

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/351054675>

# Der Zusammenhang emotionaler Kosten bei Grundschullehramtsstudierenden mit ihrer informatischen Problemlösekompetenz

Article · April 2021

CITATIONS

0

READS

120

4 authors:



[Anke M. Weber](#)

University of Luxembourg

12 PUBLICATIONS 45 CITATIONS

[SEE PROFILE](#)



[Veronika Barkela](#)

Universität Koblenz-Landau

3 PUBLICATIONS 0 CITATIONS

[SEE PROFILE](#)



[Sabrina Dämmer](#)

Universität Koblenz-Landau

3 PUBLICATIONS 1 CITATION

[SEE PROFILE](#)



[Miriam Leuchter](#)

Universität Koblenz-Landau, Landau

79 PUBLICATIONS 321 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Latent Semantic Analysis as a Tool for Formative Assessment [View project](#)



Using the lab for fostering pre-service teachers' learning [View project](#)

# EP

Empirische Pädagogik – 2021 – 35 (1)

---

Timo Reuter, Anke Weber, Sandra Nitz &  
Miriam Leuchter (Hrsg.)

Problemlösen in digitalen Kontexten

**Empirische Pädagogik**

---

**35. Jahrgang 2021**

**1. Heft**

**Herausgeber**

Zentrum für Empirische Pädagogische Forschung (zepf)  
Bürgerstraße 23, 76829 Landau/Pfalz  
Telefon: +49 6341 280 32165, Telefax: +49 6341 280 32166  
E-Mail: sekretariat@zepf.uni-landau.de  
Homepage: <http://www.zepf.uni-landau.de>

**Verlag**

Empirische Pädagogik e. V.  
Bürgerstraße 23, 76829 Landau/Pfalz  
Telefon: +49 6341 280 32180, Telefax: +49 6341 280 32166  
E-Mail: [info@vep-landau.de](mailto:info@vep-landau.de)  
Homepage: <http://www.vep-landau.de>

**Druck**

DIFO Bamberg

**Zitiervorschlag**

Reuter, T., Weber, A., Nitz, S. & Leuchter, M. (Hrsg.). (2021). Problemlösen in digitalen Kontexten (Empirische Pädagogik, 35 (1), Themenheft). Landau: Verlag Empirische Pädagogik.

Alle Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung, werden vorbehalten. Kein Teil des Werks darf in irgendeiner Form (durch Fotografie, Mikrofilm oder ein anderes Verfahren) ohne schriftliche Genehmigung des Verlags reproduziert oder unter Verwendung elektronischer Systeme verbreitet werden.

ISSN 0931-5020

ISBN 978-3-944996-73-8

© Verlag Empirische Pädagogik, Landau 2021

## Inhalt

### Editorial

Timo Reuter, Anke Weber, Sandra Nitz & Miriam Leuchter Problemlösen in digitalen Kontexten.....	5
--	---

### Originalarbeiten

Tobias Przywarra, Alexander Engl & Björn Risch Entwicklung eines Messinstruments zur Qualitätsbestimmung von (digitalen) Anschauungsmodellen.....	19
Florian Gigl, Eva Cauet, Samuel Greiff & Alexander Kauertz Entwicklung eines Instruments zur Erfassung physikbezogener komplexer Problemlösefähigkeit.....	38
Morten Bastian, Yannick Schneider & Andreas Mühling Diagnose von Fehlvorstellungen bei der Ablaufverfolgung von Programmen in einem webbasierten Testsystem .....	72
Anke M. Weber, Veronika Barkela, Sabrina Stiel-Dämmer & Miriam Leuchter Der Zusammenhang emotionaler Kosten bei Grundschullehrramtsstudierenden mit ihrer informatischen Problemlösekompetenz .....	93
Christoph Thyssen, Alexandra Pankow, Kristine Klaeger & Daria Chernyak Kompetenzen und Nutzungsperspektiven von Lehrkräften zum Einsatz digitaler Medien zur Erkenntnisgewinnung im naturwissenschaftlichen Unterricht .....	112
Impressum .....	136

# Contents

## Articles

- Tobias Przywarra, Alexander Engl & Björn Risch  
Development of a measuring instrument to determine the quality of  
(digital) illustrative models..... 19
- Florian Gigl, Eva Cauet, Samuel Greiff & Alexander Kauertz  
Development of an instrument for complex problem solving in physics..... 38
- Morten Bastian, Yannick Schneider & Andreas Mühling  
Diagnosis of misconceptions by tracing programs in a web-based test  
environment..... 72
- Anke M. Weber, Veronika Barkela, Sabrina Stiel-Dämmer & Miriam Leuchter  
The relation of primary school student teachers' emotional costs and  
their informatic problem solving ..... 93
- Christoph Thyssen, Alexandra Pankow, Kristine Klaeger & Daria Chernyak  
Competencies and usage perspectives in the field of digital media  
knowledge acquisition among science teachers ..... 112

Anke M. Weber, Veronika Barkela, Sabrina Stiel-Dämmer & Miriam Leuchter

## Der Zusammenhang emotionaler Kosten bei Grundschullehramtsstudierenden mit ihrer informatischen Problemlösekompetenz

Computational Thinking als Aspekt des informatischen Problemlösens wird zunehmend als wichtige Kompetenz angesehen. Diese kann schon bei Grundschulkindern angebahnt werden, allerdings ist der Unterricht zu Computational Thinking in der Grundschule bisher nur vereinzelt gegeben. Ein möglicher Grund dafür könnte die Unsicherheit bezüglich Informatik und Programmieren auf Seiten der Grundschullehrkräfte sein. Deshalb wurden in einem Seminar Kompetenzen des Computational Thinking an Grundschullehramtsstudierende vermittelt, mit dem Ziel, mögliche Ängste abzubauen. In diesem Rahmen wurden interindividuelle Unterschiede bezüglich der informatischen Problemlösekompetenz und des logischen Schlussfolgerns als Voraussetzungen eines Abbaus dieser emotionalen Kosten untersucht. 70 Studierende nahmen an der Untersuchung mit Prä- und Posterhebung teil. Eine multiple Regression zeigte keine Abnahme der emotionalen Kosten, allerdings waren die emotionalen Kosten durch die informatische Problemlösekompetenz bedingt. Studierende mit höherer Kompetenz hatten geringere emotionale Kosten. Die Kompetenz des logischen Schlussfolgerns hatte keinen Einfluss. Die Ergebnisse deuten darauf hin, dass in der Lehre von Computational Thinking verstärkt auf die Ängste der Studierenden eingegangen werden sollte.

Schlagwörter: Computational Thinking – Emotionale Kosten – Grundschullehramtsstudierende – Informatische Problemlösekompetenz – Logisches Denken

## The relation of primary school student teachers' emotional costs and their informatic problem solving

Computational thinking as an aspect of informatic problem-solving is considered to be an increasingly important competence. Computational thinking can already be fostered in primary school children, but its teaching in primary school remains sparse. A possible reason might be primary school teachers' uncertainty regarding computer science and programming. Therefore, we developed a university course to teach the implementation of computational thinking to primary school student teachers with the goal of reducing possible emotional costs such as anxiety. Within this framework, we examined interindividual differences in informatic problem solving and logical thinking as prerequisites for reduction in these emotional costs. 70 students participated in the study with pre- and a post-assessment. A multiple regression showed that students' emotional costs remained unchanged, however, informatic problem solving had a negative influence. Students with higher informatic problem solving competencies reported lower emotional costs. Logical thinking had no effect. The results indicate that students' anxiety should be considered when teaching computational thinking.

