (.pdf)
- Lösungsvideo Idas Schulweg (Ozobot®) (.mp4)
- Lösungsvideo Noas Schulweg (Ozobot®) (.mp4)
- Aktions- und Koordinatenkarten (Dash®) (.pdf)
- Plan von Eckenstadt (Dash® und mTiny) (.pdf)
- Plan von Eckenstadt (Ozobot®) (.pdf)



Vorwort

Wozu Maschinen oder Spielzeuge programmieren in der Grundschule? Sie haben doch begrenzte Lebenszyklen. Wenn schon, dann sind nachhaltig bleibende Wissensstrukturen notwendig: das Wissen um Kommunikationsstrukturen im Dialog mit Maschinen. Aber auch wegen des Erlebnisses, eine Maschine korrekt und wie gewünscht reagieren zu sehen. Und weil man dem Ergebnis in der Regel nicht „straight forward“, sondern erst durch mehrfache Versuche und Probeläufe nahekommt. Das ist nicht nur in der Schule so, das ist genauso im professionellen Bereich.

Der Inhaltsbereich „Raum und Form“ eignet sich dazu besonders. Im Gegensatz zum Rechnen lassen sich die Algorithmen umgangssprachlich formulieren. Allerdings werden Unzulänglichkeiten deutlich. Verschärfungen und Präzisierungen werden notwendig, Missverständnisse sind sukzessive auszuschließen.

Man findet bei diesen Versuchen Kriterien für erfolgreiche Sprache in der Mathematik insgesamt. Im Gegensatz zu der literarischen Sprache aus dem Sprachunterricht ist die Sprache in der Mathematik präziser. Was heißt das? Zuerst und wesentlich, dass sie funktional ist: Sie soll kokonstruktive Arbeitsprozesse und Lernprozesse unterstützen. Das umfasst drei Komponenten:

zweckbestimmt, deutungssicher, adressiert.

So entsteht ein Modell zum Kommunizieren. Nun gelten diese Gestaltungskriterien für Dialoge zur Mathematik, speziell zu „Raum und Form“, nicht nur zwischenmenschlich, sondern auch und insbesondere im Dialog mit Maschinen, also beim Programmieren. Maschinen erweisen sich dabei als Dialogpartner mit Eigenheiten, auf die man als Ansprechende/-r eingehen muss, wenn man etwas erreichen möchte. So wie man als Ansprechende/-r zwischenmenschlich ein Adressatenmodell benötigt, ist das auch hier der Fall.

Im Einzelnen:

- ◆ **Zweckbestimmt.** Zu vergegenwärtigen ist, welche Art Aktivität man von der Maschine benötigt und erwarten kann. Das macht man sich zunächst in der eigenen Sprache klar. Nicht verwundern darf dabei, dass dieser eigene Text aus der Sicht der Maschine und ihrer Designer als Pseudocode bezeichnet wird.
- ◆ **Deutungssicher.** Der eigene Text ist darauf zu prüfen, ob er Missverständnisse ausschließt und genau das bezeichnet, was man intendiert. Unausgesprochene Annahmen sind zu reflektieren, Verstehensformen der Maschine zu berücksichtigen.
- ◆ **Adressiert.** Der abgeklärte eigene Text ist in die Sprache der Maschine zu übertragen, der Code entsteht. Grammatische Eigenheiten des Codes sind zu berücksichtigen, Abweichungen werden durch Fehlermeldungen und Aktionsverweigerung quittiert: Maschinen sind keine „großzügig entgegenkommenden Dialogpartner“.

Auch wenn es noch so fremd erscheint, wer Maschinen in diesem Sinne ansprechen will, muss auf ihre Dialekte eingehen, sonst läuft nichts oder es läuft falsch.









