

# Digitale Selbstlern-Elemente für Grundlagen der Informatik

David Georg Reichelt

School of Computing and Communications

d.g.reichelt@lancaster.ac.uk

Natanael Arndt

Agile Knowledge Engineering and Semantic Web (AKSW) e.V.

n.arndt@lancaster.ac.uk

## Zusammenfassung

Module zu theoretischen Grundlagen der Informatik haben oft hohe Durchfallquoten und niedrige Grade der Lernzielerreichung. Im internationalen Kontext wird das Lernen zusätzlich dadurch erschwert, dass Studierende an unterschiedliche Lern- und Lehrmethoden gewöhnt sind. Durch Unterstützung der Lehrmethoden mit digitalen Selbstlern-Elementen werden Studierende befähigt, eigenständig Lerninhalte zu wiederholen und zu vertiefen, und dadurch die Lernzielerreichung zu erhöhen. An der Lancaster University ist das Modul Grundlagen der Informatik (Fundamentals of Computer Science) für die Bachelorstudiengänge in Informatik und Softwaretechnik verpflichtend. Die Prüfungsergebnisse zeigen, dass bei der Lernzielerreichung Verbesserungspotenzial besteht. Der vorliegende Beitrag beschreibt den Ausgangsstand des Moduls, der ohne digitale Lehrtechnologien stattfand, sowie zwei Iterationen eines Konzepts zur Verbesserung der Lehre durch regelmäßige, freiwillige Online-Quizze und deren Evaluationen.

## 1 Einleitung

Theoretische Grundlagen der Informatik, insbesondere Grundlagen der Logik und der Mathematik, Algorithmen und Datenstrukturen sowie Grundlagen von Data Science sind für Arbeitsstellen in Informatik und Softwareentwicklung unerlässlich. Durch diese Module wird das informatische Denken (engl. *computational thinking*) geschult, das u. a. mit erhöhter Qualität entwickelter Programme einhergeht [Boom et al., 2022]. Module, die theoretische Grundlagen der Informatik abdecken, sind daher Bestandteil beinahe aller Informatik-, Wirtschaftsinformatik- und Softwaretechnikstudiengänge. Verschiedene Faktoren, wie fehlendes mathematisches Vorwissen und soziale Aspekte, führen zu hohen Durchfallraten und niedriger Lernzielerreichung in vergleichbaren Modulen [Knobelsdorf et al., 2014, Frede und Knobelsdorf, 2018].

An der Lancaster University Leipzig werden Grundlagen der Informatik im Modul „Fundamentals of Computer Science“ (FCS) unterrichtet. FCS ist ein Bachelor-Modul mit 20 Credits, dessen Belegung im ersten Jahr verpflichtend ist. Das Modul wird in allen drei Trimestern unterrichtet und wird durch zwei Zwischenprüfungen am Beginn des 2. und des 3. Trimesters sowie eine Abschlussprüfung am Ende des Jahres geprüft. Im akademischen Jahr 2022/23 haben 13 % das Modul nicht nach der Abschlussprüfung bestanden, und mussten daher eine Wiederholungsprüfung absolvieren. Im akademischen Jahr 2023/24 war dies für 31,2 % der Studierenden der Fall (bei erhöhter Schwierigkeit der Abschlussprüfung).

Die vorliegende Fallstudie beschreibt den Status quo zur Einführung von digitalen Quizzen zur wöchentlichen Überprüfung von Lerninhalten durch die Studierenden. Dabei wird der Ausgangsstatus vor Einführung der digitalen Quizze, die Einführung der Quizze im akademischen Jahr 2023/24 sowie die Restrukturierung im akademischen Jahr 2024/25 dargestellt.

Die Fallstudie ist dabei wie folgt aufgebaut: Eingangs werden bestehende Konzepte zur Lehre in der theoretischen Informatik, sowie deren Konzepte zur Nutzung digitaler Werkzeuge dargestellt. Anschließend wird der Ausgangszustand in FCS dargestellt. Darauf basierend werden die Ergebnisse der Einführung von Online-Quizzen im akademischen Jahr und das Konzept zur Verbesserung der

Quizze im akademischen Jahr 2024/25 beschrieben. Abschließend wird eine Zusammenfassung und ein Ausblick gegeben.

## 2 Konzepte zur Lehre theoretischer Informatik

Junior et al. [Junior et al., 2021] geben einen Überblick über Forschungsarbeiten zur Vermittlung theoretischer Informatik. Sie stellen fest, dass 6 von den 17 untersuchten Arbeiten sich mit der Einführung digitaler Werkzeuge zur Unterstützung der Lehre beschäftigen. Im Folgenden wird ein Überblick über traditionelle Ansätze und den Einsatz digitaler Werkzeuge bei der Vermittlung theoretischer Informatik gegeben.

### 2.1 Traditionelle Ansätze

Knobelsdorf und Frede [Knobelsdorf und Frede, 2016] beschreiben eine Fallstudie in einem Kurs in theoretischen Grundlagen der Informatik, der im zweiten Jahr des Informatikstudiums stattfindet. Durch Beobachtung der Aktivitäten der Studierenden analysieren sie, welche Probleme vorliegen. Eine Beobachtung ist das häufige Ändern der Lernaktivitäten, welches es unmöglich macht, systematisch eine Lösung zu erarbeiten. Darüber hinaus zeigen sich generelle Probleme der Studierenden mit fehlenden Vorkenntnissen in theoretischer Informatik.