Keywords: computational thinking – emotional costs – informatic problem solving competency – logical thinking – primary school student teachers

## 1 Einleitung

In einer zunehmend digitalisierten Welt kommen Kinder bereits früh mit Medien in Kontakt und nutzen programmierte Geräte wie Smartphones und Computer, verstehen aber nicht, wie sie funktionieren (Feierabend, Rathgeb, Reutter, 2019). Deshalb wird verstärkt diskutiert, ob und unter welchen Bedingungen informatisches Problemlösen, z. B. in Form von Programmieren, bereits in der Grundschule unterrichtet werden sollte (Lamprou & Repenning, 2018; Yadav, Mayfield, Zhou, Hambrusch & Korb, 2014). Das informatische Problemlösen führt die Schüler\*innen in informatische mentale Werkzeuge ein, die als Computational Thinking (CT) bezeichnet werden können, z. B. algorithmisches Denken (Wing, 2006). Diese mentalen Werkzeuge sind jedoch nicht nur im Kontext des informatischen Problemlösens einsetzbar, sondern können auch als Kulturtechnik verstanden werden (Bers, 2018). In internationalen Studien konnte gezeigt werden, dass Kindern CT schon in der Grundschule nähergebracht werden kann (Angeli et al., 2016; Di Lieto et al., 2017; Dickses, Sengupta, Farris & Basu, 2016; OECD, 2018).

Für einen solchen Unterricht benötigen Grundschullehrkräfte fachliches Wissen, fachdidaktisches Wissen und pädagogisches Wissen, das miteinander verknüpft angelegt und erworben werden muss (Kleickmann & Hardy, 2019). Bezüglich des informatischen Problemlösens wird dieses Amalgam als Technological Pedagogical Content Knowledge (TPACK; Angeli et al., 2016) bezeichnet, welches unter anderem grundlegendes Wissen in der Anwendung einer Programmiersprache (in dieser Studie NEPO®), Kompetenzen im CT und Kompetenzen bzgl. Unterstützungsmaßnahmen beim informatischen Problemlösen umfasst. Einstellungen und individuelle Ängste seitens der Lehramtsstudierenden können jedoch den Erwerb von TPACK beeinflussen und die Umsetzung eines solchen Unterrichts verhindern (Shapiro & Williams, 2012). In welchem Zusammenhang Ängste und Programmier-Kompetenzen von angehenden Grundschullehrkräften stehen, wurde bisher wenig erforscht. Deren Untersuchung verspricht jedoch wichtige Erkenntnisse, die in die Implementation von entsprechenden Lerninhalten in die universitäre Lehrer\*innenbildung einfließen können. Wir untersuchen deshalb, ob Lehramtsstudierende im Rahmen eines Seminars zum Thema CT beim Abbau von Ängsten in Bezug auf das Programmieren unterstützt werden können.

## 2 Problemlösen in informatischen Kontexten

Nach Newell und Simon (1972) ist ein Problem ein gegebener Zustand, der in einen Zielzustand verändert werden soll. Der/die Problemlöser\*in kennt jedoch keinen offensichtlichen Weg, um vom gegebenen Zustand in den Zielzustand zu gelangen und benötigt mentale Werkzeuge für diese Transition (Reed, 2017).



Als mentale Werkzeuge identifiziert Mayer (2013) Phasen und Subprozesse des Problemlösens, die in komplexen Problemsituationen nicht linear verlaufen müssen. In der ersten Phase wird ein mentales Modell zur Repräsentation des Problems aufgebaut. Die zweite Phase unterteilt sich in die drei Subprozesse Planen des Problemlöseprozesses, Ausführen des Plans und Evaluieren der Ausführung. Diese Prozesse münden in Problemlösestrategien wie z. B. Planen eines Lösungswegs mit Teilzielen, Analogien finden und Analyse der Mittel-/Endpunkte des Problems (Mayer, 1998; Newell, Shaw & Simon, 1959a; Reed & Vallacher, 2020).

Für die Koordination von informatischen Problemlöseprozessen werden verschiedene spezifische mentale Werkzeuge benötigt (u. a. Zerlegen, Abstrahieren, Sequenzieren eines Algorithmus, Kontrollfluss, Fehler finden und beheben und Generalisieren; White & Sivitanides, 2003). Diese werden in der Informatikdidaktik als Aspekte des CT verstanden und entsprechen in weiten Teilen dem informatischen Problemlösen im Sinne von Software-Entwicklungsmodellen (Bers, 2018; Sarker, Faruque, Hossen & Rahman, 2015).

Die systematische Förderung von CT dient dazu, Kinder an informatisches Problemlösen heranzuführen. Damit steht nicht das Bedienen bestimmter Endgeräte oder das Beherrschen bestimmter Programmiersprachen im Vordergrund, sondern eine Anwendung von CT als mentale Werkzeuge zum Verständnisaufbau und zur Anwendung des informatischen Problemlösens (Angeli et al., 2016; Bers, 2018).

## **2.1 Computational Thinking**

Im Folgenden werden sechs Aspekte des CT aus der informatikdidaktischen Literatur und Software-Entwicklungsmodellen entsprechend synthetisiert (Angeli et al., 2016; Sarker et al., 2015; Selby & Woollard, 2014). Dabei orientieren wir uns an den Aspekten nach Angeli et al. (2016) und stellen das Zerlegen in Teilprobleme, das Abstrahieren, das Sequenzieren eines Algorithmus, den Kontrollfluss sowie das Fehlerentdecken und -beheben und das Generalisieren in den Vordergrund. Nach Modellen der Softwareentwicklung sowie MINT-didaktischen Prozessmodellen werden diese Aspekte nicht als zwingende Reihenfolge verstanden und einzelne Aspekte können ggf. übersprungen oder wiederholt ausgeführt werden.

Die Förderung von grundlegenden Kompetenzen im Bereich CT soll die Studierenden dazu befähigen, Kinder an informatisches Problemlösen heranzuführen. Diese Kompetenz soll mit der Anwendung einer Programmiersprache und Unterstützungsmaßnahmen im informatischen Unterricht vernetzt aufgebaut werden. Diese Vorgehensweise entspricht den aktuellen Anforderungen an eine Lehrer\*innenbildung, in der vermieden werden soll, träges Wissen aufzubauen (vgl. Kleickmann & Hardy, 2019). Deshalb wird CT im Kontext von fallbasierten grundschulpädagogisch ausgerichteten aufgabenbezogenen Settings verknüpft erarbei-

tet. Einzelne Aspekte von CT werden daher im Folgenden in den Kontext der Aufgabe „Fahrradlicht“ gestellt, mit exemplarischer Programmierung erörtert und mit lernunterstützenden Maßnahmen verbunden, die Schüler\*innen bei Schwierigkeiten unterstützen können.

### **2.1.1 Zerlegen**

Komplexe Aufgaben werden in Teilprobleme zerlegt. Diese werden dabei so aufgliedert, dass sie entweder leichter zu verstehen und zu lösen sind (Newell, Shaw & Simon, 1959b) oder dass Lösungsstrategien bereits bekannt sind (Faber, Wierdsma, Doombos, Van der Ven, & Vette, 2017). Anschließend können mehrere Teilprobleme zusammengesetzt werden. Beispielsweise kann die Lösung der Aufgabe „Erstelle ein automatisches Fahrradlicht, bei dem ausgewählt werden kann, ob die Lampen durchgehend leuchten oder blinken.“ in drei Teilprobleme zerlegt werden:

1. die Lampe leuchtet durchgehend.
2. die Lampe blinkt.
3. Bedienungsmenü, in dem „durchgehend leuchten“ oder „blinken“ eingestellt werden kann.

Das Zerlegen in Teilprobleme ist relevant, um die Komplexität des Problems zu reduzieren (Angeli et al., 2016). Lernende können Schwierigkeiten mit dem Verstehen der Aufgabe und dem Zerlegen in Teilprobleme haben und bedürfen ggf. spezifischer Hilfen. So könnten sie gefragt werden, was sie unter dem Begriff „automatisch“ verstehen, oder sie könnten, wenn sie Teilproblem 1 und 2 nicht unterscheiden können, zwei Lampen bekommen, von denen eine leuchtet und eine blinkt.

