Integrating Computational Thinking *Within* and *Across* Disciplines in the Context of Teacher Professional Development

Amanda Nolte
University of Delaware
United States
anolte@udel.edu

Hilary Mead
University of Delaware
United States
hmead@udel.edu

Chrystalla Mouza
University of Illinois Urbana-Champaign
United States
cmouza@illinois.edu

Rosalie Rolón-Dow
University of Delaware
United States
rosa@udel.edu

Lori Pollock
University of Delaware
United States
pollock@udel.edu

Abstract: Teachers' lack of computer science (CS) content knowledge and limited opportunities to incorporate CS in existing curricula pose unique challenges at the elementary level. Despite the crucial role of professional development (PD) in preparing elementary school teachers to integrate CS in classroom instruction, there is little research documenting PD programs that focus on integration in literacy and mathematics when compared to other subject areas. In this work, we present a PD program that integrates CS with disciplinary content to support teachers as they integrate CS with literacy and mathematics in elementary school classrooms. Using data from multiple sources, we present findings from two case studies to examine the impact of the PD on teachers' integration of CS with content in lesson planning and implementation. Findings have implications related to the integration of CS in elementary school and teacher professional development.

Keywords: computer science, elementary school, teacher professional development

Introduction

Recognition of the need for computer science (CS) education in K-12 schools has driven various reform initiatives and the development of CS curriculum standards for many years now, but much of the existing research focuses on the role of CS education at the secondary level. Recently, however, there has been increasing attention given to the important role of CS education in the elementary grades. Elementary schools are a natural starting point for CS education because the early years of schooling play a crucial role in developing positive attitudes towards CS amongst young students (Ching et al., 2018).

Teachers' lack of CS content knowledge and limited opportunities to incorporate CS in existing curriculum pose unique challenges at the elementary level (Rich et al., 2021; Rich & Hu, 2019; Repenning et al., 2015). In order to prepare elementary teachers to provide CS education to students, professional development (PD) that directly addresses these challenges is necessary (Mouza et al., 2021, 2022; Vivian & Falkner, 2019). For instance, while elementary school schedules dedicate a majority of time to high-accountability subjects such as literacy and mathematics, teachers who are equipped to integrate CS in core content instruction can overcome limited opportunities to incorporate CS in required curriculum (Century et al., 2020; Waterman et al., 2020).

Despite the crucial role of PD in preparing elementary school teachers to integrate CS in classroom instruction, there is little research documenting PD programs that focus on integration in literacy and mathematics when compared to other subject areas, such as science. In this work, we present a PD program that integrates CS with disciplinary content to support teachers as they integrate CS with literacy and mathematics in elementary school classrooms. Specifically, we describe the structure of the PD delivered during the 2021-2022 academic year and explore the following questions through two case studies: (1) *In what ways was CS integrated into fifth grade core content lesson planning and instruction in two implementations?* and (2) *How did teachers apply concepts from the PD in their lesson planning and implementation? What guided their decision-making?*

Context and Methods

Description of PD Program

Existing research indicates that high quality PD is characterized by a focus on content, pedagogy, active learning, coherence, and sustained duration (e.g., Desimone, 2009; Rich et al., 2021). Our PD program included a simultaneous focus on CS content knowledge, pedagogical practices for integrating CS with content (e.g., literacy and math), and culturally responsive pedagogy (CRP) in relation to computing. It included opportunities for active learning (e.g., hands-on investigations) as well as time to integrate CS into existing literacy and math curricula that teachers were expected to teach back in their classrooms. Further, the program extended over a 1-year period allowing teachers time to build their knowledge and apply new learning into practice.

In the fall/winter, teachers met for four 2-hour sessions where they were introduced to CS content and tools (e.g., object-oriented programming, physical computing), CRP, and associated instructional resources to support integration. The main objective was for teachers to design and implement instructional plans that integrated CS in literacy or mathematics. In the spring, they engaged in monthly one-on-one meetings with members of the research team to discuss their instructional plan, receive personalized technical support, and plan for implementation.