Zweierlei Erfahrungen gewinnt man dabei:

1. Auch wenn gefühlte Kuriositäten im Code der Maschine zunächst abstoßend und demotivierend wirken, so lohnt sich das Durchhalten dennoch, nicht nur weil das Angestrebte schließlich gelingt, sondern auch weil das Anpassen an einen weiteren neuen Adressaten leichter und schneller gelingt, wenn einmal eine solche Programmiersituation zum Erfolg geführt hat.
2. Der Transfer auf Formen des zwischenmenschlichen Dialogs ist erfolgversprechend: Auch dort erfordert die Zweckbestimmung eine Vorauseinschätzung dessen, was der Dialogpartner grundsätzlich kann und versteht. Auch dort ist zu prüfen, wo Missverständnisse auftreten können und wie man sie ausschließt. Und schließlich sind auch dort das vorhandene Sprachvermögen oder neu vereinbarte Sprachelemente beim Adressieren der Botschaft einzubeziehen.

„Raum und Form“ in der Grundschule erfordert eine Sprachbildung, die ausgehend von der Umgangssprache und elementaren Erfahrungen zwar nicht durch Formalisierung, wohl aber durch vereinbarte Präzisierungen gekennzeichnet ist. Genau dies ist der zentrale Nutzen des Programmierens in der Grundschule.

Bernd Wollring, Berlin 20. Juli 2022

Übersicht der Lernumgebungen

Lernumgebung	Schwerpunkt	Klasse	Programmierbare Materialien	Dauer
Ebene Figuren zeichnen	Vierecke zeichnen, Konstruktionsschritte beschreiben, Programmierungen unter Vorgaben verändern und Eigenschaften von Figuren überprüfen	2. bis 4. Klasse	mTiny, Dash® und Scratch  	2 bis 4 Schulstunden
Bandornamente	Muster, Parkette und Bandornamente	1. bis 4. Klasse	mTiny, Scratch  	je Modul 2 bis 4 Schulstunden
Koordinatensystem „Corolina-Freizeitpark“	Algorithmen, Punktkoordinaten in der Ebene, Koordinatensysteme, sich im zweidimensionalen Raum orientieren, Wege beschreiben	4. Klasse	Scratch Jr 	2 bis 4 Schulstunden
Netze und Wege in Eckenstadt	sich im zweidimensionalen Raum orientieren, Wege beschreiben, präformale Anbahnung des Koordinatensystems	2. bis 4. Klasse	Dash®, mTiny, Ozobot®   	3 bis 4 Schulstunden

Ozobot © 2022 Ozo EDU, Inc., Dash © makeondercom ; Logo Scratch © Scratch Team, CC BY-SA 2.0, verfügbar unter: <https://commons.wikimedia.org/wiki/File:Scratchlogosvg>; Logo ScratchJr © Lirex, CC BY-SA 4.0, verfügbar unter: <https://descratch-wiki.info/wiki/Datei:ScratchJrLogo.png>; mTiny © education.makeblock.com

Fachdidaktische Einordnung

Algorithmen im Mathematikunterricht der Grundschule

In den Diskursen in den Medien und in den Schulkollegien rund um das Thema digitale Bildung finden sich immer wieder Begriffe wie *21st-century skills*, *computational thinking* oder *4K*.

Doch was genau bedeuten sie konkret für den Mathematikunterricht?

In einem ersten Schritt hat die Kultusministerkonferenz (KMK, 2016) in einem Strategiepapier dazu Stellung genommen. Die dort genannten Kompetenzanforderungen für eine moderne Bildung sind jedoch fachübergreifend und themenunabhängig formuliert. Bestandteil dessen ist u. a. der kompetente Umgang mit Algorithmen, der mangels eines eigenen Fachs Informatik in der Grundschule fachintegriert vermittelt werden soll. Schülerinnen und Schüler sollen in diesem Zusammenhang:

- ◆ Algorithmen erkennen und formulieren, [...]
- ◆ algorithmische Strukturen in genutzten digitalen Tools erkennen und formulieren [und]
- ◆ eine strukturierte, algorithmische Sequenz zur Lösung eines Problems planen und verwenden (KMK, 2016, S. 18).

