

*Stefan Aufenanger*  
21st Century Skills

Glaubt man den Stimmen aus der (Bildungs-)Politik, so kann allein das Programmieren kommende Generationen fit machen für eine digitalisierte Welt. Nach wie vor verhindern aber unklare Definitionen und Missverständnisse eine konstruktive Diskussion darum, wie sinnvoll eine Integration des Programmierens in Unterricht und schulische Bildung wirklich ist. Eine genauere Untersuchung des Begriffes Programmieren ist daher unumgänglich, und auch ein Rückblick auf erste pädagogische Versuche mit kindgerechter Software sowie die internationale Perspektive helfen dabei, sich dem tatsächlichen Potenzial des Programmierens anzunähern.

*Computer+Unterricht 28 (2017), Heft 107, S. 4–7.*

*Ludger Humbert; Dorothee Müller*  
Pflichtfach Informatik

In der digitalisierten Welt sind wir täglich von Informatik umgeben und auf sie angewiesen. Die Funktionsweise der Informatiksysteme bleibt dabei unverstanden. Die Spezialisten haben den Ruf von sozial inkompetenten „Nerds“. Dabei erfordert und befördert informatisches Arbeiten hohe Kreativität und Kooperation. Im Informatikunterricht erfahren Schüler die komplexen Systeme unserer Lebenswelt als beherrschbar. Darüber hinaus stärken die Schüler ihre sozialen und Problemlösefähigkeiten. Neben einem Lernort in allen Schulen bedarf es gut qualifizierter Informatiklehrkräfte, um den Kompetenzerwerb zu befördern.

*Computer+Unterricht 28 (2017), Heft 107, S. 8–11.*

*Daniel Siebrecht*  
Raus aus dem Computerraum!

Muss Informatikunterricht zwangsläufig in stickigen, mit Technik vollgestellten Computerräumen stattfinden? Autor Siebrecht ruft dazu auf, von der Gerätefixierung wegzukommen. Grundprinzipien des Programmierens lassen sich nicht nur am Rechner erlernen, sondern können in jedem beliebigen Raum und sogar auf dem Pausenhof spielerisch dargestellt und dadurch anschaulich und verständlich gemacht werden. Die Implementierung ist dann nur noch ein Übersetzungsvorgang in eine konkrete Programmiersprache.

*Computer+Unterricht 28 (2017), Heft 107, S. 12–13.*

*Adrian Salamon*  
Spione an der Zimmerdecke?

Wir verlieren mehr und mehr den Bezug zu den technischen Gerätschaften unseres Alltags. Dem selbstverständlichen Umgang mit ihnen steht eine immer größere Unkenntnis über ihre komplexen technischen Funktionsprinzipien gegenüber. Autor Salamon sieht die Informatik in der Verantwortung, dem entgegenzusteuern: Am Beispiel eines vernetzten Rauchmelders gibt er Anregungen, wie über die ethischen und gesellschaftlichen Implikationen des „Internet of Things“ Schülern die mündige Nutzung von Technik nahegebracht werden kann.

*Computer+Unterricht 28 (2017), Heft 107, S. 14–15.*

*Christian Kleinhanß*  
Heiße Drähte in die Zukunft

Das Strategiepapier der Kultusministerkonferenz (KMK) „Bildung in der digitalen Welt“ vom Dezember 2016 regt die Länder an, Schülern die Funktionsweise von und den Umgang mit Algorithmen beizubringen. Christian Kleinhanß führt ein Beispiel an, wie dies in der Praxis altersangemessen und lebensweltorientiert umgesetzt werden kann: An einem Projekttag „Mein heißer Draht zur digitalen Zukunft“ bauten und programmierten Schüler einer 6. Klasse ein Geschicklichkeitsspiel. Dadurch erhielten sie einen handwerklichen und haptischen Eindruck, wie ein Computercode aufgebaut ist.

*Computer+Unterricht 28 (2017), Heft 107, S. 16–18.*

*Christian Kleinhanß*  
Einen Rechner braucht es nicht

Informatische Grundbildung kann an der Schule auch stattfinden, ohne die Schülerinnen und Schüler vor den nächsten Bildschirm zu setzen. Gerade für den Anfang empfiehlt es sich, auf technische Unterstützung weitgehend zu verzichten und auf anderen Wegen Verständnis fürs Programmieren zu vermitteln. In seinem Beitrag erläutert Autor Kleinhanß an Beispielen aus der Unterrichtspraxis, wie in der Jahrgangsstufe 6, aber auch schon in der Grundschule ganz analog in Richtung Informatik gestartet werden kann.

*Computer+Unterricht 28 (2016), Heft 107, S. 19–21.*

Janina Bistron  
Im Programmierfieber

Janina Bistron berichtet von ihren eigenen Lehrerfahrungen mit Grundschulkindern: Die Autorin betreute eine fünfmonatige Programmier-AG, in der Schüler der Klassenstufen 2 bis 4 mit der speziell auf jüngere Kinder zugeschnittenen Software Scratch einfache Computerspiele programmierten. Dabei beobachtete sie positive Auswirkungen auf die Medienkompetenz, das technische Verständnis und die Fähigkeit der Schüler zu selbstgesteuertem Lernen. Im Artikel liefert sie Anleitungen, wie ähnliche Projekte mit Scratch im Unterricht umgesetzt werden können.

