



HAL
open science

Upper-secondary school students' mistakes in using loops while coding cubic solids in a block-based coding environment

Martin Cápáy, Eva Schmidthaler, Janka Medová

► To cite this version:

Martin Cápáy, Eva Schmidthaler, Janka Medová. Upper-secondary school students' mistakes in using loops while coding cubic solids in a block-based coding environment. Paul Drijvers; Csaba Csapodi; Hanna Palmér; Katalin Gosztanyi; Eszter Kónya. Thirteenth Congress of the European Society for Research in Mathematics Education (CERME13), Jul 2023, Budapest, Hungary. Alfréd Rényi Institute of Mathematics; ERME, Proceedings of the Thirteenth Congress of the European Society for Research in Mathematics Education (CERME13), TWG16 (7). <hal-04406832>

HAL Id: hal-04406832

<https://hal.science/hal-04406832v1>

Submitted on 19 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Upper-secondary school students' mistakes in using loops while coding cubic solids in a block-based coding environment

Martin Cápaj¹, Eva Schmidthaler² and Janka Medová¹

¹Constantine the Philosopher University in Nitra, Slovakia; jmedova@ukf.sk

²Johannes Kepler University, Linz School of Education, Linz, Austria

Keywords: Computation, algorithms, geometry.

Loops in algorithmic thinking

In recent years there have been several attempts to design problems where mathematical abilities are intertwined with the development of algorithmic thinking (Weintrop et al., 2016). As Weber (2022) states, “loops are a constituent element of algorithms, thinking in loops can also be considered as a constituent element of algorithmic thinking” (Weber, 2022, p.3). There are two basic types of loops, count-controlled loops usually modelled by “for” and event-controlled loops. The research in loops education revealed several misconceptions related to count-controlled loops: students expect the exact same output in every iteration; repeating each action separately instead of an entire sequence (Grover and Basu, 2017); incorrectly added code before or after loop; and not removing excess repetitions inside one loop (Vaniček et al., 2023). The task of creating planar patterns is very typical for loop education. Furthermore, augmented reality (AR) allows the building of spatial objects, and therefore, brings new challenges for students. In the project <colette/> (Computational Thinking Learning Environment for Teachers in Europe) there was a web-based learning environment designed, using block-based coding language, based on Blockly (Google and MIT). The AR feature of the <colette/> app offers the possibility to design tasks in a spatial coordinate system (Milicic et al., 2021) which may cause new types of students' misconceptions, not necessarily related to the concept of loops. In this contribution, we will try to identify students' errors when creating the patterns from cubes in <colette/> environment and describe the sources of these mistakes.

Methods

The 24 students of three classes (15 girls, 9 boys), grades 11 to 13, from a grammar school in Slovakia solved the sequence of tasks focused on building cubic solids with the <colette/> type format Building Cubes. All the participating students had prior experience in programming (e.g., utilization of loops with Python) in different levels. Students should do the abstraction of the shape and create the object from little cubes. We focus on the task when the coniferous tree is to be modelled. Three nested loops are needed. The first two construct the square with a given dimension and the third one controls the size of the square and the height of the layer. The Blockly-based code editor was used. The session was led by the first and tasks were developed by the second author. The session was audiotaped, the students' codes were saved, and pictures of students-created cubic solids were taken. Then the students' work was analyzed, and students' mistakes were identified.

Findings

The occurred misconceptions can be divided to two basic categories: (1) misconceptions related to loops, and (2) misconceptions related to special geometry. The first category comprised two types of

students' mistakes: The source of the first one is the difference between Python that students usually use and the Blockly-based learning environment of <colette/> in the borders of the count-controlled loops. Several students found out that their programmed shape differs in dimensions from what they expected. They were suggested to find the difference, make a prediction about why it is like that, and come up with a solution. The second revealed misconception was identified by one student who used a variable for drawing the layers of the pyramid representing the coniferous tree but did not use the loop, instead, the participant placed each "cube" (block) for each level. It is similar to the findings by Grover and Basu (2017) when students repeated each action separately instead of an entire sequence. Regarding the geometry misconceptions, there occurred three various mistakes: One student did not build the full solid, only the surface layer. The second occurred for several students. They produced only a one-layer high object. They did not change the coordinate for height in spite of using three nested loops with the variable controlling the dimension of the layers of the pyramid. They needed to be prompted that, unlike planar objects, shapes in space have three coordinates and they need to change all of them to build a spatial object.

Conclusions

There were two mistake-types identified, one related to the shift between Python to another programming language in the <colette/> environment, and the second based on students' abstraction of modelled objects, not using the loops at all, and errors related to students' ability to work with coordinates (spatial skills). The activity revealed these sources of mistakes in students-produced code.

Acknowledgment

The publication is funded by Slovak Research and Development Agency, grant no. APVV-20-0599.

References

- Grover, S., & Basu, S. (2017). Measuring Student Learning in Introductory Block-Based Programming: Examining Misconceptions of Loops, Variables, and Boolean Logic. In M.E. Caspersen et al. (Eds.) *SIGCSE '17: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. (262–272). ACM. <https://doi.org/10.1145/3017680.3017723>
- Milicic, G., van Borkulo, S., Medova, J., Wetzel, S., & Ludwig, M. (2021). Design and development of a learning environment for computational thinking: The Erasmus+ project. In L. Gómez Chova et al. (Eds.) *EDULEARN21 Proceedings* (pp.7376–7383). IATED. [10.21125/edulearn.2021.1495](https://doi.org/10.21125/edulearn.2021.1495)
- Vaniček, J., Dobiáš, V., Šimandl, V. (2023). Understanding loops: What are the misconceptions of lower-secondary pupils? *Informatics in Education*. 22(3). 525–554. [10.15388/infedu.2023.20](https://doi.org/10.15388/infedu.2023.20)
- Weber, C. (2022). Advantages and disadvantages of using loops in algorithms: conceptions of pre-service primary teachers learning Scratch. In Hodgen, J., Geraniou, E., Bolondi, G. & Ferretti, F. (Eds.) (2022). *Proceedings of the Twelfth Congress of the European Society for Research in Mathematics Education (CERME12)*. Free University of Bozen-Bolzano and ERME.
- Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, 25(1), 127–147. <https://doi.org/10.1007/s10956-015-9581-5>