Exploring the Impact of Multiple Representations in Introductory Programming
A Pilot Study

Sibia, Naaz; Osorio, Valeria Ramirez; Bernuy, Angela Zavaleta; Aivaloglou, Efthimia; Engineer, Rutwa; Petersen, Andrew; Liut, Michael; Nobre, Carolina

DOI
10.1145/3699538.3699587

Publication date
2024

Document Version
Final published version

Published in
Proceedings of 24th International Conference on Computing Education Research, Koli Calling 2024

Exploring the Impact of Multiple Representations in Introductory Programming: A Pilot Study

Naaz Sibia
Department of Computer Science,
University of Toronto
Canada
naaz.sibia@utoronto.ca

Valeria Ramirez Osorio
Department of Mathematical and
Computational Sciences, University of
Toronto Mississauga
Canada
valeria.ramirezosorio@mail.utoronto.ca

Angela Zavaleta Bernuy
Department of Computer Science,
University of Toronto
Canada
angelazb@cs.toronto.edu

Efthimia Aivaloglou
Technical University of Delft
Netherlands
E.Aivaloglou@tudelft.nl

Rutwa Engineer
Mathematical and Computational
Sciences, University of Toronto
Mississauga
Canada
rutwa.engineer@utoronto.ca

Andrew Petersen
Mathematical and Computational
Sciences, University of Toronto
Mississauga
Canada
andrew.petersen@utoronto.ca

Michael Liut
Mathematical and Computational
Sciences, University of Toronto
Mississauga
Canada
michael.liut@utoronto.ca

Carolina Nobre
Department of Computer Science,
University of Toronto
Canada
carolina.nobre@utoronto.ca

Abstract

This pilot study explores how visualization strategies, grounded in multiple representations theory, impact novice students' engagement, and cognitive load during program tracing tasks. Students were were shown a visualization of the three-variable swap problem at the start of an introductory programming course (CS1) at a large public North American research-intensive university. We compared three conditions: interactive multiple representations, Python Tutor (a single-representation tool), and text-only methods. Preliminary results indicate that interactive multiple representations increase engagement for students with prior programming experience, while no significant differences were observed for students without prior experience. These findings suggest that while multiple representations may boost engagement, identifying how to effectively support students of all experience levels and reduce cognitive load requires further study.

CCS Concepts

• **Social and professional topics → Computing education**.

Keywords

visualization, multiple representations theory, code tracing, introductory programming

ACM Reference Format:
Naaz Sibia, Valeria Ramirez Osorio, Angela Zavaleta Bernuy, Efthimia Aivaloglou, Rutwa Engineer, Andrew Petersen, Michael Liut, and Carolina Nobre. 2024. Exploring the Impact of Multiple Representations in Introductory Programming: A Pilot Study. In *24th Koli Calling International Conference on Computing Education Research (Koli Calling '24), November 12–17, 2024, Koli, Finland.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3699538.3699587

1 Introduction

Understanding and tracing code is a critical challenge for students without prior experience [5, 7]. Many struggle with ineffective strategies, such as single-value tracing [7], which can hinder their ability to develop fundamental problem-solving skills for advanced programming concepts.

While visualization and sketching have been proposed as solutions, results have been mixed [4, 8]. Interactive visualizations that engage students with multiple representations of a program (e.g., visual memory models or diagrams) may be more effective for novices [1, 8]. Multiple representations theory suggests combining abstract and concrete representations enhances engagement and learning outcomes [3]. In contrast, tools like Python Tutor, which provide only a single, abstract representation, may lack the depth of interaction and scaffolding that multiple representation approaches offer.

Koli Calling '24, November 12–17, 2024, Koli, Finland

ACM ISBN 979-8-4007-1038-4/24/11
https://doi.org/10.1145/3699538.3699587

This pilot study, conducted in an introductory programming course, involved 829 students randomly assigned to three conditions: (1) interactive multiple representations, (2) Python Tutor, and (3) text-only tracing. A survey collected early in the course assessed students' initial experiences with these tools.