Parker und Lewis [Parker und Lewis, 2014] analysieren die Probleme von Studierenden beim Erlernen der Landau-Notation. Durch qualitative Interviews stellen sie fest, dass sowohl die mathematischen Grundlagen (insbesondere Funktionen wie der Logarithmus) als auch deren Anwendung Probleme verursachen. Das reduktive Denken, d. h. das Zurückführen eines Problems auf ein anderes Problem, ist ebenfalls zu wenig ausgeprägt.

Knobelsdorf et al. [Knobelsdorf et al., 2014] beschreiben einen Kurs in theoretischer Informatik, der aus 135 Minuten Vorlesungen, wöchentlichen Hausaufgaben und 90-minütigen Übungen alle zwei Wochen, in denen die Hausaufgaben diskutiert werden, bestand. Aufgrund schwankender und teilweise hoher Durchfallraten wurde die Vorlesung nach dem Modell der *Cognitive Apprenticeship* restrukturiert. Ein Problem herkömmlichen Unterrichts ist, dass der Fokus zu stark auf Wissensvermittlung und zu wenig auf Ausbildung (engl. *apprenticeship*) liegt. Die Wissensvermittlung erfolgt hierbei oft ohne den Kontext. Um diesen zu vermitteln, sollen äquivalent zur Ausbildung in traditionellen Handwerksberufen das benötigte Wissen und die benötigten Fähigkeiten im Kontext demonstriert werden, sodass die Studierenden die Fähigkeiten anschließend unter Anleitung des Lehrpersonals anwenden können. Dabei kommt insbesondere Scaffolding zum Einsatz, bei dem die Studierenden ein Gerüst zur Lösung bestimmter Problemtypen erhalten, das sie anschließend anwenden können. Am Ende einer zweistufigen Restrukturierung des Moduls nach dem Modell der *Cognitive Apprenticeship* wurde daher mehr Wert auf die Demonstration der einzelnen Lösungsverfahren gelegt. Dabei wurden vorbereitende Übungen gestaltet, die ein Grundgerüst für die Lösung typischer Probleme bieten und die zusätzlich eingerichteten Tutorien besprochen wurden. Anschließend wurden detaillierte Lösungsskizzen zur Verfügung gestellt. Durch die Angleichung von vorbereitenden Übungen und den anschließenden regulären Hausaufgaben war es möglich, die Lernzielerreichung zu erhöhen. Nach Änderung des Moduls sanken die Durchfallquoten auf unter 10 %.

### 2.2 Digitale Werkzeuge

Cerone et al. [Cerone et al., 2015] definieren zehn Prinzipien für die Lehre theoretischer Informatik, wobei eines der Prinzipien lautet „Werkzeuge lehren die Methoden - Benutze sie“ (engl. *Tools teach*

*the method - use them*). In diesem Sinn beschäftigt sich eine Vielzahl von Autoren damit, digitale Werkzeuge zur Unterstützung der Lehre theoretischer Grundlagen der Informatik einzusetzen.

Da Silva et al. [da Silva et al., 2023] präsentieren ein Konzept um problemorientiertes Denken durch Spiele in der Schulbildung zu integrieren. Dies geschah im Rahmen der Integration von Informatik-Themen in das brasilianische Schulcurriculum. Dabei wurden Graphgrammatiken in Super-Mario-ähnlichen Spielen implementiert, um Rekursion zu vermitteln.

Knobelsdorf et al. [Knobelsdorf et al., 2017] beschreiben die Nutzung des Theorembeweislers `coq`<sup>1</sup> für das Lernen des Beweisens im Kontext des Bachelorstudiums Informatik (das Konzept wird in [Böhne et al., 2016] erstmals beschrieben). Eine Hürde beim Lernen des Beweisens ist das Überprüfen eines Beweises; sofern die Studierenden Probleme nicht selbst feststellen, können Beweise nur vom Lehrpersonal geprüft werden, was meist mit großer zeitlicher Verzögerung verbunden ist. Der Einsatz von `coq` ermöglicht es, direktes Feedback zum aktuellen Stand zu erhalten. Scaffolding, umgesetzt durch die Bereitstellung von Beweistaktiken, ermöglicht es für die Studierenden, auch ohne Detailkenntnisse von `coq` Beweise durchzuführen. Die Nutzung von `coq` erfolgte in Übungen, die durch eine Vorlesung eingeleitet wurden. Die Analyse der Problemtypen, die in der Fallstudie aufgetreten sind, zeigt, dass der Einsatz von `coq` in der Lehre die Notwendigkeit der Unterstützung durch Lehrpersonal reduziert.

Ähnliche Fallstudien wurden von Avigad [Avigad, 2019], Karsten et al. [Karsten et al., 2023] und Thomas und Iannone [Thoma und Iannone, 2021] durchgeführt. Alle berichten eine Verbesserung der Lernzielerreichung, die jeweils durch Interviews oder Fragebögen erhoben wurde.