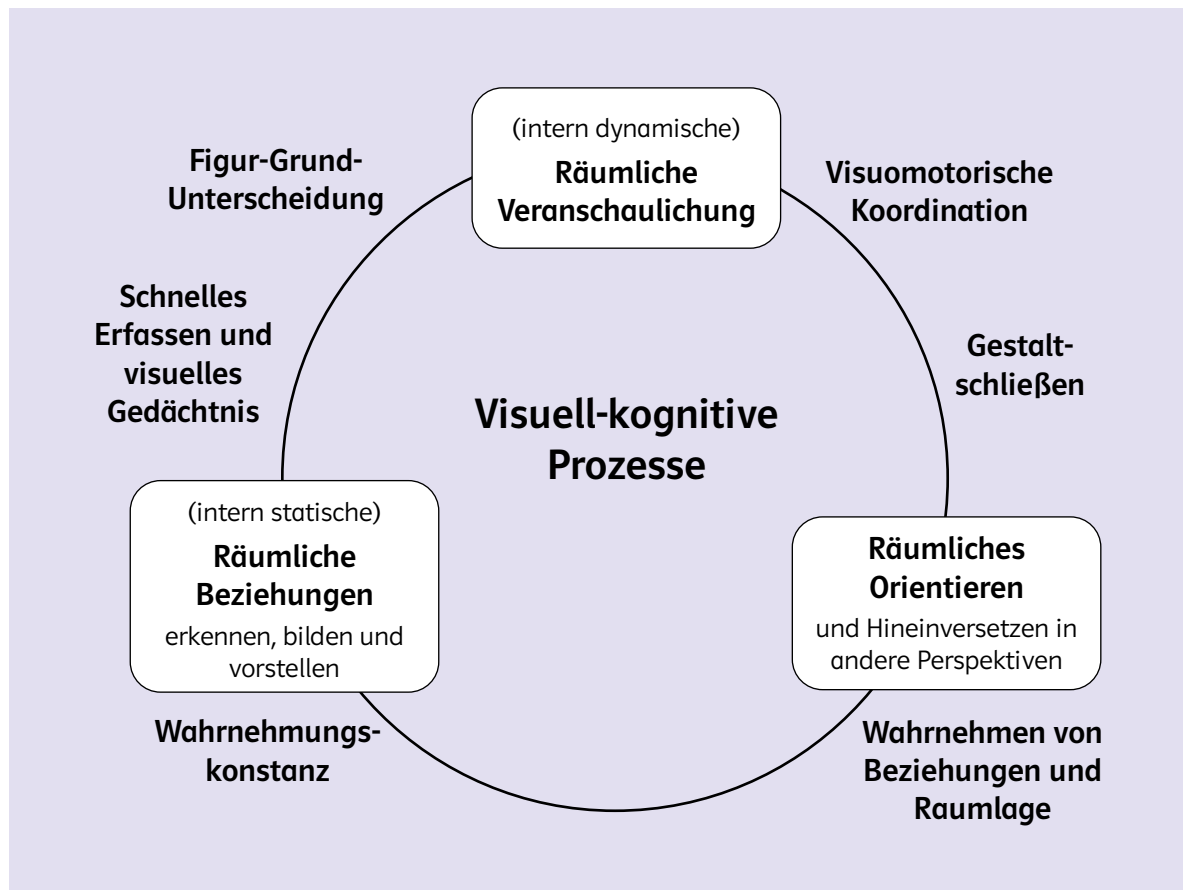
Algorithmen finden sich auch in der Mathematik und sollen hier als fachbezogener Anknüpfungspunkt für die Implementation der allgemeinen KMK-Kompetenzanforderungen dienen. Das Prinzip des Algorithmus ist eine fundamentale Idee des Mathematikunterrichtes. Winter (1976) beschreibt fundamentale Ideen als „Ideen, die starke Bezüge zur Wirklichkeit haben, verschiedene Aspekte und Zugänge aufweisen, sich durch hohen inneren Beziehungsreichtum auszeichnen und in den folgenden Schuljahren immer weiter ausbauen lassen“ (ebd., S. 15). In der Definition als präzise Beschreibung von Aktivität in Form eindeutiger Handlungsvorschriften findet man zahlreiche Anknüpfungspunkte zu Algorithmen im Mathematikunterricht der Grundschule, u. a. schriftliche Normalverfahren, größter gemeinsamer Teiler, kleinstes gemeinsames Vielfaches oder auch Konstruktionsverfahren (Möller, Eilerts, Collignon & Beyer, 2022).