*Computer+Unterricht 28 (2017), Heft 107, S. 22–23.*

Hannah Hoffmann; Martin Uckert  
Mit Herz, Hand und Verstand

Dem hohen Bedarf an Arbeitskräften im naturwissenschaftlich-technischen Sektor und der Allgegenwart von technischen Geräten in unserem Alltag steht nur geringes Interesse und Verständnis von Schülern für die sog. MINT-Fächer in der Schule gegenüber. Hoffmann und Uckert berichten vom Projekt MINTplus der TH Bingen, mit dem Schüler ab der 5. Klassenstufe durch spielerisches Lernen, eine regelmäßige Einbindung der technischen Inhalte in den Unterrichtsalltag und einen guten Betreuungsschlüssel für diese Fächer begeistert werden sollen.

*Computer+Unterricht 28 (2017), Heft 107, S. 24–26.*

Heiko Jochum; Klaus Becker; Peter Dauscher; Daniel Jonietz  
Voll digital

Schon seit einigen Jahren gibt es Versuche, Schulbücher digital zu erstellen und verstärkt so genannte OER (Open Educational Resources) im Unterricht einzusetzen – offene Bildungsmaterialien, die kostenlos und frei verwendet, bearbeitet sowie weitergegeben werden können. Das digitale Schulbuch Inf-schule.de lässt sich als eine Kombination aus beiden Ansätzen verstehen. Die Autoren stellen das Projekt vor, das im Wesentlichen durch freiwillige Arbeit außerhalb des Unterrichts entstanden ist.

*Computer+Unterricht 28 (2017), Heft 107, S. 27–29.*

Kathrin Mertes  
Programmieren mit Tynker

Eine Realschule plus in Mainz, drei Nachmittage à 90 Minuten, zehn Kinder aus den Klassenstufen 5 und 6, und als Material die App „Tynker“: Autorin Mertes berichtet aus ihrem Projekt, das zum Ziel hat, den Schülerinnen und Schülern ein erstes Verständnis von Programmierlogik und von informatischen Zusammenhängen zu vermitteln. Die Vorteile von „Tynker“ sieht sie vor allem im spielerisch-kreativen Ansatz des Programms, das es u.a. erlaubt, eigene Spiele und Geschichten zu entwickeln.

*Computer+Unterricht 28 (2017), Heft 107, S. 30–31.*

Holger Schmidt  
Coding-Kenntnisse hamstern

Um Lernende in Bezug auf das Computational Thinking zu fördern, braucht es Unterrichtsarrangements ohne allzu hohe Hürden und mit schnellen Erfolgserlebnissen. Aus eigener Lehrerfahrung, auch im inklusiven Unterricht, empfiehlt Autor Holger Schmidt in seinem Beitrag die deutschsprachige Software „Hamster Simulator“, wo ein virtueller Hamster mithilfe von Programmierbefehlen u.a. Körner suchen und fressen darf. Das Niveau von Aufgabenstellungen ist variabel veränderbar, was besonders in heterogenen Lerngruppen einen großen Vorteil darstellt.

*Computer+Unterricht 28 (2017), Heft 107, S. 32–33.*

Stefan Aufenanger, Ingrid Bounin  
New Kid on the Block: Calliope mini

Als Calliope-mini auf den Markt kam, war die (pädagogische) Fachwelt begeistert und sah gar eine Revolutionierung des Schulsystems gekommen. Aber ist dieser Jubel berechtigt? Die Autoren Stefan Aufenanger und Ingrid Bounin (Landesmedienzentrum Baden-Württemberg) unterziehen das hochgelobte Tool einer kritischen Überprüfung. Daher weist der Beitrag neben praktischen Tipps für eine fachgerechte Wartung und Programmierung auf Vor- und Nachteile von Calliope hin und empfiehlt ggf. Alternativen.

*Computer+Unterricht 28 (2017), Heft 107, S. 34–36.*

Stefan Aufenanger  
Bienen, Kräne, Dinos und Schildkröten

Bereits Kinder im Vorschulalter kann man behutsam ans Thema „Coding“ heranführen: Es gibt sowohl liebevoll gestaltete Brettspiele, die schon ab drei Jahren gespielt werden können, als auch spielerisch gestaltete Anwendungen für Tablets und Notebooks. Autor Stefan Aufenanger schildert in seinem Beitrag eine Reihe spannender Anwendungen fürs Vor- und Grundschulalter – von „Daisy the Dinosaur“ bis „Cargo Bot“.

*Computer+Unterricht 28 (2017), Heft 107, S. 37–40.*

Alexander König  
Combat und Coding

Alexander König stellt mit Kara, Anyone can Learn und der OpenSource-Entwicklung CodeCombat eine Reihe von Programmen vor, die es auf Deutsch bzw. auf Deutsch und Englisch gibt, und die sich für Anfänger wie für Fortgeschrittene, die aufs Coding neugierig sind, gleichermaßen eignen. Die ebenfalls im Beitrag vorgestellte Webseite Computer Science Unplugged bietet darüber hinaus Lernarrangements, die sich ganz ohne unterstützende Technologie durchführen lassen.

*Computer+Unterricht 28 (2017), Heft 107, S. 41.*

Rudolf Peschke, Gerhard Röhner  
Informatik wird zum Thema

Offenbar bewegt sich etwas in der Bildungspolitik – aktuelle Koalitionsverträge der Bundesländer bzw. Regierungsprogramme von Parteien äußern sich ausnahmslos zu Medienbildung, Schulinformatik oder Programmierung. Aber reicht das aus, um die bisherigen Versäumnisse des Bildungssystems auszugleichen? In ihrem Beitrag setzen sich die Autoren Peschke und Röhner mit den aktuellen Entwicklungen auf bildungspolitischer Ebene auseinander und ziehen (kritische) Bilanz.

*Computer+Unterricht 28 (2017), Heft 107, S. 42–43.*