Andreas Dengel [Dengel, 2018] beschreibt einen Ansatz zur Vermittlung Endlicher Automaten durch Virtual Reality. Ziel ist es, durch eine immersive Erfahrung die Lernergebnisse zu verbessern. Kern des Ansatzes ist die Abbildung von Endlichen Automaten, die ihrerseits eine reguläre Grammatik darstellen, durch eine Insellandschaft. Spieler starten im Anfangszustand, und können anschließend zu verschiedenen Inseln durch Schiffe gelangen, die mit Symbolen aus dem Eingabealphabet gekennzeichnet sind. Im Fall der Akzeptanz der Eingabe gelangen Spieler am Schluss zu einer Schatzinsel. Durch das Ausprobieren verschiedener Pfade können Spieler prüfen, ob die Eingabe akzeptiert wird. Der Automat wird durch eine Strukturgraphik visualisiert, die der typischen Automaten-darstellung ähnelt. Dengel geht davon aus, dass schwächere Studierende von der VR-Visualisierung profitieren können, während stärkere Studierende traditionelle Aufgaben bevorzugen.

Die Visualisierung von endlichen Automaten wird ebenfalls von Tom Kranz [Kranz, 2018] untersucht. In seiner Arbeit erstellt er eine Webanwendung zur Visualisierung regulärer Grammatiken mithilfe von deterministischen endlichen Automaten.

### 2.3 Erkenntnisse für das betrachtete Modul

Insbesondere die vorgestellten Arbeiten zu traditionellen Ansätzen beschreiben die Abläufe und Probleme der Vermittlung theoretischer Informatik. In der Lehrveranstaltung FCS treten die von [Knobelsdorf und Frede, 2016, Parker und Lewis, 2014] beschriebenen grundlegenden Probleme ebenfalls auf. Um die Vermittlung von Grundlagen theoretischer Informatik effizienter zu gestalten, wird in der vorliegenden Arbeit der von [Knobelsdorf et al., 2014] beschriebene Ansatz des Scaffolding übernommen. Die betrachteten Ansätze zu digitalen Werkzeugen zeigen, dass diese Technologien die Lernzielerreichung im Kontext theoretischer Informatik verbessern können. Die konkreten Werkzeuge sind jedoch eng mit den jeweiligen Lehrinhalten verbunden. Durch diese enge Verbindung sind sie daher nicht geeignet, auf die Vermittlung aller Lehrinhalte der Lehrveranstaltung FCS zu übertragen. Die Erfolge beim Einsatz von Spielen und weiteren Technologien sind ermutigend, eine Übertragung z. B. des Ansatzes für Endlicher Automaten [Dengel, 2018, Kranz, 2018] scheint

---

<sup>1</sup><https://coq.inria.fr/>

nahe liegend, dennoch stellt die Einbindung von Virtual Reality Technologien [Dengel, 2018] eine wesentliche Hürde dar.

### 3 Ausgangsstand

Im Folgenden wird das Modul Fundamentals of Computer Science beschrieben, sein Ablauf dargestellt sowie die Lernzielerreichung vor der Einführung digitaler Quizze diskutiert.

#### 3.1 Moduldetails

Das Modul findet im ersten Bachelorjahr statt und ist Teil der Studiengänge Computer Science und Software Engineering. Da das Modul der Qualitätssicherung des Vereinigten Königreichs unterliegt, ist der dort erstellte Moduldeskriptor verbindlich. Dieser definiert die Lernziele, u.a. „Erklären und Anwenden von Schlüsselkonzepten der diskreten Mathematik wie Mengen, Relationen, Funktionen und Rekursion“ (engl. *Explain and apply key discrete mathematic concepts such as sets, relations, functions and recursion*) und „Analyse und Klassifizierung der Effizienz von Algorithmen und Computerprogrammen“ (engl. *Analyse and classify the efficiency of algorithms and computer programs*), sowie die Aufteilung zwischen sog. *Coursework*, d.h. Aufgaben, die zur Endnote beitragen, und Abschlussprüfungen (30 % vs. 70 %). Schwerpunktsetzungen innerhalb der Lernziele sowie die konkrete Ausgestaltung der *Coursework*, bspw. durch Programmierprojekte, Seminararbeiten oder Zwischenprüfungen, obliegt dem Modulverantwortlichen.

#### 3.2 Ablauf

Der Ablauf der Lehrveranstaltung ist in Abbildung 1 dargestellt und entspricht dem klassischen Ablauf ähnlicher Veranstaltungen [Knobelsdorf et al., 2014]. Die Vorlesung, bestehend aus zweimal aufeinanderfolgend 50 Minuten, beschreibt grundlegende Definitionen, Konzepte und Algorithmen. Nach der Vorlesung werden den Studierenden Aufgaben zur Verfügung gestellt, die bis zur Übung gelöst werden sollen. Die Übung besteht aus Gruppen von maximal 20 Studierenden, in der die Lösungen schwerpunktmäßig von den Studierenden vorgestellt werden. Der Übungsleiter moderiert die Vorstellung der Lösungen, korrigiert diese und stellt Lösungen ggf. selbst vor, wenn niemand der Studierenden in der Lage war, die Aufgaben zu lösen und die Lösung zu präsentieren. Die Teilnahme an der Vorlesung und Übung ist formal verpflichtend, eine Sanktionierung von Nichtteilnahme erfolgt allerdings in der Regel nicht.

