

Analyzing students' recontextualization strategies for algorithmic concepts

Jacqueline Nijenhuis-Voogt
Radboud University
Nijmegen, The Netherlands
j.nijenhuis@science.ru.nl

Paulien C. Meijer
Radboud University
Nijmegen, The Netherlands

Durdane Bayram-Jacobs
Radboud University
Nijmegen, The Netherlands

Erik Barendsen
Radboud University & Open University
Nijmegen, The Netherlands

ABSTRACT

Learning to apply fundamental algorithmic concepts in a variety of contexts is a challenge in secondary computer science education. The aim of this ongoing study is to investigate what recontextualization strategies students use and specifically how students recognize what algorithmic concepts they can use in given situations. A card sorting task was developed in which students match algorithmic problems into given categories. Preliminary findings indicate that students experience difficulties with recognizing the underlying algorithms and use mainly surface features to sort the algorithmic problems.

CCS CONCEPTS

• **Social and professional topics** → **Computer science education; K-12 education.**

KEYWORDS

algorithms, context, recontextualization, secondary education, card sorting

ACM Reference Format:

Jacqueline Nijenhuis-Voogt, Durdane Bayram-Jacobs, Paulien C. Meijer, and Erik Barendsen. 2019. Analyzing students' recontextualization strategies for algorithmic concepts. In *14th Workshop in Primary and Secondary Computing Education (WiPSCE'19), October 23–25, 2019, Glasgow, Scotland Uk*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3361721.3361734>

1 INTRODUCTION

It is an essential learning goal in education that students learn to recontextualize, defined as the ability to apply knowledge outside the context in which it has been introduced. For secondary computer science education, learning to recontextualize is an important factor in improving students' algorithmic thinking as, according to CS curricula in several countries, e.g., [3], students are supposed to be able to adapt and use several key algorithms. Hence, students are not only expected to learn these standard algorithms, but, in addition, they are expected to recognize what algorithms are appropriate to use in a given situation. Therefore, students need to learn strategies and skills to transfer their knowledge of standard algorithms to a new context.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

WiPSCE'19, October 23–25, 2019, Glasgow, Scotland Uk

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-7704-1/19/10...\$15.00

<https://doi.org/10.1145/3361721.3361734>

It is interesting to examine how students develop the ability to use algorithms in a given context. As a first step, we investigated how students recognize what algorithm to use in a certain context, what elements in a context support determining what algorithm is appropriate to use, and what similarities between contexts play a role in this process. Although research has been carried out on transfer and recontextualization [8, 14], to our knowledge, no studies have been found which focus on recontextualization of algorithmic thinking. This work in progress explores students' strategies for recognizing abstract concepts in application situations to be able to apply an algorithm in a given situation.

2 RECONTEXTUALIZATION

To enable students to apply their knowledge to given problems and settings, well-chosen contrasting cases as alternative contexts may be offered [2]. Guzdial [5] argued also for the use of multiple contexts, as he stated that "the only way to achieve decontextualized knowledge is to teach beyond a single context" (p. 6). Likewise, existing research in biology education [14] indicated that using a concept in different contexts facilitates transfer.

Previous research has shown the complexity of recontextualization or transfer of programming skills and knowledge to other domains. Papert claimed that when children learn to program computers, their problem-solving abilities in non-programming domains will increase as well [10]. This claim has been examined [6, 7] with inconclusive results. These variations in findings might be explained by the difference in 'high road' and 'low road' transfer [11], as suggested by Voogt et al. [13].

Students' recontextualization strategies may be investigated by using a card sorting task to trigger their thinking. The aim of card sorting tasks is not to assess the presence or absence of certain knowledge, but to characterize conceptual knowledge [12]. Smith et al., [12] found that experts sorted the cards based on hypothesized deep features where novices appeared to sort based on surface features.

3 AIM OF THE STUDY

The aim of this study is to investigate students' recontextualization strategies and to examine how students recognize what algorithmic concepts they can use in a given context. This aim raises the following research questions: (1) How do students match algorithmic concepts to contexts? and (2) What difficulties do students experience during recontextualization? To examine how students match algorithmic concepts to contexts, we investigated their reasoning. Furthermore, we examined which characteristics of the new situation play a key role in their reasoning.

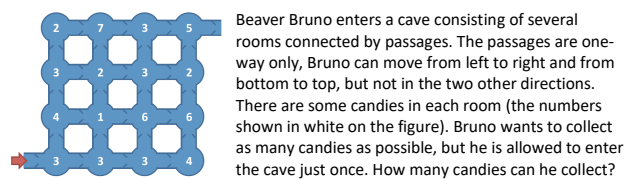


Figure 1: Collecting candies - An example of a Bebras task

4 METHOD

In this study, the data is collected during a card sorting task, as in the study of Smith et al. [12]. Twelve students from 10th grade, 15 or 16 years old, participated in this study, after some introductory lessons on algorithms. We selected a set of algorithmic problems derived from Bebras tasks [4] used in an earlier study regarding concepts in K-9 CS education [1], see Figure 1 for an example. Problems differ in distance of transfer, some are very similar to the context in which a concept was learned ('low road') while others are more distant ('high road').

First, students were individually offered these problems and were asked to sort them into groups using algorithm as criterion (more than one group and fewer groups than the number of cards, cards can only be added to one group). We used a framed task condition protocol which means that the different categories were given to students. We gave the students four categories: sorting algorithms, searching algorithms, shortest path algorithms, unfeasible (i.e., no efficient algorithm known). We asked students to come up with a new category when they thought a card did not fit in any of the given categories. We asked the participants not to solve the algorithmic problems. The participating students were given as much time as needed to sort the cards. Furthermore, we tried to create a 'safe' environment by emphasizing that the task was not a test and that results would not get rated.

