Studying Computational Thinking Practices Through Collaborative Design Activities with Scratch

Joey Huang, huang220@iu.edu, Indiana University
Kylie Peppler, kpeppler@uci.edu, University of California, Irvine

Abstract: Previous studies have focused on examining individuals' computational thinking (CT) practices in varied learning contexts. This study aimed to expand the current framework of CT by investigating how CT is practiced through collaborative design activities with Scratch. We analyzed students' CT practices as a group in different design stages. By identifying the patterns of CT practices which emerged through collaborative design activities, this study informed how CT is socially practiced in small groups.

Introduction

The notion of computational thinking (CT) is becoming increasingly important for all citizens in the digital age. Wing (2006) argues that computational thinking will be a fundamental skill by the middle of the 21st century, just like reading, writing, and arithmetic. CT entails a series of problem-solving processes, such as recognizing patterns, and systematically breaking down a problem, and then composing an algorithmic solution. Recent studies have begun to develop operational definitions or frameworks in order make the concept of CT accessible to educators (e.g., Grover et al. 2014). Brennan and Resnick (2012) propose a new framework for studying computational thinking from three dimensions to provide a comprehensive understanding of CT. Particularly, CT practices involve 4 aspects: (1) experimenting and iterating, (2) testing and debugging, (3) reusing and remixing, and (4) abstracting and modularizing. They argue that the development of computational thinking is an interactive and reflective processes in which learners create, revise, and share their creations during learning process. Learners develop their CT through the processes of making and cultivate their knowledge based on their experiences.

Collaborative design activities are defined to be a knowledge creation process which involves students actively communicating and working together to create a shared view of joint design ideas and decisions (Hennessy & Murphy, 1999). Learning through collaborative design process has been proven to deepen students' content knowledge through practices and advance their problem-solving skills to solve complex and multifaceted problems (Hakkarainen et al., 2013). Many studies have focused on studying individuals' CT practices and development in varied learning contexts (e.g., Weintrop et al., 2016). However, little attention has been paid to computational thinking practice in a collaborative learning environment, focusing on how CT is socially situated and practiced through interaction (e.g., Chowdhury, 2015). In this study, we applied Brennan and Resnick (2012)'s CT framework to examine students' CT practices through collaborative design activities with Scratch, which involves planning and coding stages to extend the implication and understanding of CT.

Methods

We conducted the pilot study in the after-school program in a suburban school district in a midwestern U.S. state, Indiana. The data were collected in two two-hour sessions in which nine 6^{th} to 8^{th} graders in three groups used Scratch to remix and design a game project collaboratively. Students had varied programming backgrounds and the groups were a mixture of novice to experienced students. The sessions were video-recorded and transcribed to examine students' computational thinking practices through collaborative design activities with Scratch. Scratch (https://scratch.mit.edu/) is a block-based programming language and is designed to enable users to learn computational concepts, while also including problem solving, creative learning, and systemic reasoning (Resnick et al., 2009). During the collaborative design processes, students used 3D printed Scratch blocks, papers, and pens to brainstorm and plan their project ideas and design elements (e.g., backdrops, game components) (planning stage), and then they moved to Scratch to implement their design (coding stage).

We used small group as the unit of analysis to examine students' CT practices through collaborative design activities. We applied Brennan and Resnick (2012)'s CT framework to investigate students' CT practices. The data were analyzed based on a qualitative coding process and coded with small segments to identify CT practices within the group. Additionally, in efforts to examine CT practices during collaborative design activities, we utilized iterative content coding to identify the patterns of CT practices throughout the design stages.

Findings

Preliminary coding results showed that patterns of CT practiced emerged through the collaborative design processes. For example, all three groups showed a higher number of experimenting and iterating in the planning stage while they brainstormed the project. Particularly, we found that CT practices demonstrated different levels of complexity in different design stages (planning and coding). All three groups showed experimenting and iterating in both planning and coding stages; however, compared to the experimenting and iterating practices in the planning stage, students showed a deeper level of experimenting and iterating in the coding stage (see examples below). In the planning stage, students identified concepts of their project and developed a script to implement the design. In the coding stage, they were able to experiment and iterate their design by identifying the variables of the script and developing a plan to modify the variables. These two stages involved different levels of CT practice regarding experimenting and iterating. Examples below showed the excerpts which students applied experimenting and iterating in two design stages (planning and coding). The group was designing a maze game in which they tried to add moving obstacles (birds) to increase the difficulty of the game.

Planning stage (off-screen, working with 3D-printed blocks, papers, and pens):

P1 Sarah: Okay, so we can start from here (point on the paper), so they can go either this way or that way

P2 Zach: I think we can put a lot of birds (*Authors' Note (AN)): moving obstacles*) here…

P3 Lisa: Here, in the middle.

P4 Zach: Or here, something like that.

P5 Sarah: Like make it (*one of obstacles*) disappears.

P6 Zach: Or make it move slower. Maybe birds over there can move really fast…

Coding stage (on-screen, working with Scratch):

C1 Zach: I think I will set it (*start point of the maze*) as 0 and we want to move it here. When the player hits the bird, he will go back to the original location, so that's the location I want. This is the end point. End point is…

C2 Lisa: 212 and -25.

C3 Zach: So you will move from left to right, and I want it move back to...

C4 Sarah: -25

C5 Zach: Something. They will move. That's really fast (check the sprite variable). So I want to change the time (*variable*) to make it slower.

C6 Sarah: That makes sense.

Discussion

The pilot study serves as a springboard to explore how computational thinking is practiced through collaborative design activities. This study was part of a larger study. The preliminary results supported that patterns of CT practices were emerged across two design stages. Particularly, students demonstrated different levels of CT practices in different design stages which inferred that different collaborative design activities might facilitate different level of CT practices. The results show great promise for future research to further investigate how different collaborative design activities promote different aspects and levels of CT practices, and to what extent that the design of these activities may advance collaborative CT practices.

References

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada* (Vol. 1, p. 25).

Chowdhury, B., Bart, A. C., & Kafura, D. (2018, February). Analysis of Collaborative Learning in a Computational Thinking Class. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (pp. 143-148). ACM.

Grover, S., Cooper, S., & Pea, R. (2014, June). Assessing computational learning in K-12. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 57-62). ACM.

Hennessy, S., & Murphy, P. (1999). The potential for collaborative problem solving in design and technology. *International journal of technology and design education*, *9*(1), 1-36.

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.

Webb, N. M. (2013). Information Processing Approaches to Collaborative Learning, in In C. E. Hmelo-Silver, C. A. Chinn, C. Chan, & A. M. O'Donnell (Eds.), *The international handbook of collaborative learning* (pp. 19-40). New York: Routledge.