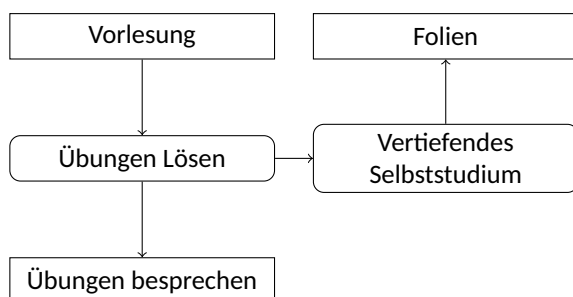


Abbildung 1: Ablauf der Lehrveranstaltung im akademischen Jahr 2022/23

Der zentrale Unterschied zwischen Vorlesung und Übung ist die Gruppengröße: Große Gruppen sind kosteneffizient und angemessen, wenn umfangreiche Sachinformationen präsentiert werden müssen, während kleine Gruppen geeignet sind, damit die Studierenden ihr Denken organisieren und ihr eigenes Lernen überwachen können [Grace und Gravestock, 2008, S. 43, 49]. Kleine Gruppen sind wichtig, um zu erkennen, ob Studierende Schwierigkeiten haben [Grace und Gravestock, 2008, S. 48],

was in den Übungen gelegentlich vorkommt.

Studierende können mit einem Oberflächen- oder Tiefgreifendes-Lernen-Ansatz lernen [Fry et al., 2008, S. 66-67]. Tiefgreifendes Lernen kann durch einen konstruktiv ausgerichteten Lehrplan gefördert werden, bei dem die Studierenden aufgrund der Prüfungsmodalitäten nicht die Möglichkeit haben, oberflächlich zu lernen. Allein die Vorlesungen fördern das oberflächliche Lernen: Durch die Präsentation von Definitionen und einführenden Beispielen könnten die Studierenden den Eindruck gewinnen, dass das Lernen von Definitionen ausreicht. Bspw. würde das Lernen der Definition von aufzählender und beschreibender Notation von Mengen, jeweils mit einem Beispiel  $\{1, 3, 5\}$  und  $\{x|x \in \mathbb{N} \text{ und } 1 \leq x \leq 5 \text{ und } x/2 \notin \mathbb{N}\}$  zu einem oberflächlichen Verstehen der Aufgabenstellung führen.

Daher werden die Vorlesungen durch die Übungen ergänzt. In einer Übungsaufgabe zum selben Thema werden die Studierenden gefragt, wie die Menge in aufzählender Notation, bspw.

$\{-4, 4, -3, 3, -2, 2, -1, 1, 0\}$  in beschreibender Notation dargestellt werden kann. Hierfür ist tiefgreifendes Lernen sowie gewisses mathematisches Vorwissen notwendig: Studierende müssen hier erkennen, dass es sich um die ganzen Zahlen zwischen -4 und 4 handelt, und dies anschließend in die Notation  $\{x|x \in \mathbb{Z} \text{ und } -4 \leq x \leq 4\}$  (o. ä.) umsetzen. Die Lösung von Aufgaben ähnlicher Schwierigkeit gelingt in der Regel einigen Studierenden. Die Mehrheit der Studierenden vermittelt den Eindruck, die Lösung der Aufgabe nach deren Besprechung verstanden zu haben.

Der internationale Kontext bestimmt den Ablauf der Übungen: Während in einigen kulturellen Kontexten, bspw. in Deutschland, das Fragen einzelner Studierender nach Lösungen angemessen ist, wird dies in anderen kulturellen Kontexten, bspw. im asiatischen Raum, als Affront empfunden. Insbesondere wenn der Studierende die Lösung nicht kennt, wird dies als Bloßstellung empfunden. Dadurch wird es schwieriger, das Verstehen der gesamten Kohorte als Übungsleiter zu prüfen und ggf. problematische Themen erneut zu besprechen.

### 3.3 Lernzielerreichung

Die Lernzielerreichung wird vor allem durch die Abschlussklausur kontrolliert, die sich im akademischen Jahr 2022/23 an den Vorjahren orientierte. Die Klausuraufgaben sind dabei leichter als die Übungsaufgaben, bspw. wurde für die Mengenlehre gefragt, was für gegebene Mengen  $A$  und  $B$  die Menge  $(A \cup B) \setminus \mathbb{N}$  ist.

Die Durchfallquote von 13 % ist, im Kontext eines privaten und kostenpflichtigen Studiums, verhältnismäßig hoch. Darüber hinaus gelten Klausuren im ersten Jahr als nicht bestanden, wenn weniger als 40 % der Punkte erreicht wurden. Auch bestandene Klausuren haben daher signifikante Lücken bei der Lernzielerreichung, bspw. hinsichtlich Eigenschaften von Funktionen und Aussagenlogik, aufgewiesen. Dies ist problematisch, da beides Voraussetzung für ein weiteres erfolgreiches Studium ist.