### **2.1.2 Abstrahieren**

Durch Abstrahieren werden die hinter einer Aufgabe liegenden Algorithmen aufgedeckt. Für das Abstrahieren müssen irrelevante Eigenschaften ausgeschlossen werden. Im grundschulpädagogischen Kontext werden abstrakte Probleme durch konkrete Aufgaben veranschaulicht (vgl. Wing, 2006), Komplexität reduziert (Angeli et al., 2016), und das eigentliche Problem fokussiert (Curzon, Dorling, Ng, Selby & Woollard, 2014).

Die Aufgabe „das Fahrradlicht soll leuchten, wenn es dunkel wird“ ist ein Beispiel für das abstrakte Problem „wenn Lichtsensor unter einer bestimmten Zahl, dann Aktion“. Um diese Aufgabe zu lösen, ist es notwendig herauszufinden, wie eine logische Abfolge von Befehl und Aktion umzusetzen ist. Zur Programmierung des Fahrradlichts muss also erkannt werden, dass ein Licht leuchten soll, es jedoch irrelevant ist, ob gelb oder rot.

Den Kindern könnte es schwerfallen, eine wenn-dann-Funktion zu erstellen. Die Lehrkraft könnte sie dabei unterstützen, indem sie ihnen zeigt, wo diese im Alltag

zu finden ist. Als Beispiel könnte sie den Kühlschrank heranziehen, da das Licht im Kühlschrank immer angeht, wenn man die Tür öffnet (vgl. Angeli et al., 2016).

### **2.1.3 Sequenzieren eines Algorithmus**

Das Sequenzieren eines Algorithmus wird benötigt, um das Programm fehlerfrei verfassen zu können (Angeli et al., 2016). Jeder Schritt einer gewünschten Aktion muss detailliert im Computerprogramm definiert werden. Eine Schwierigkeit dabei ist, zu erkennen, aus welchen Schritten ein Algorithmus besteht (Angeli et al., 2016; Curzon et al., 2014).

Die einzelnen Schritte der Aufgabe „das Fahrradlicht soll leuchten, wenn es dunkel wird“ wären z. B.

1. lasse Lampe leuchten,
2. mache 1., wenn Wert des Lichtsensors unterhalb 30 % Lichtstärke ist,
3. trifft 2. nicht zu, leuchtet Lampe nicht,
4. wiederhole die Schritte 1. – 3. unendlich oft.

Nach Zhang und Nouri (2019) können Kinder mit Hilfe von Analogien aus dem täglichen Leben, wie z. B. To-Do-Listen oder Bedienungsanleitungen, das Sequenzieren erlernen (vgl. auch Angeli et al., 2016).

### **2.1.4 Kontrollfluss**

Damit ein Algorithmus die Aufgabe wie gewünscht ausführen kann, ist es nicht nur notwendig, jeden Schritt zu identifizieren und zu definieren, sondern diese auch in der richtigen Reihenfolge anzuordnen und auf ihre Funktionalität hin zu überprüfen (Angeli et al., 2016).

Bei der oben genannten Aufgabe wird durch eine erste Ausführung des Programms überprüft, ob die geplanten Schritte in dieser Reihenfolge ablaufen können.

Um die Kinder bei der Überprüfung des Kontrollflusses zu unterstützen, kann die Lehrkraft mit den Kindern die verschiedenen Programmmentwürfe vergleichen und evaluieren, um unterschiedliche Kontrollflüsse sichtbar zu machen (Faber et al., 2017).

### **2.1.5 Fehler entdecken und beheben**

Anschließend wird die gewünschte Aktion ausgeführt und überprüft, ob diese gelingt. Fehler im Algorithmus und im Kontrollfluss werden identifiziert und beseitigt. Ein sicherer Umgang mit den einzelnen Befehlsbausteinen ist dabei ausschlaggebend (Hsu, Chang & Hung, 2018).

Ein gängiger Fehler bei der Aufgabe „das Fahrradlicht soll leuchten, wenn es dunkel wird“ ist, dass der Befehl „wiederhole unendlich oft“ nicht programmiert wurde. So würde der Befehl nur einmal verarbeitet werden.

Zur Unterstützung können Kinder Teile der Programme ausführen und mit kleinen Hilfen, wie z. B. Fragen stellen, darauf aufmerksam gemacht werden, dass Fehler vorhanden sind und wo sie zu finden sein könnten (vgl. Angeli et al., 2016).

### **2.1.6 Generalisieren**

Generalisieren ist für ein flexibles Lösen von verschiedenen informatischen Problemen relevant, dabei werden gefundene Lösungen auf weitere Aufgabenstellungen mit gleichen Eigenschaften übertragen (Angeli et al., 2016; Selby, 2014). Wenn Aufgabenstellungen auf das zugrundeliegende Problem abstrahiert werden können, gelingt es leichter, auch in anderen Aufgabenstellungen dieses abstrakte Problem zu entdecken. So können Probleme leichter präzisiert sowie Lösungsstrategien synergetisch genutzt werden (Curzon et al., 2014).

In unserem Beispiel kann der Algorithmus nicht nur für das Programmieren eines Fahrradlichts verwendet, sondern auch auf einen Bewegungsmelder oder eine Alarmanlage für ein Kinderzimmer übertragen werden.

Die Lehrkraft kann mit den Kindern reflektieren und gemeinsam überlegen, ob die Lösung auf ähnliche Probleme übertragen werden kann (vgl. Angeli et al., 2016). Beim Fahrradlicht könnte der Code z. B. auf einen Lichtschalter im Kühlschrank übertragen werden.

Die hier vorgestellten sechs CT-Aspekte werden den Studierenden in einem Seminar nähergebracht, um sie grundlegend mit dem Themenbereich CT vertraut zu machen. So sollen sie befähigt werden, CT zu unterrichten und die angewandten niedrigschwelligen Beispielaufgaben (z. B. ein Fahrradlicht programmieren) sollen die Studierenden langsam und schrittweise an das Thema heranzuführen.

## **2.2 Zusammenhänge zwischen informatischer Problemlösekompetenz und logischem Schlussfolgern sowie emotionalen Kosten**

Der Erwerb von CT-Kompetenzen stellt Studierende vor große Herausforderungen. Bei naturwissenschaftlichen oder technischen Fachthemen wie z. B. Optik kann teils auf Allgemeinbildung zurückgegriffen werden, was bei CT vermutlich nicht der Fall ist (Hsu et al., 2018). Deshalb sind das Wissen und die Bereitschaft, sich mit CT auseinanderzusetzen, möglicherweise gering und ein Gefühl der Überforderung schnell erreicht (Bescherer & Fest, 2019). Dies kann sich durch Angst vor dem Fach zeigen.

### **2.2.1 Emotionale Kosten**

Vor allem Studentinnen berichten von Angst in MINT-Fächern (Marsh et al., 2019). Diese können unter anderem durch einen Stereotype Threat begründet sein. Dieser internalisierte Glaube, nur geringe Fertigkeiten zu besitzen, betrifft vor allem Frauen

und kann zu Ängsten und zur Ablehnung von der Beteiligung an MINT-Fächern führen (Shapiro & Williams, 2012). Bezüglich des Programmierens wurden emotionale Kosten wie bspw. Angst bereits als mögliche Faktoren aufgedeckt, welche mit der Leistung einer Person zusammenhängen können. So haben gemäß dem Review von Nolan und Bergin (2016) Computerangst und Programmierleistungen einen negativen Zusammenhang. Demnach haben selbst Studierende eines informatischen Fachs oft Angst vor dem Programmieren. Nach Marsh et al. (2019) haben die emotionalen Kosten einen größeren Effekt auf die Wahl eines MINT-Fachs an der Universität als die in der Schule erbrachten Leistungen in MINT-Fächern. Da das Grundschullehramt zum großen Teil von weiblichen Studierenden belegt wird, ist es relevant, den Zusammenhang der emotionalen Kosten mit informatischen Problemlösekompetenzen zu untersuchen.