Table 1. PD Design

PD Sessions (November – February)			
	CS	Content	CRP
Session 1	Introduction to CS Unplugged & CS tools	Aligning sample CS lessons with math and literacy standards	Introduction to CRP: Role of personal identities in planning & instruction
Session 2	Programming in Scratch: Remixing	CS-integrated instructional resources	Modeling CRP in a CA-integrated lesson: Scratch Animate a Name
Session 3	Programming in Scratch: Classroom logistics	CS lesson from the perspective of a learner	Adapting CS-integrated lessons to incorporate CRP
Session 4	CS-integrated lesson planning: Exemplars, collaborative lesson planning, and whole group share-out		CRP strategies: Review & quick reference resource
Individual Teacher Consultations, Lesson Planning, & Classroom Implementation (February – May)			

Participants

The three teachers in this study participated in the full PD program during the 2021-2022 academic year. These teachers attended the fall and winter face-to-face sessions, designed and implemented lessons, and facilitated data collection. Among the group of five PD participants, they were chosen because they were the only ones who fully implemented the plans developed in this PD. Courtney is a fifth grade teacher at Burris Elementary, a diverse public school, where she teaches literacy and mathematics. At Burris, about three quarters of the student population are students of color, and over a third are classified as low income. Ana and Emma both teach at Myrtle Academy, an independent school serving students pre-K-12, located just outside a major city. Ana is a fifth-grade math teacher, while Emma teaches a standalone CS course. All of Ana's students are also enrolled in Emma's CS class, which was the motivation for working together in this lesson.

Data Collection

Qualitative data were collected from four sources during the 2021-2022 school year: 1) teacher lesson plans; 2) field notes from lesson observations; 3) teacher interviews; and 4) student interviews. Semi-structured interviews conducted with each of the three teachers included 12 questions exploring their perceptions of the PD overall as well as specific themes within it; their experiences planning and implementing their integrated CS lessons; their students' responses to these lessons; and their needs for follow up. Semi-structured interviews were also conducted with Courtney's students (n = 4) and students shared by Ana and Emma (n = 10). For Courtney, all students whose parents/guardians returned consent forms were interviewed. Ana and Emma had a large number of students with parental consent, so Ana chose a sample of ten students who were available during the interview timeframe. Students were asked nine questions related to the CS lessons and their experiences with and perceptions of coding. Interviews were conducted via Zoom, recorded, and then transcribed.

Data Analysis

This research is a multiple case study in which each instructional implementation constituted a case. Courtney is the first case. Since Ana and Emma developed their plans in concert with each other and taught them sequentially to the same group of students, they together make up the second case. Case study is appropriate for our goals to show what particular instances of integrating CS into elementary content instruction looked like and how teachers experienced them (Yin, 2002; Merriam, 1998).

Lesson plans and observations were analyzed using a deductive content analysis approach (Elo & Kyngäs, 2008). A deductive approach was appropriate given our research questions, which specified core content and CS as *a priori* categories of interest. One author analyzed each lesson plan and set of observation field notes independently to identify all references to core content and CS, as well as instances in which references to CS and core content co-occurred. Observation field notes were then compared to the corresponding lesson plan to evaluate alignment of CS-integration across planning and instruction. For the second case study, an additional layer of analysis was conducted to compare lesson plans and observations for each teacher. Annotations were used to track instances of content, CS, and CS-integration in each data source and analytic memos were used to reflect on findings throughout the process.

Teacher interviews were transcribed and analyzed in Dedoose (SocioCultural Research Consultants, 2023). Structured codes based upon prior work (Mouza, 2022) were used for the 'parent codes' or major ideas. Within each parent code, subthemes were identified using open coding, and constant comparisons were made between codes within a single transcript and between excerpts from the three transcripts. Analytical memoing was used to distill assertions from the coded data. One author analyzed the teacher interviews and wrote memos, which were then shared and discussed with the full team.

Findings

Case Study 1: Courtney at Burris Elementary

Lesson Planning and Implementation

While Courtney teaches both fifth grade English Language Arts (ELA) and math, she chose to integrate CS into an ELA lesson. Courtney's lesson addressed content standards that require students to use narrative techniques to show characters' responses to situations. Drawing on a book the class previously read together, Courtney planned for students to engage with the content objectives by using Scratch, a block-based programming tool, to develop and animate dialogue that might occur between themselves and the main character. Her lesson plan achieved the PD goal that teachers integrate CS with content instruction, rather than including each component separately.