## 4 Fallstudie: Freiwillige Online-Quizze

Zur Verbesserung der Lernzielerreichung wurden im akademischen Jahr 2023/24 drei verschiedene Maßnahmen ergriffen: (1) Die in Kapitel 2.2 dargestellten Werkzeuge sind nicht geeignet, um die breiten Lehrinhalte von FCS abzudecken. Daher wurden die wöchentlichen Übungen durch **Multiple-Choice-Quizze** ergänzt, die den Studierenden die eigenständige Überprüfung des Lernens der Lehrinhalte ermöglichen sollen. (2) In den wöchentlichen Übungen wurden die Multiple-Choice-Fragen stichprobenartig im Plenum diskutiert. (3) Das Niveau der Prüfungsfragen wurde erhöht,

um ein vertieftes Verständnis der zugrundeliegenden Inhalte zu überprüfen. Die Umsetzung dieser Maßnahmen sowie die Evaluation derselben wird im Folgenden diskutiert.

## 4.1 Umsetzung

Die Einführung der Multiple-Choice-Quizze zielte darauf ab, das Verstehen der Vorlesungsinhalte selbstständig durch die Studierenden überprüfen zu lassen. Daher war das Schwierigkeitslevel der Quizze niedriger als das der Übungsaufgaben.

Der um die Quizze erweiterte Ablauf der Lehrveranstaltung ist in Abbildung 2 dargestellt: Jede Woche beginnt mit der Vorlesung (mit der gesamten Kohorte), in der die grundlegenden Inhalte besprochen werden. Anschließend wurde den Studierenden bis Donnerstag Zeit gegeben, die digitalen Quizze zu lösen. Sollten die Studierenden die Quizze nicht selbstständig lösen können, wird Selbststudium anhand der Folien empfohlen. Anschließend sind die Studierenden aufgerufen, bis Freitag die Übungen zu lösen. Auch hier wurde Selbststudium anhand der Folien empfohlen, sofern Probleme beim Lösen der Aufgaben besteht. Anschließend wurden die Aufgaben im Plenum in den Übungen (in Kleingruppen bis zu 20 Studierende) besprochen.

## 4.2 Evaluation

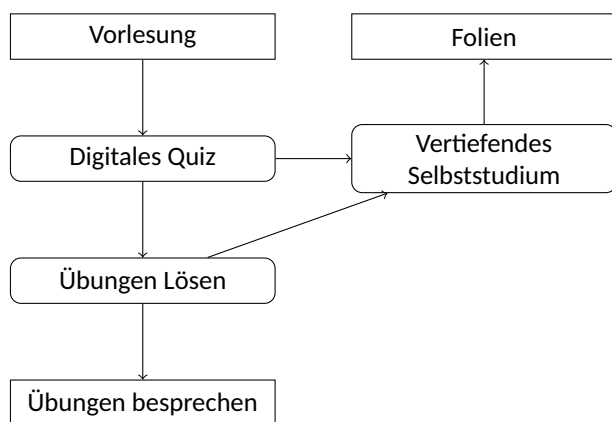


Abbildung 2: Ablauf der Lehrveranstaltung im akademischen Jahr 2023/24

Im textuellen Teil der Modulevaluation wurden die Quizze zweimal erwähnt: Einmal als wertvollster Bestandteil des Moduls, und einmal wurde erwähnt, dass mehr Quizze eine mögliche Verbesserung des Moduls darstellen würden.

Neben diesem qualitativen Feedback durch die Studierenden ist es möglich, die Teilnehmeraten und die Korrektheit der gewählten Ergebnisse zu analysieren. Diese sind in Abbildung 3 visualisiert, wobei Woche 11 ein Zusammenfassungsquiz zur Vorbereitung auf die Zwischenprüfung ist.

Dies zeigt, dass die Teilnehmerate zwischen 45 % und 9 % schwankt,

d. h. es haben zu keinem Zeitpunkt mehr als die Hälfte der Studierenden das Online-Quiz absolviert. Darüber hinaus spiegelt die Tendenz der Teilnehmeraten am Quiz die Teilnehmeraten der physischen Veranstaltungen wider: Während am Semesteranfang die Teilnehmerate hoch ist, sinkt sie bis zum Semesterende, um dann zur finalen Veranstaltung erneut anzusteigen.

Die Korrektheit der Ergebnisse schwankt zwischen 69 % und 35 %. Hierbei ist keine eindeutige Tendenz oder Korrelation mit anderen Einflussfaktoren zu erkennen. Auch vom Anstieg der Durchfallquote von 13 % auf 31,2 % lässt sich, aufgrund erhöhter Schwierigkeit der Abschlussprüfung, kein Rückschluss auf die Wirkung der Quizze ziehen.