Als zeitgemäßen Zugang zur Förderung dieser Kompetenzen berichten Wilson, Hainey & Connolly (2012) über den vielversprechenden Einsatz von programmierbaren Materialien wie Coding-Apps oder Roboter. Diese verhelfen den Kindern zu Einblicken in die Konzepte von Algorithmen, sind sehr motivierend und handlungsorientiert. Kyriakides & Meletiou-Mavrotheris (2018) arbeiten heraus, dass Programmieraktivitäten Lernenden dabei helfen, besser mit abstrakten mathematischen Ideen umzugehen und diese konkreter zu formulieren. Des Weiteren verbessern sie ihre Problemlösefähigkeiten, das logische Schlussfolgern, das Verständnis für arithmetische und algebraische Prozesse sowie ihre geometrischen Fähigkeiten.

Förderung des räumlichen Wahrnehmungs- und Vorstellungsvermögens

Bei den geometrischen Fähigkeiten ist insbesondere die Entwicklung des räumlichen Wahrnehmungs- und Vorstellungsvermögens hervorzuheben. Bisherige Untersuchungen in diesem Feld verweisen darauf, dass eine mangelhafte Förderung räumlich-visueller Fähigkeiten u. a. mitverantwortlich für Schwierigkeiten von Kindern im Umgang mit arithmetischen Inhalten und Visualisierungen sein kann. Außerdem wird der Kompetenzbereich als bedeutsame Komponente menschlicher Intelligenz angesehen und steht in enger Beziehung zu schulischen Leistungen – nicht nur im Mathematikunterricht (Franke & Reinhold, 2016).

Es gibt verschiedene Modelle zur Beschreibung des räumlichen Wahrnehmungs- und Vorstellungsvermögens. Im Folgenden wird ein Drei-Komponenten-Modell in Anlehnung an Franke & Reinhold (2016) vorgestellt. In der Praxis lassen sich die drei Komponenten zwar nicht eindeutig voneinander trennen, weil Aufgaben immer alle Bereiche ansprechen. Allerdings können sie als Grundlage für die Diagnose und Förderung des räumlichen Wahrnehmungs- und Vorstellungsvermögens dienen.



Drei-Komponenten-Modell der räumlichen Fähigkeiten nach Franke & Reinhold, 2016, S. 85.

Die erste Komponente wird als „**Räumliche Beziehungen**“ bezeichnet. Diese kommt insbesondere bei Bearbeitungsprozessen zum Tragen, die mit gedanklichen Drehungen und Spiegelungen starrer Konfiguration in verschiedenen Positionen arbeiten. Der Fokus der räumlichen Vorstellungsprozesse lässt sich hier als intern statisch beschreiben.

Eine weitere Komponente in diesem Modell ist die „**Räumliche Veranschaulichung**“. Diese kommt insbesondere bei Bearbeitungsprozessen mit gedanklichen Veränderungen von Teilen einer Figur oder der Figur als Ganzes zum Tragen. Das heißt, es wird räumlich etwas verschoben, zerlegt, gefaltet oder vergrößert.

Die dritte Komponente im Modell ist die „**Räumliche Orientierung**“. Dieser Bereich schließt sowohl die Orientierung im wahrgenommenen Raum als auch das gedankliche Hineinversetzen in eine andere Perspektive ein. Hier geht es folglich auch darum, räumliche Beziehungen in Bezug auf den eigenen Körper und dessen räumliche Ausrichtung zu erfassen, sich andere Raumlagen vorzustellen oder eine Rechts-Links-Unterscheidung vorzunehmen.