### **2.2.2 Logisches Schlussfolgern**

Das logische Schlussfolgern kann mit dem informatischen Problemlösen zusammenhängen. So fand Fletcher (1984) z. B., dass logisches Schlussfolgern wie das Lösen von Matrizen und räumliches Vorstellungsvermögen positiv mit Programmierleistungen zusammenhängt. Shute (1991) kam in ihrer Studie zu einem ähnlichen Ergebnis. Das informatische Problemlösen wird mit dem logischen Schlussfolgern in Bezug auf die emotionalen Kosten in Verbindung gesetzt, da es sich um eine relevante interindividuelle Lernvoraussetzung handelt.

Eine Auseinandersetzung mit angstbesetzten Inhalten kann zu einer Reduktion der Angst führen (vgl. Wannemüller, 2018). Im Schulkontext wurde dies für die Mathängstlichkeit mehrfach belegt (vgl. Dowker, Sarkar & Looi, 2016). Deshalb haben wir ein Seminar entwickelt, in dem sich Grundschullehramtsstudierende mit CT befassen.

## **2.3 Seminarkonzept**

Ziel des Seminars ist der Abbau von Ängsten hinsichtlich CT und die Einführung in zentrale Aspekte des TPACK, welches im Unterricht zur Förderung von informatischer Problemlösekompetenz in der Grundschule eine Rolle spielen kann. Dazu gehören grundlegende Kompetenzen in der Anwendung einer Programmiersprache, CT und Unterstützungsmaßnahmen im informatischen Unterricht.

### **2.3.1 Unterstützungsmaßnahmen**

Hier wird der Fokus auf kognitive Aktivierung gelegt, da diese sich als lernwirksames Merkmal guten Unterrichts herausgestellt hat (Klieme, Funke, Leutner, Reimann & Wirth, 2001). Kognitive Aktivierung ist z. B. gekennzeichnet durch das Provozieren kognitiver Konflikte und das Einfordern von Begründungen der Lernenden (Kleickmann, 2012). Da davon auszugehen ist, dass der Einsatz kognitiv aktivierender

Maßnahmen im Grundschulunterricht auch für Themen im Bereich des CT lernwirksam ist, analysieren die Studierenden Videos, in denen unterschiedliche Strategien der kognitiven Aktivierung zu sehen waren (vgl. Seidel, Stürmer, Blomberg, Kobarg & Schwindt, 2011). Darüber hinaus erhalten die Studierenden eine Einführung in Möglichkeiten des Diagnostizierens von Vorwissen von Schüler\*innen (Möller, 1999) und der Rückmeldung hinsichtlich ihrer Lernprozesse (vgl. Hardy, Jonen, Möller & Stern, 2006). Die Studierenden entwickeln konkrete Beispiele, wie sie Kinder sinnvoll kognitiv aktivieren, ihr Vorwissen erfragen und ihnen Rückmeldung zu ihrem Problemlöseprozess geben können.

### **2.3.2 CT und Anwendung einer Programmiersprache**

Das Programmieren mit Calliope Mini® wird im Kontext von grundschulspezifischen Aufgaben erlernt. Calliope Mini® ist ein Mikrocontroller, der für Bildungszwecke entwickelt wurde und sich durch seine graphische Programmiersprache und sein einfaches Design auszeichnet. Den Studierenden werden die CT-Kompetenzen theoretisch erklärt und durch eine Einführung ins Programmieren und das anschließende Schreiben eigener Programme für den Calliope Mini® nähergebracht. Die Studierenden erhalten über das Semester zehn aufeinander aufbauende virtuelle Programmieraufgaben. Sie programmieren z. B. einen Smiley, einen Würfel, ein Fahrradlicht und eine Alarmanlage, die mehrere wenn-dann-Befehle beinhalten.

## **2.4 Fragestellungen**

Der Literatur zu (Mathe-)Ängstlichkeit folgend, kann eine Auseinandersetzung mit angstbesetzten Inhalten zu einer Abnahme der Angst führen (vgl. Dowker et al., 2016; Wannemüller, 2018). Daher beschäftigt sich diese Pilotstudie mit dem Effekt eines Seminars zum CT auf emotionale Kosten, wie Angst vor dem Programmieren. Außerdem wird ein möglicher Zusammenhang mit der informatischen Problemlösekompetenz und dem logischen Schlussfolgern untersucht (Nolan & Bergin, 2016).

Daraus lassen sich folgende Fragestellungen ableiten:

1. Verändern sich die emotionalen Kosten der Studierenden in Bezug auf CT über die Zeit des Seminarbesuchs?
2. Hat die informatische Problemlösekompetenz zu Beginn des Seminars einen Effekt auf die emotionalen Kosten?
3. Hat logisches Schlussfolgern einen Effekt auf die emotionalen Kosten?

## **3 Methode**

### **3.1 Stichprobe**

Insgesamt nahmen 70 Grundschullehramtsstudierende (61 weiblich, Alter  $M = 24.87$ ,  $SD = 3.77$ ) an der Untersuchung teil, von denen alle das beschriebene Masterseminar besuchten. Die Frage, ob sie in der Schule bereits programmieren gelernt hätten, beantworteten 7 mit trifft zu, 4 mit trifft überwiegend zu, 4 mit trifft wenig zu und 55 mit trifft nicht zu. Eine ähnliche Verteilung ergab sich auf die Frage, ob sie bereits vor dem Seminar Programme geschrieben hätten. 7 antworteten mit trifft zu, 4 mit trifft überwiegend zu, 6 mit trifft wenig zu und 53 mit trifft nicht zu. Die Studierenden hatten also größtenteils kaum oder keine Erfahrungen mit dem Programmieren.

### **3.2 Studiendesign**

Die Studie bestand aus einer Prä- und einer Posterhebung. Die Präerhebung wurde durchgeführt, nachdem die Studierenden eine kurze Einführung in das Thema Programmieren erhalten hatten. Die Posterhebung wurde in der letzten Seminarwoche durchgeführt.

### **3.3 Messinstrumente**

#### **3.3.1 Emotionale Kosten**

Die emotionalen Kosten wurden mit einer auf Deutsch übersetzten und für das Programmieren adaptierten Version der Items zur Matheängstlichkeit aus der PISA-Studie erhoben. Die Items basieren auf Wigfield und Meece (1988). Insgesamt wurden sechs Items mit einer vierstufigen Likert Skala von 0 (trifft gar nicht zu) bis 3 (trifft genau zu) eingesetzt. Die Items erfragten die Gefühle von Sorge, Nervosität und Hilflosigkeit sowie Anspannung in Bezug auf das Programmieren (Tabelle 1). Höhere Werte deuten auf höhere emotionale Kosten hin. Die emotionalen Kosten wurden sowohl zu Beginn als auch zum Ende des Seminars erhoben.

#### **3.3.2 Informatische Problemlösekompetenz**

Die Erfassung der informatischen Problemlösekompetenz zu Beginn des Seminars erfolgte mit dem Testsystem Progry (Bastian, Schneider & Mühling, 2021). Die Studierenden erhielten insgesamt 9 Aufgaben, die jeweils eine Figur auf einem  $8 \times 8$  Felder großen Gitternetz zeigt, die mit den Pfeiltasten einer Computertastatur bewegt werden kann. Den Studierenden wurde für jede Aufgabe ein anderer graphischer Programmiercode vorgegeben, den sie durch Tasteneingabe ausführen sollten. Item 1 beschreibt eine Schrittabfolge, bei Item 2 wird ein Wiederhole-Baustein eingefügt und Item 3 zeigt zwei ineinander geschachtelte Wiederhole-Bausteine. Ab Item 4

werden bedingte Wiederhole-Bausteine („Wiederhole solange“) mit der Bedingung „IstGelb?“ eingefügt, für die Teile des Rasters gelb eingefärbt sind. Dies definiert, ob Befehle ausgeführt und wie oft eine Befehlsabfolge wiederholt werden sollte, solange die Bedingung erfüllt ist. Die Items 4 bis 9 basierten auf solchen „IstGelb?“-Bedingungen mit unterschiedlichen Variationen von Wiederhole- und Falls-Bausteinen. Jede richtig gelöste Aufgabe gibt einen Punkt, sodass ein Maximum von 9 Punkten erreicht werden kann.

