Support for Programmed Instruction in an eTextbook

Mostafa Mohammed
Virginia Tech
Blacksburg, Virginia
profmdn@vt.edu

Piexuan Ge
Virginia Tech
Blacksburg, Virginia
gpeixuan@vt.edu

Samnyeong Heo
Virginia Tech
Blacksburg, Virginia
hsn1017@vt.edu

Clifford A. Shaffer
Virginia Tech
Blacksburg, Virginia
shaffer@vt.edu

ABSTRACT

Students often skip through instructional material without paying enough attention to the content, resulting in less understanding. Inspired by the Programmed Instruction (PI) technique, we implemented extensions to an eTextbook system to support instructional slideshows with a large number of interspersed questions. Students must answer a question correctly to pass to the next slide. This completely changes how students interact with the material. Our initial results show a significant increase in students' grades when they used PI for a Formal Languages course.

CCS CONCEPTS

• **Social and professional topics** → *Student assessment*; • **Applied computing** → **Interactive learning environments**; • **Software and its engineering** → *Simulator / interpreter*.

KEYWORDS

Programmed Instruction, Auto-grading, student engagement

ACM Reference Format:
Mostafa Mohammed, Piexuan Ge, Samnyeong Heo, and Clifford A. Shaffer. 2021. Support for Programmed Instruction in an eTextbook. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21), March 13–20, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3408877.3439586

1 BACKGROUND

Students often skim or skip material in a textbook without really engaging. OpenDSA [2] lets instructors create eTextbooks that focus on visualizations and exercises. The slideshow is a common element in OpenDSA, they allow students to see content broken into small amounts of text, each slide associated with a visual element. Our previous research shows that students are more likely to engage nearly identical material presented as a slideshow rather than as prose. But some students still skip over slides without reading, or do not engage enough to understand the slide material.

SIGCSE '21, March 13–20, 2021, Virtual Event, USA

ACM ISBN 978-1-4503-8062-1/21/03.
https://doi.org/10.1145/3408877.3439586

PI teaching techniques were once in actively researched, with good understanding related to instruction effectiveness, learner pacing, reinforcement strategies, and long-term effects [3]. Modern web-based interactive systems make PI easily implementable. To increase engagement, we adopted PI methods to enhance slideshow presentations. A PI-enhanced slideshow (called a frameset) interspersed simple questions with nearly every frame. Once the question is answered correctly, students can advance to the next frame.

2 METHODOLOGY

Our implementation allows developers to easily integrate questions with visual content. Frameset question types include T/F, multiple choice, select all correct answers on the list, fill in the blank, and embedded proficiency exercise (in general, any existing interactive exercise available within OpenDSA). A question object consists of several parts speicified in a JSON file: question type, description (this frame's instructional content), question (this frame's question), correct answer, and (depending on the question type) a list of choices. Instructors can specify text to be given in response to the correct answer, and other text with the incorrect answer. These hints allow students stuck in a question to advance. Our hypothesis is that it is good to keep most questions simple. However, a fraction of the questions require real thought, because they might have half a dozen or more choices, and several must be selected. These tend to be "proficiency" questions, which require students demonstrate their working knowledge.

We hypothesize that not all topics are equally valuable for presentation using PI techniques. Topics like data structures and algorithms have proven to be well presented using visualizations and interactive exercises that require students to demonstrate understanding of the algorithm in question by reproducing its behavior. But to understand more mathematical topics requires that students stay engaged and focused on the details of the presentation [1]. Thus, we focus on augmenting a Formal Languages and Automata course, as this is one of the most mathematical topics in CS.

REFERENCES

[1] M. F. Farghally, K. H. Koh, H. Shahin, and C. A. Shaffer. Evaluating the effectiveness of algorithm analysis visualizations. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pages 201–206, 2017.
[2] E. Fouh, V. Karavirta, D. A. Breakiron, S. Hamouda, S. Hall, T. L. Naps, and C. A. Shaffer. Design and Architecture of an Interactive ETextbook–The OpenDSA System. *Science of Computer Programming*, 88:22–40, 2014.
[3] M. Molenda. When Effectiveness Mattered. *TechTrends*, 52(2):53, 2008.