Courtney's lesson implementation was closely aligned with her plan, maintaining the integration of content and CS. During observations, we noted that Courtney also demonstrated integration beyond the parameters of her plan by pausing on several occasions to ask students about the connection between the content objectives and their Scratch projects. On the other hand, Courtney's selection of a narrow content area objective and her students' limited prior experience with Scratch worked together to limit students' active engagement with coding. While the lesson spanned two days, the first day was spent reading and discussing the book that provided lesson context and briefly introducing students to Scratch. Therefore, students spent only one class working in Scratch.

Courtney's lesson plan and implementation strongly suggested that she assumed students had no prior experience with Scratch. On the first day of the lesson, she showed a video to introduce students to Scratch. On the second day, when students worked in Scratch to create their projects, Courtney guided them through the process via an instructional video, pausing after each step to allow students to follow along. However, observation of her lesson implementation revealed that some students had prior experience working in Scratch. Multiple students also worked ahead, though this could be attributed either to prior experience or an intuitive understanding of the platform.

Although content and CS were integrated, students engaged with the coding activity for a limited amount of time. Students began their project on the day that the lesson was observed but did not spend the entire class period working in Scratch. Courtney paused the class to ask students about their perception of the difficulty of the project and provide the option of reading or adding to their project for the remainder of the class period, suggesting that she expected students to have completed the project. According to her lesson plan, students were required to animate at least three lines of dialogue for each of two sprites. It is reasonable to assume that once a student figured out how to code one line of dialogue, they would have been able to complete the remainder of the assignment quickly.

Teacher Learning

Courtney described CS as far "outside [her] comfort zone" and identified herself as a total beginner: "I really didn't know anything about computer science, programming." She chose Scratch as a tool to integrate because it was accessible to beginners (as well as free and available): "I felt like the day [during the PD] we just focused on Scratch and did stuff I was getting it pretty quickly, and I could see my kids getting it." Courtney also capitalized on existing instructional resources for Scratch, which a PD facilitator shared with her during individual consultation. Courtney framed that activity as an effective support: "Not that I didn't do anything, I wrote the lesson plan, but the computer science part was done for me. That's what I need. …You need something that's fast and easy to use."

Courtney described herself as a teacher who preferred teaching ELA to math, but her choice of content area for integration was more practical. In a chaotic year, it was hard to predict her class pacing in math, and she felt that ELA had more flexibility in terms of the skills and sequence: "I was like 'I can't plan this lesson on volume and then…when I go to teach it, we're in fractions. I felt like pacing wise, I can always teach an ELA lesson, you can always do a read-aloud." Interestingly, although Courtney's lesson integrated CS in ELA, she opened it with discrete disciplinary "chunks." She did a read aloud of a mentor text, introducing the character of the "Invisible Boy." She then shared an instructional video about how to use Scratch. On the following day, students integrated these two strands as they created characters and backgrounds in Scratch and then wrote and coded a dialogue.

As Courtney reflected on her lesson, she noted several ways in which the implementation experience diverged from her plans and changed the typical classroom dynamic. As our observation noted, two students brought prior background with Scratch, and thus could take on new leadership among their peers: "They're not kids who are always up and about, but they had the skill that the other kids didn't have and were helping everyone." Second, the project contrasted with the more structured tasks that dominated the post-pandemic school year: "They're so far behind…this year's been tough. And so it was fun to do something different." Courtney saw her students thrive on the creativity and openness of the task. Third, Courtney's own newness to coding seemed to support this student-centered approach as she positioned herself as a co-learner, not as the source of knowledge in the classroom.

Case Study 2: Emma & Ana at Myrtle Academy

Lesson Planning and Implementation

Ana and Emma, who teach the same group of fifth grade students, worked together to integrate CS and content. Ana provided content instruction in her math classes, and Emma subsequently worked with students on a CS application of the math content during her CS classes. Ana's lesson plans and implementation indicated that she exclusively provided math instruction. Her role was to equip students with the content knowledge needed to complete a CS project in Emma's class. Neither Ana's plans nor her implementation incorporated CS, demonstrating her reliance on Emma to integrate CS and content. On the other hand, Emma integrated content with CS by planning and implementing a lesson in which students animated geometric transformations in Scratch. Ana's lesson spanned three days, and while Emma's was planned for about three days, its implementation took five days.