Unterrichten mit und Gestaltung von Lernumgebungen

Ausgangspunkt der im Folgenden vorgestellten Arbeitsmaterialien ist das Konzept der guten Aufgaben und deren Erweiterung zu Lernumgebungen. Gute Aufgaben ermöglichen ein Lernen in unterschiedlichen Intensitäten am selben Gegenstand und erlauben, unterschiedliche Fähigkeiten, Fertigkeiten, Lösungsideen, Strategien und Darstellungsweisen zu berücksichtigen. Darauf aufbauend fassen Lernumgebungen mehrere dieser (Teil-)Aufgaben unter einem mathematischen bzw. sachbezogenen Leitgedanken zu einer flexiblen, umfangreichen Aufgabenstellung zusammen (Wollring, 2009; Hirt & Wälti, 2016). Das Konzept der Lernumgebungen

- ◆ repräsentiert zentrale Ziele und Inhalte des Mathematikunterrichtes,
- ◆ bietet reichhaltige Möglichkeiten für mathematische Aktivitäten der Lernenden,
- ◆ kann aufgrund der Flexibilität den spezifischen Bedingungen einer Klasse angepasst werden,
- ◆ integriert mathematische und pädagogische Aspekte des Lehrens und Lernens (Krauthausen, 2018, S. 257f.).

Zum Design von Lernumgebungen hat Wollring (2009) sechs Leitideen formuliert. Sie beschreiben die Ganzheit von Lernumgebungen anhand folgender Aspekte:

1. Gegenstand und Sinn
2. Artikulation, Kommunikation und soziale Organisation
3. Differenzierung
4. Logistik
5. Evaluation
6. Vernetzung mit anderen Lernumgebungen

In diesem Sinne sind Lernumgebungen ein Ausformungsrahmen für Lehrende, die durch bewusste, lokale und temporäre Schwerpunktsetzungen konkrete Problemlagen angehen können. Die Leitideen können außerdem auf analoge und digital unterstützte Lernumgebungen angewendet werden.

Neben diesen grundlegenden Charakteristika für das Gesamtdesign der Lernumgebung werden für die Gestaltung der Teilaufgaben die vier Erfahrungen zur Förderung des *computational thinking* nach Kotsopoulos et al. (2017) berücksichtigt. Jede Erfahrungsstufe stellt höhere kognitive Anforderungen an die Lernenden. So ermöglichen die Aufgaben eine zunehmend tiefere Auseinandersetzung mit dem Lerngegenstand. Die Stufenabfolge kann variieren oder es können Erfahrungen wiederholt verwendet werden.

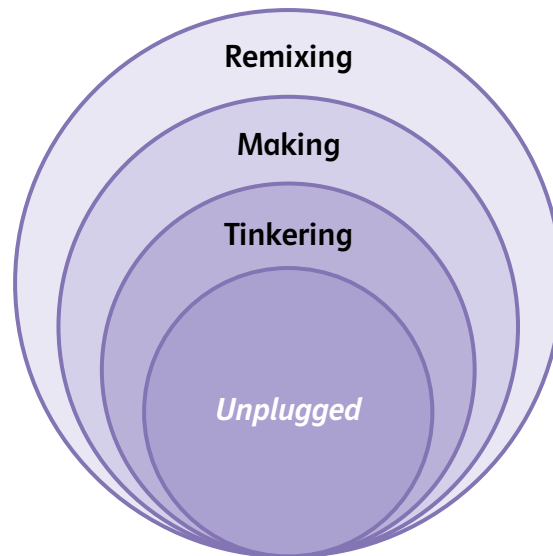
Pädagogische Erfahrungen von Kotsopoulos et al. (2017) nach Komplexität aufsteigend:

- ◆ Die erste Stufe nennt sich „Unplugged“. Hier werden Erfahrungen durch die Auseinandersetzung mit Materialien ohne digitale Unterstützung gemacht.
- ◆ Auf der Stufe „Tinkering“ werden durch Basteln und Herumtüfteln bestehende Objekte unter der Frage „Was, wenn ...?“ modifiziert.

kurz
gesagt

„Lernumgebungen sind flexible große Aufgaben, die in der Regel aus mehreren Teilaufgaben bestehen und durch bestimmte innermathematische oder sachbezogene Leitgedanken zusammengebunden sind.“
(Hirt & Wälti, 2016, S. 13)

- ◆ Daran schließen sich Erfahrungen der dritten Stufe „Making“ an. Hier stehen Aktivitäten im Mittelpunkt, die neue Objekte erschaffen.
- ◆ Die komplexesten Erfahrungen ermöglicht die Stufe „Remixing“. Dabei werden (Teil-)Objekte zur Verwendung in anderen Objekten bzw. für andere Zwecke modifiziert und eingesetzt.