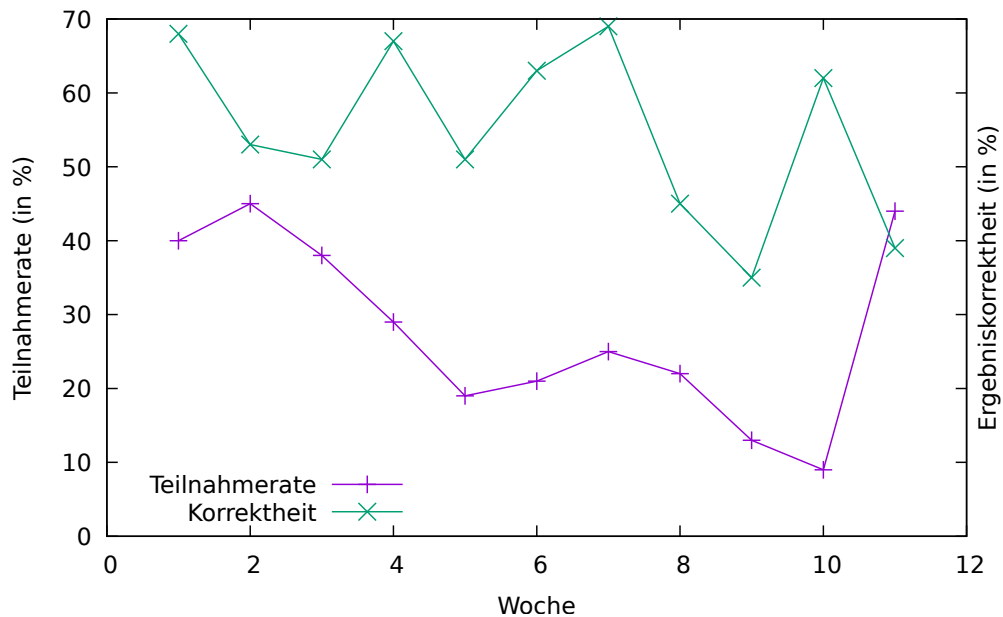


Abbildung 3: Teilnahmerate und Ergebniskorrektheit im akademischen Jahr 2023/24

## 5 Restrukturierung

In diesem Kapitel wird zuerst das Konzept zur Restrukturierung der Quizze und anschließend dessen Evaluation präsentiert.

### 5.1 Konzept

Problem der bisherigen Online-Quizze war, dass sie keine ausreichende Schwierigkeit aufwiesen, um die Übungsaufgaben effektiv vorzubereiten – es war möglich, die Quizze erfolgreich abzuschließen und dennoch die Übungsaufgaben nicht lösen zu können, und es war möglich, die Übungsaufgaben ohne eine Vorbereitung durch die Quizze zu lösen. Ein konstruktiv ausgerichteter Lehrplan sollte dagegen sicherstellen, dass die Lehraktivitäten der Lernzielerreichung und der erfolgreichen Absolvierung der Prüfungen dienen.

Um einen stärker konstruktiv ausgerichteten Lehrplan zu erreichen, planen wir daher, analog zu [Knobelsdorf et al., 2014], Scaffolding einzusetzen. Unser erster Ansatz dazu war, die Online-Quiz-Aufgaben so anzupassen, dass sie eine Vorbereitung für die Übungsaufgaben sind. Dies ist möglich, indem ein Aufgabentyp schrittweise durch ein Gerüst (engl. *scaffold*) bei den Online-Quiz-Aufgaben gelöst wird, der anschließend bei den vorbereitenden Übungen geprüft wird.

Tabelle 1 zeigt beispielhaft für die vorher genannte beschreibende Notation von Mengen, wie diese vorbereitenden Übungen aufgebaut ist: Jeder einzelne Schritt des Lösungsverfahrens wird den Studierenden in Multiple-Choice-Fragen gezeigt, wobei es direkte Rückmeldungen zu falschen und richtigen Lösungen gibt. Sobald die Studierenden in den Online-Quizzen die Lösung selbstständig erarbeitet haben, können sie durch den Lösungsansatz die eigentlichen Aufgabenstellungen selbst lösen.

Somit wird durch die Nutzung eines digitalen Werkzeugs der Lehrplan konstruktiv ausgerichtet.

Vorbereitende Übung	Übungsaufgabe
<p><b>Aufgabe</b> Welche Eigenschaften haben die Set-Elemente von <math>\{2, 4, 6, 8, 10\}</math>?</p> <p>a.) Es handelt sich um natürliche Zahlen. b.) Es handelt sich um ganze Zahlen. c.) Es handelt sich um durch zwei teilbare Zahlen. d.) Es handelt sich um reelle Zahlen.</p> <p><b>Aufgabe</b> Wie kann man formal definieren, dass Zahlen durch zwei teilbar sind?</p> <p>a.) <math>x/3 \in \mathbb{N}</math> b.) <math>x/3 \notin \mathbb{N}</math> c.) <math>x/2 \in \mathbb{N}</math> d.) <math>x/2 \notin \mathbb{N}</math></p> <p><b>Aufgabe</b> Wie kann die aufzählende Notation <math>\{2, 4, 6, 8, 10\}</math> in die beschreibende Notation überführt werden?</p> <p>a.) <math>\{x x \in \mathbb{N} \text{ und } 2 \leq x \leq 10 \text{ und } x/2 \notin \mathbb{N}\}</math> b.) <math>\{x x \in \mathbb{N} \text{ und } 1 \leq x \leq 12 \text{ und } x/2 \notin \mathbb{N}\}</math> c.) <math>\{x x \in \mathbb{Z} \text{ und } 2 \leq x \leq 10 \text{ und } x/2 \notin \mathbb{N}\}</math> d.) <math>\{x x \in \mathbb{Z} \text{ und } 1 \leq x \leq 12 \text{ und } x/2 \notin \mathbb{N}\}</math></p>	<p><b>Aufgabe</b> Wie kann die aufzählende Notation <math>\{-4, 4, -3, 3, -2, 2, -1, 1, 0\}</math> in die beschreibende Notation überführt werden?</p> <p>Selbstständige Lösung durch den Studenten: Erkennen: Es sind Ganzzahlen Es sind Zahlen zwischen -4 und 4</p> <p><b>Formalisieren:</b> Ganzzahlen bedeutet <math>x \in \mathbb{Z}</math></p> <p><b>Formalisieren:</b> <math>-4 \leq x \leq 4</math> <b>Zusammenfassen:</b> <math>\{x x \in \mathbb{Z} \text{ und } -4 \leq x \leq 4\}</math></p>