When students had finished their individual sorting, they were given a form with two questions to answer: 'For which problems did you see immediately to which category it belongs? Why?' And the other question was 'For which problem was it difficult to decide to which group to add the problem? Why?' The next step was a discussion in a focus group of three students. We asked the students to come up with a final grouping of the cards. While discussing in what category an algorithmic problem could be placed, they explained to each other where they had originally placed the problem. We encouraged the students to talk about the underlying reasons for placing a problem in a certain category.

A qualitative data analysis approach [9] was used to analyze the written answers to the evaluation questions and the transcripts of the focus group discussion.

5 FIRST FINDINGS

Preliminary findings suggest that it is not obvious for students to recognize underlying algorithms when given a set of algorithmic problems. Regarding the example task, given in Figure 1, students struggle to see that a shortest path algorithm may be used. Students appear to get confused because this problem does not ask for a path, but the problem is about collecting candy. The problem does not ask for a minimum (shortest distance) but for a maximum (collect as many candies as possible), which adds to the confusion. A student commented "you have to collect as many candies but the shortest route does not always end up with the most candies". And another student pondered "you have to collect the most candies and therefore you need a route, you have

to find the correct route that yields the most candies, and you can pretend that that is the length of the route and this way you can calculate the longest route". In addition, our first findings indicate that students may sort based on surface features. One student commented: "If the text contained the word 'route', the problem belonged to 'shortest path'", apparently missing the fact that some problems that ask for a route belong to the category of unfeasible problems.

6 DISCUSSION AND FUTURE WORK

The purpose of the present study was to investigate students' strategies when they are asked to apply their knowledge of algorithms in new contexts. Preliminary findings indicate that students make use of surface features to match algorithmic concepts to new contexts. Both the wording and the graphics of the algorithmic problems seem to play an important role in students' strategies to recontextualize algorithmic concepts. These findings contribute to our understanding of students' recontextualization strategies and insights gained from this study may be of assistance to teachers and curriculum developers. Despite these results, questions remain regarding the process of recontextualization. Further research is required to investigate how students develop the ability to recontextualize.

ACKNOWLEDGMENTS

This research received funding from the Dutch Ministry of Education, Culture and Science under the Dudoc programme.

REFERENCES

- [1] Erik Barendsen, Linda Mannila, Barbara Demo, Natasa Grgurina, Cruz Izu, Claudio Mirole, Sue Sentance, Amber Settle, and Gabriele Stupuriene. 2015. Concepts in K - 9 Computer Science Education. In *Proceedings of the 2015 ITICSE on Working Group Reports*. ACM, New York, NY, 85–116.
- [2] John D. Bransford, Ann L. Brown, and Rodney R. Cocking. 2000. *How People Learn (Expanded ed.)*. National Academy, Washington, DC.
- [3] Computing at School Working Group. 2012. *Computer Science: A curriculum for schools*.
- [4] Valentina Dagiene and Gabriele Stupuriene. 2016. Bebras - a sustainable community building model for the concept based learning of informatics and computational thinking. *Informatics in Education* 15, 1 (2016), 25–44.
- [5] Mark Guzdial. 2010. Does contextualized computing education help? *ACM Inroads* 1, 4 (2010), 4–6.
- [6] David Klahr and Sharon Mc Coy Carver. 1988. Cognitive objectives in a LOGO debugging curriculum: Instruction, learning, and transfer. *Cognitive Psychology* 20, 3 (1988), 362–404.
- [7] D. Midian Kurland, Roy D. Pea, Catherine Clement, and Ronald Mawby. 1986. A study of the development of programming ability and thinking skills in high school students. *Journal of Educational Computing Research* 2, 4 (1986), 429–458.
- [8] Joanne Lobato. 2006. Alternative perspectives on the transfer of learning: History, issues, and challenges for future research. *The Journal of the Learning Sciences* 15, 4 (2006), 431–449.
- [9] Matthew B. Miles, A. Michael Huberman, and Johnny Saldana. 2014. *Qualitative data analysis: A method sourcebook*. Sage, Thousand Oaks, CA.
- [10] Seymour Papert. 1980. *Mindstorms: children, computers, and powerful ideas*. Basic Books, New York, NY.
- [11] Gavriel Salomon and David N. Perkins. 1989. Rocky roads to transfer: Re-thinking mechanism of a neglected phenomenon. *Educational Psychologist* 24, 2 (1989), 113–142.
- [12] Julia I. Smith, Elijah D. Combs, Paul H. Nagami, Valerie M. Alto, Henry G. Goh, Muryam A. A. Gourdet, Christina M. Hough, Ashley E. Nickell, Adrian G. Peer, John D. Coley, and Others. 2013. Development of the biology card sorting task to measure conceptual expertise in biology. *CBE-Life Sciences Education* 12, 4 (2013), 628–644.
- [13] Joke Voogt, Petra Fisser, Jon Good, Punya Mishra, and Aman Yadav. 2015. Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies* 20, 4 (2015), 715–728.
- [14] Menno Wierdsma, Marie-Christine Knippels, Bert van Oers, and Kerst Boersma. 2016. Recontextualising cellular respiration in upper secondary biology education. Characteristics and practicability of a learning and teaching strategy. *Journal of Biological Education* 50, 3 (2016), 239–250.