Vier pädagogischen Erfahrungen zur Förderung des „Computational Thinking“ nach Kotsopoulos et al., 2017, S. 159.

Scaffolding des Programmierens

Je nach Vorwissen der Kinder kann im Sinne des Scaffoldings (ausführlich in Möller, Eilerts, Collignon & Beyer, 2022) durch Wahl der Programmiervorlage der Schwierigkeitsgrad variiert werden:

- ◆ Für den Einstieg eignet sich eine fehlerhafte Vorlage zur selbstständigen Korrektur.
- ◆ Für Fortgeschrittene kann die stumme Vorlage genutzt werden, in der zwar durch die Farbe der Blöcke die Befehlskategorie vorgegeben ist, aber die konkrete Handlung noch gefunden werden muss.
- ◆ Für sehr Erfahrene kann auch auf eine Vorlage verzichtet werden und vollkommen frei eine passende Programmierung, z. B. zu den gegebenen Eigenschaften der Figur, gefunden werden.

Scaffolding beschreibt eine bestimmte Art und Weise, wie ein Kind unterstützt werden kann, um bei einer Aufgabe oder Tätigkeit voranzukommen. Der Grundgedanke ist, dass eine gegebene Hilfe gerade ausreicht, um die Aufgabe bewältigen und so ein höheres Bearbeitungsniveau erreichen zu können (Tavassolie & Winsler, 2018)

kurz gesagt

Außerdem kann bei den Vorlagen je nach Vorkenntnissen zusätzlich zwischen einer linearen oder schleifenbasierten Programmierung gewählt werden. Bei den linearen Vorlagen kann sich als Vertiefungsaufgabe die Optimierung und Reduzierung der Programmierblöcke anschließen.

Literaturverweise

Franke, M. & Reinhold, S. (2016). Didaktik der Geometrie in der Grundschule. 3. Auflage. Berlin: Springer Spektrum.

Hirt, U. & Wälti, B. (2016). Lernumgebungen im Mathematikunterricht – Natürliche Differenzierung für Rechenschwache bis Hochbegabte. Seelze: Kallmeyer, Klett Verlag.

Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland – KMK (2016). Bildung in der digitalen Welt – Strategie der Kultusministerkonferenz. Online https://www.kmk.org/fileadmin/Dateien/veroeffentlichungen_beschluesse/2016/2016_12_08-Bildung-in-der-digitalen-Welt.pdf [14.10.2020]

Kotsopoulos, D., Floyd, L., Khan, S. et al. (2017). A Pedagogical Framework for Computational Thinking. *Digital Experiences in Mathematics Education 3* (S. 154–171). <https://doi.org/10.1007/s40751-017-0031-2>

Krauthausen, G. (2018). Einführung in die Mathematikdidaktik – Grundschule. Wiesbaden: Springer Spektrum. <https://doi.org/10.1007/978-3-662-54692-5>

Kyriakides, A. O. & Meletiou-Mavrotheris, M. (2018). When Robot A. L. E. X. Trains Teachers How to Teach Mathematics. In N. Calder, K. Larkin, N. Sinclair (Hrsg.), *Using Mobile Technologies in the Teaching and Learning of Mathematics* (S. 71–72). Cham: Springer.

Möller, R., Eilerts, K., Collignon, P. & Beyer, S. (2022). Zur aktuellen Bedeutung von Algorithmen im Mathematikunterricht – Perspektiven der Digitalisierung. In K. Eilerts, R. D. Möller & T. Huhmann (Hrsg.), *Auf dem Weg zum neuen Mathematik-lehren und -lernen 2.0* (S. 177–193). Wiesbaden: Springer Fachmedien. https://doi.org/10.1007/978-3-658-33450-5_12

Tavassolie, T. & Winsler, A. (2018). Scaffolding. In M. Bornstein (Hrsg.), *The SAGE encyclopedia of lifespan human development* (S. 1885–1886). SAGE Publications. <https://dx.doi.org/10.4135/9781506307633.n701>

Winter, H. (1976). Was soll Geometrie in der Grundschule? *Zentralblatt für Didaktik der Mathematik*, 8, 14–18.