Emma incorporated math content in her CS classes by leveraging instruction previously provided by Ana. However, the math concepts included in Emma's lesson functioned primarily as reinforcement, rather than integration of CS with content instruction itself. The collaboration between the two teachers, therefore, limited the extent to which integration was apparent within each of their classes. Integration was further limited by the timing of implementation in the teachers' respective classes. A three-week gap between Ana's and Emma's lesson implementations contributed to a sense that the content and CS were presented as two independent components.

Observations of Emma's lesson implementation clearly indicated her expectation that students had a strong understanding of the math content covered in Ana's class. Emma told students that their goal was to create a project that would adequately explain geometric transformations to someone unfamiliar with the concepts. This is a higher order task that requires students to have significant prior content knowledge. Emma demonstrated the role of math content in her CS lesson by displaying visual representations of transformations and asking questions to confirm students' content knowledge. On two occasions, however, Emma prompted students to explain that they could not use squares or equilateral triangles in their projects because both "remain the same shape" when transformed. In fact, by definition, a shape should remain the same when transformed. These exchanges suggest further opportunity for collaboration with Ana to ensure that content knowledge was accurately conveyed and reinforced.

As students were working, Emma occasionally paused while helping an individual student and addressed the whole class. On at least one occasion, she clarified a point related to content, stating that "the reflection is flipped. When we flip it, it's the other side." However, the majority of her clarifications address CS-related project requirements, such as the number of costumes students must include for their sprites. Emma's lesson plan suggested that students had significant prior experience with Scratch and were well-equipped to meet CS-related project requirements, which was confirmed during observations of her classes and student interviews.

Teacher Learning

Both teachers addressed the choice of Scratch as a CS tool. Emma commented, "I think that Scratch is the foundation – a logical place to begin with CS." For Ana, Scratch seemed like the tool that would lend itself most easily to mathematical content: "That seemed to be the program that would facilitate whatever topic I wanted to teach." Emma had over a decade of experience using and teaching with Scratch, while Ana had had a "trial run" with the program the prior year. Myrtle students, too, were familiar with the program: "the kids use it a lot." Myrtle had an unusual level of Scratch background knowledge among both its teachers and its students.

For Ana, the choice of math for integration was self-evident, but Emma expanded on the possibilities of different content areas. As a technology teacher, she advocated for embedding concepts from other disciplines into her projects for two reasons. First, to "kind of break the stigma of specials classes, like 'technology is just a special,'" and second, to prepare students for a "multidisciplinary workplace." While this project involved Scratch as a tool for learning math content, Emma had experiences with other content areas. In fact, her first collaboration of this sort was with an English as a Second Language teacher in her home country. She expanded: "..people, might relate Scratch… coding with math. And we need to break that paradigm, that stereotype."

During the actual implementation, Ana first taught students the math concepts that would later be applied in Emma's CS class. While the implementation was largely sequential, there was some collaboration and communication across teachers. Emma observed some of the math classes to build her own understanding of the content. Later, Ana gave some class time to allow students to finish their projects and/or to share their completed work with her: "Emma came in during each of the classes and…if [students] weren't finished their project, then I was available and Emma was available to go around and answer questions." The teachers also described informal conversation that began during the PD sessions but continued at school.

Discussion & Conclusion

The overarching theme that emerged from the two cases was the contrast between Courtney's integration of CS *within* her content area class and Emma and Ana's CS integration *across* their respective content and CS classes. As a generalist classroom teacher, Courtney leveraged the opportunity to integrate CS *within* core content, specifically ELA. The course structure at Myrtle, on the other hand, in which all students were enrolled in both math and CS courses, facilitated an opportunity for Ana and Emma to integrate content with CS *across* their courses.

Our findings on teachers' decision-making processes have implications for future PD. Participants indicated that decisions they made during lesson planning and implementation were influenced by their familiarity with CS tools, as well as the time and flexibility afforded in their specific contexts. Their previous experiences with CS tools informed their beliefs that Scratch naturally lends itself to math content and provides a foundation for more complex programming. Courtney's decision, however, was driven by her lack of familiarity with CS tools and her perception that Scratch was accessible for beginners. Her rationale is particularly revealing; it shows that both teacher confidence and predictions about students' success inform teachers' approaches to integrating CS.