Die Studierenden erhielten keine Rückmeldung über ihr Abschneiden. Für eine ausführliche Beschreibung des Testsystems verweisen wir auf den Artikel von Bastian et al. (2021).

### 3.3.3 Logisches Schlussfolgern

Die Items zur Messung des logischen Schlussfolgerns (The International Cognitive Ability Resource Team, 2014) wurden zum Ende des Seminars erhoben. Insgesamt wurden 10 Items eingesetzt, 5 Matrizen und 5 Mentale Rotationsitems. Die Matrizen bestanden aus 9 Feldern, von denen 8 je ein Bild zeigten und eins ein Fragezeichen. Unter der Matrize waren 6 weitere Bilder abgebildet. Die Studierenden sollten aus den 6 Bildern das passende Bild heraussuchen, um das zugrundeliegende Muster der Matrize logisch zu vervollständigen. Die Mentalen Rotationsitems zeigten links je einen Würfel, von dem drei Seiten zu sehen waren. Daneben waren 6 weitere Würfel abgebildet, aus denen die rotierte Version des ersten Würfels herausgesucht werden sollte. Zudem gab es die Möglichkeit „keiner der Würfel ist eine Rotation“ und „ich weiß die Antwort“ nicht auszuwählen. Jedes richtig gelöste Item ergibt einen Punkt, sodass ein Maximum von 10 Punkten erreicht werden kann. Es wurde ein Mittelwert über alle Items gebildet, sodass ein Wert von 0 auf ein niedriges und ein Wert von 1 auf ein hohes Niveau des logischen Schlussfolgerns hinweist.

## 4 Ergebnisse

Die emotionalen Kosten wurden sowohl zu Beginn,  $M = 1.18$ ,  $SD = 0.72$ ,  $\alpha = .89$ , als auch zum Ende des Seminars erhoben,  $M = 1.13$ ,  $SD = 0.62$ ,  $\alpha = .83$  (Tabelle 1 und Abbildung 1).



Tabelle 1: Deskriptive Statistiken der emotionalen Kosten nach Items

Item	Beginn		Ende	
	M	SD	M	SD
Ich Sorge mich, dass ich die Programmieraufgaben nur schwer lösen kann.	1.37	0.82	1.41	0.79
Ich bin angespannt, wenn ich Programmieraufgaben lösen muss.	1.26	0.96	1.00	0.80
Ich bin nervös, wenn ich mich mit Problemen im Bereich Programmieren beschäftige.	1.00	0.82	0.99	0.84
Ich fühle mich hilflos, wenn ich vor einem Programmierproblem stehe.	1.21	0.92	1.13	0.85
Ich Sorge mich, dass ich wegen schlechter Programmierleistungen das Seminar nicht bestehe.	1.21	1.01	1.21	1.05
Ich werde nervös, wenn ich daran denke, dass ich Kindern das Programmieren beibringen muss.	1.00	0.87	1.06	0.80

Anmerkungen: M = Mittelwert, SD = Standardabweichung

Für die Kontrollvariable informatisches Problemlösen wurden die Items 1, 2 und 5 aus den Analysen ausgeschlossen, da sie nicht mit den anderen Items der Skala korrelierten. Aus den restlichen sechs Items wurde eine Mittelwertsvariable von 0 (keine Aufgabe richtig gelöst) bis 1 (alle Aufgaben richtig gelöst) gebildet,  $M = 0.43$ ,  $SD = 0.29$ ,  $\alpha = .70$ . Für die Kontrollvariable logisches Schlussfolgern wurde ebenfalls eine Mittelwertsvariable von 0 bis 1 gebildet,  $M = 0.51$ ,  $SD = 0.22$ ,  $\alpha = .69$ .

Die Korrelation zwischen den emotionalen Kosten zu Beginn und zum Ende des Seminars,  $r = .49$ ,  $p < .001$ , deutet darauf hin, dass die Rangreihe der Personen relativ stabil bleibt. Wilcoxon-Rank-Tests zeigten zu Beginn keine Geschlechtsunterschiede in den emotionalen Kosten,  $W = 196$ ,  $p = .170$ , am Ende hatten die Studentinnen höhere emotionale Kosten im Vergleich zu den Studenten,  $W = 145$ ,  $p = .023$ .

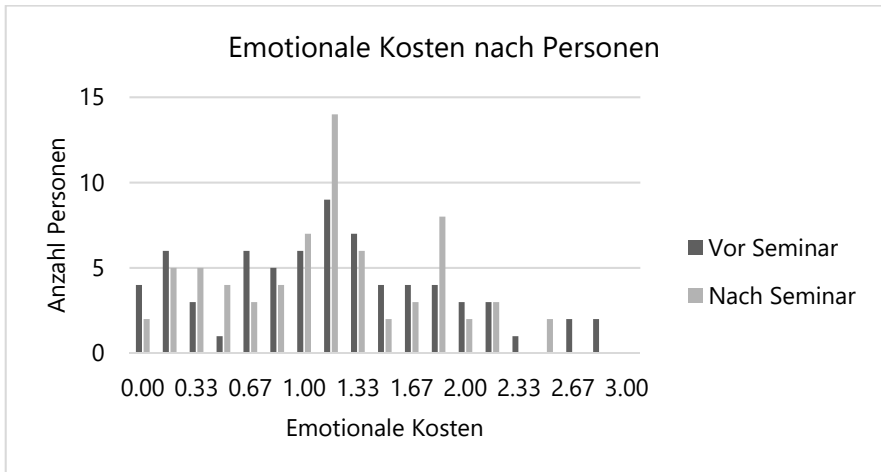


Abbildung 1: Vergleich der emotionalen Kosten zu Beginn und am Ende des Seminars nach Personen.

Als nächstes wurden die Zusammenhänge der emotionalen Kosten mit den Kontrollvariablen informatische Problemlösekompetenz und logisches Schlussfolgern überprüft (Tabelle 2). Es zeigten sich negative Zusammenhänge mit beiden Kontrollvariablen.

Tabelle 2: Korrelationen zwischen emotionalen Kosten, Problemlösekompetenz und logischem Schlussfolgern

	EK Prä	EK Post	PK
EK Post	.49***		
PK	-.38***	-.54***	
LS	-.29**	-.33**	.57***

Anmerkungen. PK = Problemlösekompetenz, EK = Emotionale Kosten, LS = Logisches Schlussfolgern, \*\* $p < .01$ , \*\*\* $p < .001$

Für die folgenden Berechnungen wurden die Skalen z-standardisiert. Zur Prüfung der Veränderung der emotionalen Kosten der Studierenden über das Semester (Fragestellung 1) wurde zunächst ein gepaarter t-Test gerechnet,  $t(69) = 0.52$ ,  $p = .602$ ,  $d_{\text{Cohen}} = 0.07$ . Das Seminar führte also nicht zu einer Abnahme der Angst der Studierenden. Danach wurde eine lineare Regression der emotionalen Kosten zu Beginn auf die emotionalen Kosten am Ende spezifiziert (Tabelle 3, Modell 1). Die emotio-

nalen Kosten zu Beginn erklärten 23 % der Varianz in den emotionalen Kosten am Ende.

Um den Einfluss der informatischen Problemlösekompetenz und des logischen Schlussfolgerns auf die Varianz der emotionalen Kosten am Ende zu prüfen (Fragestellungen 2 und 3), wurden sie in die Regression aufgenommen (Tabelle 3, Modell 2). Die Voraussetzungen für eine multiple Regression waren gegeben, keine Hinweise auf Homoskedastizität oder Multikollinearität,  $VIF < 10$ . Die informatische Problemlösekompetenz zu Beginn wirkte sich negativ auf die emotionalen Kosten am Ende aus und hatte im Vergleich zu den emotionalen Kosten zu Beginn einen größeren Einfluss. Zudem nahm der Effekt der emotionalen Kosten zu Beginn des Seminars nach Einführung der Kontrollvariablen im Vergleich zu Modell 1 ab. Logisches Schlussfolgern hatte keinen Effekt auf die emotionalen Kosten zum Ende des Seminars. Modell 2 konnte 36 % der Varianz in den emotionalen Kosten zum Ende des Seminars erklären.