Tabelle 1: Beispielhafte vorbereitende Übungen

Das Lernen der Studierenden, das durch direktes digitales Feedback unterstützt wird, wird in den Mittelpunkt gestellt. Die Vorlesung und die Übung werden zu Werkzeugen, die die Studierenden beim Lernen unterstützen.

## 5.2 Ergebnisse

Die Ergebnisse der Quizze im Michaelis-Semester 2024/25 sind in Abbildung 4 dargestellt. Zur besseren Durchführbarkeit wurde das letzte Zusammenfassungsquiz in zwei Teile unterteilt. Dabei zeigen sich zwei Änderungen zu 2023/24: (1) Während die Restrukturierung der Quizze zu einer eingangs deutlich erhöhten Teilnehmerate führt, sinkt diese auf ein ähnliches Niveau wie im Vorjahr zum Ende des Semesters. (2) Die Korrektheit der Ergebnisse bleibt auf einem ähnlichen Niveau.

Die Korrelation mit der allgemeinen Präsenz der Studierenden zum Semesterende hin legt nahe, dass die Ursache für die Teilnehmerate nicht an den Quizzen selbst liegt. Da noch keine Daten zu den Ergebnissen der Zwischen- oder Abschlussprüfungen vorliegen, können hier keine Schlüsse gezogen werden.

Dennoch zeigt sich, dass durch gutes Alignment von vorbereitenden Übungen, den Übungen selbst und der anschließenden Zwischenprüfung (die an die Komplexität der Übungsaufgaben an-



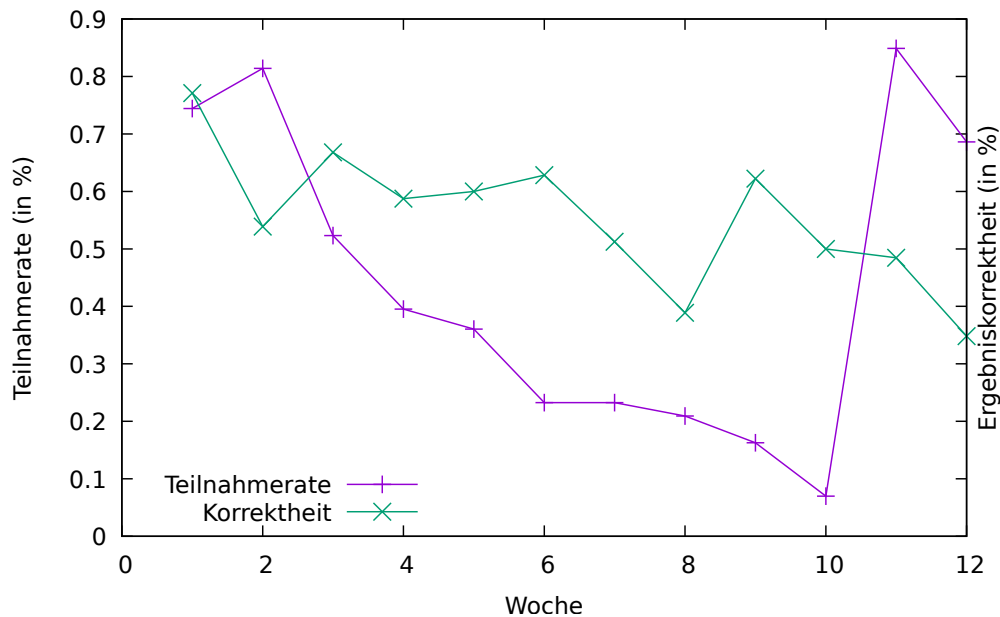


Abbildung 4: Teilnahmerate und Ergebniskorrektheit im akademischen Jahr 2024/25

gelehnt war, und die durch eine online bereitgestellte Probeklausur transparent war) die Aktivität der Studierenden nur wenig erhöht werden kann. Ein potenzieller Grund hierfür ist die enorme Heterogenität der Studierenden: Die Komplexität der Quizze richtet sich eher an die Studierenden, die wenig Vorkenntnisse haben. Mit entsprechenden mathematischen Vorkenntnissen bzw. bei gutem Verständnis der Vorlesung ist es möglich, die Übungen direkt zu lösen.