Wilson, A., Hailey, T. & Connolly, T. M. (2012). Evaluation of computer games developed by primary school children to gauge understanding of programming concepts. Beitrag der 6. European Conference on Games-Based Learning (ECGBL), 4–5 October 2012. Cork, Ireland.

Wollring, B. (2009). Zur Kennzeichnung von Lernumgebungen für den Mathematikunterricht in der Grundschule. In A. Peter-Koop, G. Lilitakis & B. Spindeler (Hrsg.), *Lernumgebungen – Ein Weg zum kompetenzorientierten Mathematikunterricht in der Grundschule* (S. 9–23). Offenburg: Mildenerger Verlag.

Vorstellung der programmierbaren Materialien

kurz
gesagt

Blockprogrammierung: Die Befehle der Programmiersprache werden durch Blöcke repräsentiert. Programme werden durch das Aneinanderfügen dieser Blöcke erstellt.

In der Forschung hat sich der Einsatz von Lernrobotern (Educational Robotics; Yanik et al., 2017) und Programmiersoftware als förderlich für die Entwicklung der bereits beschriebenen Kompetenzen erwiesen. Die verfügbaren Materialien unterscheiden sich hinsichtlich der Notwendigkeit des Einsatzes zusätzlicher Endgeräte, des Konstruierens der Roboter, des physischen Vorhandenseins oder der Komplexität der verwendeten Programmiersprache. Grundlage aller im Folgenden vorgestellten Materialien ist eine Blockprogrammierung.

Scratch

Scratch ist eine grafische Programmierumgebung, welche fächerübergreifend im Grundschulunterricht eingesetzt werden kann. Die aneinandergereihten Programmblöcke können direkt in Scratch ausgeführt werden. Scratch wurde mit dem Ziel entwickelt, durch eine einfache und übersichtliche grafische Benutzeroberfläche Kindern und Jugendlichen zu ermöglichen, digitale Geschichten, Spiele und Animationen selbst zu entwickeln und zu gestalten. Dadurch lassen sich Fähigkeiten im Bereich des Problemlösens, des kreativen Ausdrucks und der Kollaboration stärken. Scratch steht als Online- und Offlineversion zur Verfügung ([siehe QR-Code](#)).

Eine Einführung in die Programmierumgebung für Lehrkräfte gibt es u. a. in Markwardt (2021, S. 33 ff.). Hilfreiche Lernvideos zu Scratch bietet die „Coding for Tomorrow Initiative“ unter der folgenden Adresse im Netz an ([siehe QR-Code](#)).

Scratch (und ScratchJr als Teil der Scratch Familie) ist ein Projekt der Scratch Foundation und der Lifelong Kindergarten Group am MIT Media Lab. Es steht kostenlos unter <https://scratch.mit.edu> zur Verfügung.

SCRATCH



ScratchJr

ScratchJr ist eine speziell für jüngere Kinder entwickelte grafische Programmierumgebung, um erste intuitive, kreative und spielerische Programmiererfahrungen zu ermöglichen. Wie bei Scratch ist keine Internetverbindung notwendig. Die grafische Benutzeroberfläche ist jedoch für den Einsatz auf Tablets optimiert und es sind weniger Blöcke verfügbar. Die Funktionen der Blöcke sind durch Symbole dargestellt, um sprachliche Barrieren zu reduzieren. Ein Vorteil für den Einsatz in der Primarstufe ist außerdem die Verwendung eines 20x15-Koordinatensystems. Neben der App für Tablets existiert auch eine Open-Source-Desktopvariante für PC und Mac® ([siehe QR-Code](#)).

SCRATCH JR



SCHOOL-SCOUT.DE

Unterrichtsmaterialien in digitaler und in gedruckter Form

Auszug aus:

Einfach Programmieren: Raum und Form

Das komplette Material finden Sie hier:

[School-Scout.de](https://www.school-scout.de)