Tabelle 3: Regressionsanalysen zur Vorhersage der emotionalen Kosten am Ende

Modell	Prädiktor	Emotionale Kosten am Ende				
		$\beta$	SE	t	$R_{\text{korrr}}^2$	F
1	EK Beginn	.49***	.11	4.67	.23	21.78***
2						
	EK Beginn	.33**	.10	3.21		
	PK	-.42***	.12	-3.48		
	LS	.13	.12	0.11		
					.36	14.16***

Anmerkungen:  $\beta$  = standardisiertes Regressionsgewicht, SE = Standardfehler, T = T-Wert,  $R_{\text{korrr}}^2$  = korrigiertes  $R^2$ , F = F-Wert, EK = Emotionale Kosten, PK = informatische Problemlösekompetenz, LS = logisches Schlussfolgern, \*\*p < .01. \*\*\*p < .001

## 5 Diskussion

CT kann bereits im Grundschulalter unterrichtet werden (Dickes, Sengupta, Farris & Basu, 2016; Di Lieto et al., 2017), allerdings haben Grundschullehrkräfte oft Ängste in Bezug auf das informatische Problemlösen, die sie abhalten, CT zu unterrichten (vgl. Marsh et al., 2019; Nolan & Bergin, 2016). Um diesen emotionalen Kosten entgegenzuwirken, ist die Lehre zum Thema CT während des Studiums sinnvoll. Aller-

dings existieren bisher wenige Studien in diesem Bereich. In unserer Studie haben wir interindividuelle Voraussetzungen für den Abbau von Ängsten hinsichtlich CT während eines Seminars untersucht.

Die erste Fragestellung beschäftigte sich mit einer potentiellen Abnahme der emotionalen Kosten im Verlauf eines Seminars zum Thema CT. Die emotionalen Kosten nahmen jedoch nicht ab, sondern blieben im Mittel gleich. Dies kann allerdings ein Artefakt sein, begründet dadurch, dass die Studierenden bereits erste Erfahrungen im CT gesammelt hatten, da die erste Erhebung in dieser Pilotstudie nach einer Einführung in das Thema stattfand. Ob der Zeitpunkt der Befragung einen Einfluss hatte, soll in einer weiteren Studie getestet werden, in der die erste Befragung zu Beginn des Seminars stattfindet, bevor die Studierenden mit dem Thema CT in Berührung gekommen sind.

Für das Gleichbleiben der emotionalen Kosten könnte außerdem die Zusammensetzung der Stichprobe relevant sein, die zu einem großen Teil aus Studentinnen bestand. Am Ende berichteten sie, im Vergleich zu den männlichen Teilnehmenden, von höheren emotionalen Kosten, sodass ein Stereotype Threat vorliegen könnte (Marsh et al., 2019; Shapiro & Williams, 2012). Vermutlich ist es im Seminar nicht gelungen, den Studentinnen die Angst vor dem Programmieren und CT zu nehmen. Am Ende des Artikels werden didaktische Überlegungen zum Umgang damit angestellt.

Die zweite Fragestellung beschäftigte sich mit einem möglichen Einfluss der informatischen Problemlösekompetenz auf die emotionalen Kosten. Die Prüfung dieses Zusammenhangs zeigte, dass geringere informatische Problemlösekompetenzen mit höheren berichteten emotionalen Kosten sowohl zu Beginn als auch am Ende des Seminars einhergehen (vgl. Schmader, Johns & Forbes, 2008). Die Prüfung dieses Effekts ergab, dass sich eine geringe informatische Problemlösekompetenz in stärkerem Ausmaß am Ende des Seminars als zu dessen Beginn auswirkt. Dies deutet darauf hin, dass eine höhere Kompetenz potentiell die emotionalen Kosten verringert. Dies könnte so interpretiert werden, dass Studierende mit einer höheren informatischen Problemlösekompetenz sich dieser bewusst sind und deswegen während des Programmierens weniger emotionale Kosten erleben.

Die dritte Fragestellung befasste sich mit einem möglichen Einfluss von logischem Schlussfolgern auf die emotionalen Kosten. Es zeigten sich negative Zusammenhänge mit den emotionalen Kosten zu Beginn und zum Ende des Seminars und ein positiver Zusammenhang mit der informatischen Problemlösekompetenz. Dies impliziert, dass Studierende mit einer höheren Kompetenz im logischen Schlussfolgern weniger emotionale Kosten erleben. Da informatisches Problemlösen von der Kompetenz im logischen Schlussfolgern bedingt ist (Fletcher, 1984; Shute, 1991), ist es

möglich, dass eine höhere Kompetenz im logischen Schlussfolgern den Studierenden mehr Sicherheit im Programmieren gab. Es zeigte sich allerdings kein größerer Einfluss auf die emotionalen Kosten zum Ende über die emotionalen Kosten zu Beginn des Seminars und die informatische Problemlösekompetenz hinaus. Dies deutet darauf hin, dass das logische Schlussfolgern für einen Abbau der Angst weniger relevant ist als die tatsächliche Kompetenz im Programmieren. Das kann dadurch erklärt werden, dass das informatische Problemlösen einen stärkeren Bezug zu den emotionalen Kosten während des Programmierens hat als das logische Schlussfolgern.

Diese Studie zeigt zwar Zusammenhänge zwischen informatischer Problemlösekompetenz und emotionalen Kosten hinsichtlich des Programmierens, gibt jedoch keine Hinweise darauf, dass die emotionalen Kosten sich verringert haben. Weitere Studien zu Seminaren sollten so gestaltet sein, dass es möglich ist, Moderator- oder Mediatorvariablen zu finden, die mit der informatischen Problemlösekompetenz interagieren und einen tieferen Einblick in die Gründe für die emotionalen Kosten und der Möglichkeit, ihnen entgegenzuwirken, bieten.

Aus den Ergebnissen lässt sich jedoch schließen, dass es in diesem Seminar nicht gelungen ist, Ängste bezüglich CT bei Grundschullehramtsstudierenden abzubauen. Dennoch ist dies ein wichtiges Ziel der universitären Lehrer\*innenbildung. Dafür könnte es wichtig sein, den Ängsten der Studierenden frühzeitig zu begegnen. Eine Möglichkeit könnte eine niedrigschwellige Einführung in das Thema sein. So könnten die Studierenden zunächst über kleine Logikspiele an CT herangeführt werden. Schulbesuche, in denen die Studierenden Kinder in Interaktion mit Calliope Mini® beobachten können, könnten ihnen das Spielerische und die Freude der Kinder näherbringen und so Ängste reduzieren.

Forschung zur (Mathe-)Ängstlichkeit konnte zeigen, dass eine Auseinandersetzung mit den Ängsten zu deren Abbau führen kann (vgl. Dowker et al., 2016; Wannemüller, 2018). So könnte die Reduktion der didaktischen Inhalte des Seminars zugunsten von CT-Inhalten dazu führen, dass sich die Studierenden stärker auf den Erwerb der informatischen Problemlösekompetenzen konzentrieren und mehr Sicherheit in diesen gewinnen können.

Eine weitere Möglichkeit der Angst zu begegnen, ist, die Studierenden anzuleiten, ihre Ängste zu reattribuieren. So fanden Supekar, Iuculano, Chen und Menon (2015) heraus, dass ein Tutorium Matheängstlichkeit verbessern konnte. Im Rahmen des Seminars könnte Studierenden, denen das Programmieren schwerfällt, ein zusätzliches Tutorium angeboten werden, in dem gemeinsam programmiert wird und sie ihre Fragen stellen können. Zudem kann den Studierenden die Möglichkeit gegeben werden, sich über ihre Ängste mit den Dozierenden auszutauschen, was helfen kann,

Ängste zu reattribuieren. Vor allem im Hinblick auf den Stereotype Threat könnte es wichtig sein, die Studierenden über diesen aufzuklären. Dies könnte dazu führen, dass die Ängste nicht auf die vermeintlich mangelnde Kompetenz attribuiert werden (Johns, Schmader & Martens, 2005). Außerdem könnte die Dozentin über ihre eigenen Erfahrungen berichten und erklären, dass sie zu Beginn auch Unsicherheit bezüglich des Programmierens verspürt hat, die jedoch durch eine intensivere Beschäftigung mit dem Thema abgebaut werden konnte. Dabei ist es wichtig, dass die Dozentin als Modell fungiert und sicheres Verhalten in Bezug auf das Programmieren zeigt (Wannemüller, 2018). So fanden Jamieson, Mendes, Blackstock und Schmader (2010), dass bereits ein Hinweis darauf, dass Anspannung auch positive Effekte im Sinne von Leistungssteigerung haben kann, zu einer Verbesserung der Leistung führen kann.