## 6 Zusammenfassung & Ausblick

In diesem Artikel wurde eine Fallstudie zur Ergänzung klassischer Vorlesungen und Übungen durch Online-Quizze in der theoretischen Informatik beschrieben. Das Konzept war, die Quizze als Vorbereitung der eigentlichen Übungen zu gestalten. Durch die Evaluation zeigte sich, dass die Quizze nur von einer Minderheit der Studierenden konsequent genutzt werden.

Wenngleich die Nutzung der Quizze zeigt, dass diese eine sinnvolle Ergänzung der Lehrmethoden darstellen, bleibt aufgrund bislang nicht vorliegenden Prüfungsergebnisse unklar, ob die Lernzielerreichung durch die Quizze signifikant erhöht wurde. Zur Verbesserung der Veranstaltung sind darüber hinaus weitere Veränderungen nötig. Hierfür können komplexere Aufgabentypen in Übungen und Klausuren das oberflächliche Lernen für Studierende unmöglich machen. Weiterhin können spielbasierte Techniken dazu beitragen, Studierenden den Zugang zur theoretischen Informatik zu erleichtern [da Silva et al., 2023, Dengel, 2018].

## Literatur

- [Avigad, 2019] Avigad, J. (2019). Learning logic and proof with an interactive theorem prover. *Proof technology in mathematics research and teaching*, Seiten 277–290.
- [Böhne et al., 2016] Böhne, S., Kreitz, C., und Knobelsdorf, M. (2016). Mathematisches Argumentieren und Beweisen mit dem Theorembeweiser Coq. *Commentarii informaticae didacticae (CID)*, (10):69–80.
- [Boom et al., 2022] Boom, K.-D., Bower, M., Siemon, J., und Arguel, A. (2022). Relationships between computational thinking and the quality of computer programs. *Education and information technologies*, 27(6):8289–8310.
- [Cerone et al., 2015] Cerone, A., Roggenbach, M., Schlingloff, H., Schneider, G., und Shaikh, S. (2015). Teaching formal methods for software engineering—ten principles. *informatica didactica*, 9.
- [da Silva et al., 2023] da Silva, J. V., da Silva Junior, B. A., Foss, L., und da Costa Cavalheiro, S. A. (2023). Problem-solving Systematization: Introducing Formal Methods in Basic Education. In *Anais do VII Workshop-Escola de Informática Teórica*, Seiten 71–78. SBC.
- [Dengel, 2018] Dengel, A. (2018). Seeking the treasures of theoretical computer science education: Towards educational virtual reality for the visualization of finite state machines. In *2018 IEEE international conference on teaching, assessment, and learning for engineering (TALE)*, Seiten 1107–1112. IEEE.
- [Frede und Knobelsdorf, 2018] Frede, C. und Knobelsdorf, M. (2018). Explorative Datenanalyse der Studierendenperformance in der Theoretischen Informatik. *HDI*, 18:135–149.
- [Fry et al., 2008] Fry, H., Ketteridge, S., und Marshall, S. (2008). *A handbook for teaching and learning in higher education: Enhancing academic practice*. Routledge.
- [Grace und Gravestock, 2008] Grace, S. und Gravestock, P. (2008). *Inclusion and diversity: Meeting the needs of all students*. Routledge.
- [Junior et al., 2021] Junior, B. A., Cavalheiro, S. A., und Foss, L. (2021). Theoretical computer science in basic education: A systematic review. In *Anais do VI Workshop-Escola de Informática Teórica*, Seiten 133–140. SBC.
- [Karsten et al., 2023] Karsten, N., Jacobsen, F. K., Eiken, K. J., Nestmann, U., und Villadsen, J. (2023). ProofBuddy: A Proof Assistant for Learning and Monitoring. *arXiv preprint arXiv:2308.06970*.
- [Knobelsdorf und Frede, 2016] Knobelsdorf, M. und Frede, C. (2016). Analyzing student practices in theory of computation in light of distributed cognition theory. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, Seiten 73–81.
- [Knobelsdorf et al., 2017] Knobelsdorf, M., Frede, C., Böhne, S., und Kreitz, C. (2017). Theorem provers as a learning tool in theory of computation. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, Seiten 83–92.
- [Knobelsdorf et al., 2014] Knobelsdorf, M., Kreitz, C., und Böhne, S. (2014). Teaching theoretical computer science using a cognitive apprenticeship approach. In *Proceedings of the 45th ACM technical symposium on Computer science education*, Seiten 67–72.

[Kranz, 2018] Kranz, T. (2018). E-Learning-Unterstützung für Theoretische Informatik.

[Parker und Lewis, 2014] Parker, M. und Lewis, C. (2014). What makes big-O analysis difficult: understanding how students understand runtime analysis. *Journal of Computing Sciences in Colleges*, 29(4):164–174.

[Thoma und Iannone, 2021] Thoma, A. und Iannone, P. (2021). Learning about proof with the theorem prover lean: the abundant numbers task. *International Journal of Research in Undergraduate Mathematics Education*, Seiten 1–30.