The additional time and flexibility offered by the standalone CS course at Myrtle allowed Emma and Ana to integrate content and CS collaboratively and provided time for students to engage in more sustained and in-depth CS-integrated lessons. However, the standalone CS course seems to have removed the impetus for Ana to integrate CS in her content instruction. Rather, Emma's existing role as the CS teacher established her responsibility to provide all instruction that included CS. Nonetheless, Emma and Ana adapted the *within* class model to integrate CS and content to an extent *across* their respective classes. Courtney cited the availability of pre-existing instructional resources as an influential factor in her lesson design. The resources helped Courtney identify opportunities to integrate CS while attending to curriculum requirements and pacing that afforded less flexibility in her school context. At the same time, these constraints limited the depth of engagement included in her lesson.

The contrasting integration models illustrated in these two cases provide insights into teacher learning about the integration of CS and content. Future PD should include models for CS integration both *within* content area classes and *across* classes, particularly in schools with standalone CS courses. Additionally, PD should reflect factors involved in teachers' decision-making processes as they plan and implement CS-integrated lessons. Teachers' decisions during the lesson planning process may be influenced by their own comfort level with CS tools, as well as the availability of existing instructional resources that can help teachers identify opportunities to integrate CS while meeting demands posed by required standards and curricula. Given the wealth of resources, a shift in focus toward adapting pre-created lessons, rather than creating new lessons, may be warranted.

Acknowledgements

This work is funded by a grant from the National Science Foundation, Award# 1923483.

References

Century, J., Ferris, K. A., & Zuo, H. (2020). Finding time for computer science in the elementary school day: A quasi-experimental study of a transdisciplinary problem-based learning approach. *International Journal of STEM Education*, *7*(20), 1-16. https://doi.org/10.1186/s40594-020-00218-3

Ching, Y. H., Hsu, Y. C., & Baldwin, S. (2018). Developing computational thinking with educational technologies for young learners. *TechTrends*, *62*(6), 563-573. https://doi.org/10.1007/s11528-018-0292-7

Desimone, L. M. (2009). Improving impact studies of teachers' professional development: Toward better conceptualizations and measures. *Educational Researcher, 38*(3), 181–199.

Elo, S., & Kyngäs, H. (2008). The qualitative content analysis process. *Journal of Advanced Nursing, 62*(1), 107–115. https://doi.org/10.1111/j.1365-2648.2007.04569.x

Merriam, S. (1998). *Qualitative research and case study applications in education.* Jossey-Bass.

Mouza, C., Codding, D., & Pollock, L. (2022). Investigating the impact of research-based professional development on teacher learning and classroom practice: Findings from computer science education. *Computers & Education.* https://doi.org/10.1016/j.compedu.2022.104530

Mouza, C., Mead, H., Alkhateeb, B., & Pollock, L. (2022). A virtual professional development program for computing education: Design principles and lessons learned. *TechTrends, 66,* 436-449.

Rich, P.J., & Hu, H. (2019). Surveying the landscape: Statewide data on K-12 CS education implementation. *Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT),* (pp. 1-8), IEEE.

Rich, P.J., Mason, S.L., & O'Leary, J. (2021). Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking. *Computers & Education, 168*, (2021), 104196.

Repenning, A., Webb, D.C., Koh, K.H., Nickerson, H., Miller, S.B., Brand, C., ..(2015). Scalable game design: A strategy to bring systemic computer science education to schools through game design and simulation creation. *ACM Transactions on Computing Education, 15*(2), pp. 1-31.

Vivian, R., & Falkner, K. (2019). Identifying teachers' technological pedagogical content knowledge for computer science in the primary years. *Proceedings of the 1019 ACM Conference on International Computing Education Research* (pp. 147-155). ACM.

Waterman, K. P., Goldsmith, L., & Pasquale, M. (2020). Integrating computational thinking into elementary science curriculum: An examination of activities that support students' computational thinking in the service of disciplinary learning. *Journal of Science Education and Technology*, *29*(1), 53–64. https://doi.org/10.1007/s10956-019-09801-y

Yin, R. (2017). *Case study research and applications* (7th edition). SAGE Publications.