Diese Maßnahmen sollen in einem weiteren Seminar eingesetzt werden, um zu prüfen, ob ein Abbau der Ängste und Unsicherheiten und ein damit erhöhter Lernerfolg im Bereich CT erreicht werden kann.

## Anmerkung

Wir danken der Deutschen Telekom Stiftung für die Förderung des Projektes „Die Zukunft des MINT-Lernens: Digitale Lernumgebungen in Sachunterricht und Mathematik“.

## Literatur

- Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J. & Zagami, J. (2016). A K-6 computational thinking curriculum framework: Implications for teacher knowledge. *Educational Technology & Society*, 19 (3), 47-57.
- Bastian, M., Schneider, Y. & Mühling, A. (2021). Diagnose von Fehlvorstellungen anhand von Programmabläufen in einem elektronischen Testsystem. Im Druck.
- Bers, M. U. (2018). Coding and computational thinking in early childhood: The impact of ScratchJr in Europe. *European Journal of STEM Education*, 3 (3), 08. <https://doi.org/10.20897/ejsteme/3868>.
- Bescherer, C. & Fest, A. (2019). Mathematik und informatische Bildung. Programmieren mit Scratch. In T. Junge & N. Horst (Hrsg.), *Schriftenreihe Medienpädagogik interdisziplinär: Vol. 12. Digitale Medien in der Grundschullehrerbildung. Erfahrungen aus dem Projekt dileg-SL*. (S. 117-130). München: Verlag kopaed.
- Curzon, P., Dorling, M., Ng, T., Selby, C. & Woollard, J. (2014). Developing computational thinking in the classroom: A framework. Swindon, GB. Verfügbar unter: <https://eprints.soton.ac.uk/369594/1/DevelopingComputationalThinkingInTheClassroomaFramework.pdf>.
- Dickes, A. C., Sengupta, P., Farris, A. V. & Basu, S. (2016). Development of mechanistic reasoning and multilevel explanations of ecology in third grade using agent-based models. *Science Education*, 100 (4), 734-776. <https://doi.org/10.1002/sce.21217>.
- Di Lieto, M. C., Inguaggiato, E., Castro, E., Cecchi, F., Cioni, G., Dell'Omo, M., Laschi, C., Pecini, C., Santerini, G., Sgandurra, G. & Dario, P. (2017). Educational robotics intervention on executive functions in preschool children: A pilot study. *Computers in Human Behavior*, 71, 16-23. <https://doi.org/10.1016/j.chb.2017.01.018>.

- Dowker, A., Sarkar, A. & Looi, C. Y. (2016). Mathematics anxiety: What have we learned in 60 years? *Frontiers in Psychology*, 7, 508. <https://doi.org/10.3389/fpsyg.2016.00508>.
- Faber, H. H., Wierdsma, M. D., Doornbos, R. P., Van der Ven, J. S. & Vette, K. de (2017). Teaching computational thinking to primary school students via unplugged programming lessons. *Journal of the European Teacher Education Network*, 12, 13-24.
- Feierabend, S., Rathgeb, T. & Reutter, T. (2019). KIM Studie 2018. Kindheit, Internet, Medien. Basisuntersuchung zum Medienumgang 6-bis 13-Jähriger. Verfügbar unter: [https://www.mpf.de/fileadmin/files/Studien/KIM/2018/KIM-Studie\\_2018\\_web.pdf](https://www.mpf.de/fileadmin/files/Studien/KIM/2018/KIM-Studie_2018_web.pdf).
- Fletcher, S. H. (1984). *Cognitive abilities and computer programming*. Los Angeles: University of California.
- Hardy, I., Jonen, A., Möller, K. & Stern, E. (2006). Effects of instructional support within constructivist learning environments for elementary school students' understanding of 'floating and sinking'. *Journal of Educational Psychology*, 98 (2), 307-326. <https://doi.org/10.1037/0022-0663.98.2.307>.
- Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, 126, 296-310. <https://doi.org/10.1016/j.compedu.2018.07.004>.
- International Cognitive Ability Resource Team (2014). <https://icar-project.com/> [02.05.2020].
- Jamieson, J. P., Mendes, W. B., Blackstock, E. & Schmader, T. (2010). Turning the knots in your stomach into bows: Reappraising arousal improves performance on the GRE. *Journal of Experimental Social Psychology*, 46 (1), 208-212. <https://doi.org/10.1016/j.jesp.2009.08.015>.
- Johns, M., Schmader, T. & Martens, A. (2005). Knowing is half the battle: Teaching stereotype threat as a means of improving women's math performance. *Psychological Science*, 16 (3), 175-179. <https://doi.org/10.1111/j.0956-7976.2005.00799.x>.
- Kleickmann, T. (2012). Kognitiv aktivieren und inhaltlich strukturieren im naturwissenschaftlichen Sachunterricht.
- Kleickmann, T. & Hardy, I. (2019). Vernetzung professionellen Wissens angehender Lehrkräfte im Lehramtsstudium. *Unterrichtswissenschaft*, 47 (1), 1-6. <https://doi.org/10.1007/s42010-018-00035-2>.
- Klieme, E., Funke, J., Leutner, D., Reimann, P. & Wirth, J. (2001). Problemlösen als fächerübergreifende Kompetenz. Konzeption und erste Resultate aus einer Schulleistungsstudie. *Zeitschrift für Pädagogik*, 47 (2), 179-200.
- Lamprou, A. & Repenning, A. (2018). Teaching how to teach computational thinking. Paper presented at the annual conference for Innovation and Technology in Computer Science Education (ITiCSE), Larnaca, Cyprus. <https://doi.org/10.1145/3197091.3197120>.
- Marsh, H. W., Van Zanden, B., Parker, P. D., Guo, J., Conigrave, J. & Seaton, M. (2019). Young women face disadvantage to enrollment in university STEM coursework regardless of prior achievement and attitudes. *American Educational Research Journal*, 56 (5), 1629-1680. <https://doi.org/10.3102/00028312.18824111>.
- Mayer, R. E. (1998). *Thinking, problem solving, cognition* (2<sup>nd</sup> ed.). New York: W. H. Freeman and Company.
- Mayer, R. E. (2013). Problem solving. In D. Reisberg (Ed.), *The Oxford handbook of cognitive psychology*. Oxford: Oxford University Press. <https://doi.org/10.1093/oxfordhb/9780195376746.013.0048>
- Möller, K. (1999). Konstruktivistisch orientierte Lehr-Lernprozessforschung im naturwissenschaftlich-technischen Bereiches des Sachunterrichts. In W. Köhnlein (Hrsg.), *Vielperspektivisches Denken im Sachunterricht* (S. 125-191). Bad Heilbrunn: Klinkhardt.
- Newell, A., Shaw, J. C. & Simon, H. A. (1959a). Report on a general problem-solving program. Pittsburgh: Carnegie Institute of Technology.
- Newell, A., Shaw, J. C. & Simon, H. A. (1959b). A variety of intelligent learning in a general problem solver. Paper presented at the Interdisciplinary Conference on Self-Organizing Systems. Chicago Illinois.
- Newell, A. & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs: Prentice-Hall.
- Nolan, K. & Bergin, S. (2016). The role of anxiety when learning to program: A systematic review of the literature. Koli Calling '16 Proceedings of the 16<sup>th</sup> Koli Calling International Conference on Computing Education Research, Koli National Park, Finland. <https://doi.org/10.1145/2999541.2999557>.