We evaluated engagement and cognitive load using an adapted NASA-TLX scale [2] and a modified User Engagement Scale (UES) [6]. Cognitive load items included questions like "How mentally demanding was the task?" (Cronbach's $\alpha = 0.95$). Engagement questions assessed enjoyment and usability ($\alpha = 0.98$).

2 Results

An ANOVA revealed a significant difference in engagement across conditions: $F(2, 829) = 9.07$, $\boldsymbol{p} < \mathbf{.001}$, with higher engagement reported for the interactive multiple representations tool compared to Python Tutor (Mean difference = 0.455, $\boldsymbol{p} < \mathbf{.001}$). No significant difference was found between interactive and text-only conditions (Mean difference = 0.15, $p = 0.31$), and cognitive load did not vary significantly across conditions: $F(2, 829) = 0.57$, $p = 0.57$.

For students without prior experience, there were no significant differences in engagement or cognitive load across the conditions. However, for students with prior experience, engagement differed significantly: $F(2, 572) = 8.12$, $\boldsymbol{p} < \mathbf{.001}$, with higher engagement for the multiple representations tool compared to Python Tutor (Mean difference = 0.52, $\boldsymbol{p} < \mathbf{.001}$) and for text-only compared to Python Tutor (Mean difference = 0.38, $\boldsymbol{p} = \mathbf{0.03}$). Cognitive load remained consistent across all groups.

These findings suggest that prior experience affects how students engage with visualization tools. For beginners, the tools didn't significantly impact engagement or cognitive load, while for experienced students, interactive and text-only approaches showed higher engagement than Python Tutor. However, cognitive load remained similar across all groups.

3 Implications

Our results suggest that multiple representations boost engagement for more experienced students without increasing cognitive load, indicating their potential for future educational tools. For beginners, early exposure to these tools may not yield immediate benefits, but effectiveness might increase as they progress as suggested by prior work [1, 8].

This pilot study is part of a larger project aimed at tracking student engagement and cognitive load across an entire semester. We plan to investigate how these metrics shift as students encounter progressively more complex problems and gain experience using the visualizations. We hypothesize that, as students develop stronger problem-solving skills and conceptual understanding, the benefits of multiple representations will become more pronounced. We also plan to investigate whether the differences observed were influenced by the novelty of the tools and whether engagement would increase if single-representation tools were introduced later.

Acknowledgments

This project was supported by the Institute for Pandemics (IfP) Graduate Studentship Award, University of Toronto and the National Sciences and Engineering Research Council of Canada's (NSERC) Discovery grant #RGPIN-2024-04348.

References

[1] Tuukka Ahoniemi and Essi Lahtinen. 2007. Visualizations in preparing for programming exercise sessions. *Electronic Notes in Theoretical Computer Science* 178 (2007), 137–144.

[2] Sandra G Hart. 1986. NASA task load index (TLX). (1986).

[3] Jeisson Hidalgo-Céspedes, Gabriela Marín-Raventós, and Vladimir Lara-Villagrán. 2016. Learning principles in program visualizations: A systematic literature review. In *2016 IEEE frontiers in education conference (FIE)*. IEEE, 1–9.

[4] Christopher D Hundhausen, Sarah A Douglas, and John T Stasko. 2002. A meta-study of algorithm visualization effectiveness. *Journal of Visual Languages & Computing* 13, 3 (2002), 259–290.

[5] Laurie Murphy, Sue Fitzgerald, Raymond Lister, and Renée McCauley. 2012. Ability to'explain in plain english'linked to proficiency in computer-based programming. In *Proceedings of the ninth annual international conference on International computing education research*. 111–118.

[6] Heather L O'Brien and Elaine G Toms. 2010. Is there a universal instrument for measuring interactive information retrieval? The case of the user engagement scale. In *Proceedings of the third symposium on Information interaction in context*. 335–340.

[7] Jorma Sajaniemi, Marja Kuittinen, and Taina Tikansalo. 2008. A study of the development of students' visualizations of program state during an elementary object-oriented programming course. *Journal on Educational Resources in Computing (JERIC)* 7, 4 (2008), 1–31.

[8] Juha Sorva et al. 2012. *Visual program simulation in introductory programming education*. Aalto University.