- OECD. (2018). Computational thinking. In A. Paniagua & D. Istance (Eds.), *Educational research and innovation. teachers as designers of learning environments: The importance of innovative pedagogies* (pp. 101-108). Paris: OECD Publishing.
- Reed, S. K. (2017). Problem solving. In S. E. F. Chipman (Ed.), *The Oxford handbook of cognitive science* (Vol. 1). Oxford: Oxford University Press. <https://doi.org/10.1093/oxfordhb/9780199842193.013.10>.
- Reed, S. K. & Vallacher, R. R. (2020). A comparison of information processing and dynamical systems perspectives on problem solving. *Thinking & Reasoning*, 26 (2), 254-290. <https://doi.org/10.1080/13546783.2019.1605930>.
- Sarker, I. H., Faruque, F., Hossen, U. & Rahman, A. (2015). A survey of software development process models in software engineering. *International Journal of Software Engineering and Its Applications*, 9 (11), 55-70. <https://doi.org/10.14257/ijseia.2015.9.11.05>.
- Schmader, T., Johns, M. & Forbes, C. (2008). An integrated process model of stereotype threat effects on performance. *Psychological Review*, 115 (2), 336-356. <https://doi.org/10.1037/0033-295X.115.2.336>.
- Seidel, T., Stürmer, K., Blomberg, G., Kobarg, M. & Schwindt, K. (2011). Teacher learning from analysis of videotaped classroom situations: Does it make a difference whether teachers observe their own teaching or that of others? *Teaching and Teacher Education*, 27 (2), 259-267. <https://doi.org/10.1016/j.tate.2010.08.009>.
- Selby, C. C. (2014). How can the teaching of programming be used to enhance computational thinking skills? [PhD thesis]. Southampton: University of Southampton.
- Selby, C. C. & Woollard, J. (2014). Refining an understanding of computational thinking. Southampton: University of Southampton. <http://eprints.soton.ac.uk/id/eprint/372410>.
- Shapiro, J. R. & Williams, A. M. (2012). The role of stereotype threats in undermining girls' and women's performance and interest in STEM fields. *Sex Roles*, 66 (3-4), 175-183. <https://doi.org/10.1007/s1199-011-0051-0>.
- Shute, V. J. (1991). Who is likely to acquire programming skills? *Journal of Educational Computing Research*, 7 (1), 1-24. <https://doi.org/10.2190/VQJD-T1YD-5WVB-RYPJ>.
- Supekar, K., Iuculano, T., Chen, L. & Menon, V. (2015). Remediation of childhood math anxiety and associated neural circuits through cognitive tutoring. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience*, 35 (36), 12574-12583. <https://doi.org/10.1523/JNEUROSCI.0786-15.2015>.
- Wannemüller, A. (2018). Spezifische Phobien. In J. Margraf & S. Schneider (Hrsg.), *Lehrbuch der Verhaltenstherapie* (Band 2, S. 29-47). Berlin: Springer.
- White, G. & Sivitanides, M. (2003). An empirical investigation of the relationship between success in mathematics and visual programming courses. *Journal of information systems education. Journal of Information Systems Education*, 14 (4), 409-416.
- Wigfield, A. & Meece, J. L. (1988). Math anxiety in elementary and secondary school students. *Journal of Educational Psychology*, 80 (2), 210. <https://doi.org/10.1037/00220663.80.2.210>.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49 (3), 33-35.
- Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S. & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education*, 14 (1), 1-16. <https://doi.org/10.1145/2576872>.
- Zhang, L. & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education*, 141, 103607. <https://doi.org/10.1016/j.compedu.2019.103607>.



#### Autorinnen

Anke Weber, M.Sc., Universität Koblenz-Landau, Institut für Erziehungswissenschaft

Veronika Barkela, Dipl. Sozialwissenschaftlerin, Universität Koblenz-Landau, Institut für Erziehungswissenschaft

Sabrina Stiel-Dämmer, zweites Staatsexamen für das Lehramt an Grund- und Hauptschulen, Universität Koblenz-Landau, Institut für Erziehungswissenschaft

Prof. Dr. Miriam Leuchter, Universität Koblenz-Landau, Institut für Erziehungswissenschaft

Korrespondenz an: [weber-a@uni-landau.de](mailto:weber-a@uni-landau.de)

## Impressum

### **Empirische Pädagogik**

Zeitschrift zu Theorie und Praxis erziehungswissenschaftlicher Forschung

ISSN 0931-5020

Bürgerstraße 23, 76829 Landau/Pfalz

Telefon: +49 6341 280 32180; Telefax: +49 6341 280 32166

E-Mail: [ep@vep-landau.de](mailto:ep@vep-landau.de)

Homepage: <http://www.vep-landau.de>

### **Beirat**

Roland Arbinger (Landau), Susanne Buch (Wuppertal), Monika Buhl (Heidelberg), Marten Clausen (Duisburg), Andreas Frey (Mannheim), Julia Fluck (Aachen), Michaela Gläser-Zikuda (Erlangen-Nürnberg), Reinhold S. Jäger (Landau), Gisela Kammermeyer (Landau), Katharina Maag Merki (Zürich), Jörn Sparfeldt (Saarbrücken), Olga Zlatkin-Troitschanskaia (Mainz).

### **Redaktion**

Gabriele E. Dlugosch, Ingmar Hosenfeld, Heike Liepelt, Sarah Poersch, Inga Wagner, Michael Zimmer-Müller.

### **Beiträge**

Die Empirische Pädagogik veröffentlicht Beiträge zu folgenden Rubriken: Originalarbeiten, Forschungsmethoden, Forum, Historische Seite und Rezensionen. Beiträge nimmt die Redaktion entgegen.

### **Manuskriptgestaltung**

Die aktuellen Autorenrichtlinien finden Sie im Internet unter <http://www.vep-landau.de>. Im Übrigen müssen die eingereichten Beiträge den „Richtlinien zur Manuskriptgestaltung“ (Göttingen: Hogrefe, 2017) entsprechen. Abgabe des Manuskripts per E-Mail als Word- oder pdf-Datei. Endfassung (Text und Abbildungen) auf CD-ROM oder als E-Mail-Anhang.

### **Erscheinungsweise/Preis**

Die Zeitschrift erscheint viermal jährlich. Der Bezugspreis beträgt € 45,00/Jahr zzgl. Porto. Kündigung bis spätestens 8 Wochen vor Jahresende.

## Editorial

- Timo Reuter, Anke Weber, Sandra Nitz & Miriam Leuchter: Problemlösen in digitalen Kontexten

## Originalarbeiten

- Tobias Przywarra, Alexander Engl & Björn Risch: Entwicklung eines Messinstruments zur Qualitätsbestimmung von (digitalen) Anschauungsmodellen
- Florian Gigl, Eva Cauet, Samuel Greiff & Alexander Kauertz: Entwicklung eines Instruments zur Erfassung physikbezogener komplexer Problemlösefähigkeit
- Morten Bastian, Yannick Schneider & Andreas Mühling: Diagnose von Fehlvorstellungen bei der Ablaufverfolgung von Programmen in einem webbasierten Testsystem
- Anke M. Weber, Veronika Barkela, Sabrina Stiel-Dämmer & Miriam Leuchter: Der Zusammenhang emotionaler Kosten bei Grundschullehramtsstudierenden mit ihrer informatischen Problemlösekompetenz
- Christoph Thyssen, Alexandra Pankow, Kristine Klaeger & Daria Chernyak: Kompetenzen und Nutzungsperspektiven von Lehrkräften zum Einsatz digitaler Medien zur Erkenntnisgewinnung im naturwissenschaftlichen Unterricht



[www.vep-landau.de](http://www.vep-landau.de)

ISSN 0931-5020

ISBN 978-3-944996-73-8



9 783944 996738