TechTrends (2022) 66:436–449
https://doi.org/10.1007/s11528-022-00731-y

ORIGINAL PAPER

A Virtual Professional Development Program for Computer Science Education During COVID-19

Chrystalla Mouza[1] · **Hilary Mead**[2] · **Bataul Alkhateeb**[1] · **Lori Pollock**[3]

Accepted: 25 March 2022 / Published online: 24 May 2022

Abstract

The need to expand computer science learning for all students has led to an increase in professional development (PD) opportunities for teachers. The Covid-19 pandemic, however, necessitated changes in well-established PD programs and a shift to virtual delivery. In this work, we describe our transition to a virtual PD institute, including the topics and design principles guiding the institute. We also examine how participation in the virtual PD institute influenced teacher outcomes. Data were collected from two cohorts of teachers. Data sources included surveys (N=30), lesson plans (N=22), and interviews (N=17) from a purposeful sample of participants. Findings gleaned from quantitative and qualitative analysis suggest an increase in teachers' knowledge and self-efficacy while highlighting the affordances of virtual PD most valued by teachers. Findings have implications for research and practice.

Keywords Professional development · Virtual learning · Computer science education

Introduction

Throughout the world, efforts are underway to expand K-12 student participation in computer science (CS). In the U.S., 51% of high schools offer CS while across 19 states, 30% of K-8 schools offer foundational CS courses (Code.org et al., 2021). Although progress has been made, data suggests that opportunity gaps continue to exist in relation to female and minoritized student participation in computing (Code.org et al.). For example, minoritized students are less likely to attend a school that offers CS courses while females are less likely to be enrolled in foundational CS courses (Code.org et al.). At the same time, global events such as the Covid-19 pandemic have further highlighted the critical role of CS knowledge in helping individuals understand data and computer models used to explain the pandemic itself, the economy, and the well-being of citizens (Lee & Campbell, 2020). Further, with existing inequalities exacerbated during the pandemic, understanding of how to use CS to identify the impact of complex societal problems on one's communities has made computing more personally relevant to students (Lee & Campbell, 2020). In turn, these opportunities necessitate that teachers engage *all* students with CS in ways that help them address pressing societal issues.

A major challenge in engaging *all* students with CS is building teacher capacity for equitable, engaging, and rigorous CS instruction. Towards this end, culturally relevant pedagogy (CRP) frameworks which help teachers integrate knowledge relevant to student identities and communities with CS activities are key for enabling effective, meaningful, and equitable learning (Codding et al., 2021; Gay, 2018). Equity-oriented PD prepares teachers to integrate CRP into their teaching by engaging them in self-reflection that requires acknowledging their biases and positionality (Borrero et al., 2018). Yet compared to other subjects (e.g., mathematics), there are fewer opportunities as well as research

---

✉ Chrystalla Mouza
cmouza@udel.edu

Hilary Mead
hmead@udel.edu

Bataul Alkhateeb
bataul@udel.edu

Lori Pollock
pollock@udel.edu

[1] School of Education, University of Delaware, Newark, DE 19716, USA

[2] Center for Research in Education & Social Policy, College of Education & Human Development, University of Delaware, Newark, DE 19716, USA

[3] Computer & Information Sciences, University of Delaware, Newark, DE 19716, USA

documenting the design, implementation, and outcomes of equity-oriented CS PD programs (e.g., Codding et al., 2021; Menekse, 2015; Mouza et al., 2018). Importantly, the Covid-19 pandemic necessitated changes in the design and delivery of PD programs, further complicating current efforts to investigate the characteristics of effective CS PD and its associated impacts on teacher learning and practice. In this work, we present the transition from a face-to-face program to a virtual PD institute necessitated during the Covid-19 pandemic. We describe the topics and design principles guiding the virtual pivot and investigate the impact of the virtual PD institute by answering the following questions:

1. How did participation in a virtual PD institute influence teachers' knowledge and confidence in teaching CS skills?
2. How did participating teachers apply CS content and pedagogy learned during the virtual PD institute in conceptual lesson designs?
3. In what ways, if at all, did participating teachers use or plan to use new resources and strategies from the virtual PD institute in their instruction?
4. What elements of the virtual PD institute were most valued by participating teachers?

Prior Literature: Professional Development in Computer Science Education

The current demand for K-12 CS instruction has given rise to a number of PD initiatives both in the U.S. and abroad. PD is essential for helping teachers build their knowledge of CS content along with high quality pedagogical strategies that will help meet the needs of all students (Goode et al., 2014; Fancsali et al., 2020). Further, PD is important for positively influencing teachers' confidence or self-efficacy -- their beliefs in their ability to both learn and teach CS (Bandura, 1977; Rich et al., 2021). A growing body of research indicates that teachers' self-efficacy has implications for teaching effectiveness, instructional practices, and student achievement (Klassen & Tze, 2014). Therefore, efforts to support the teaching of CS should focus on teachers' knowledge and confidence as well as their ability to apply what they learn in practice.

Findings from Face-to-Face PD Efforts

Existing research indicates that participation in PD driven by a focus on content, pedagogy, active learning, coherence, and sustained duration could facilitate changes in teachers' knowledge, self-efficacy, and practice in a variety of K-12 settings (e.g., Desimone, 2009; Goode et al., 2014; Rich et al., 2021). For instance, in prior work we have found that participation in a PD program which included an intensive face-to-face week-long institute and follow-up support during the academic year, helped teachers acquire CS content knowledge and a variety of pedagogical strategies for teaching CS (Mouza et al., 2016; Pollock et al., 2017). Further, interview data following participation in PD indicated that teachers implemented content and strategies from the PD in their classrooms, reaching a broad population of students (Pollock et al., 2017).

Findings from PD efforts associated with the popular Exploring Computer Science (ECS) high-school curriculum indicated that participation in PD strengthened teachers' practice by building their capacity to foster equity, inquiry, and development of CS concepts. In turn, teachers' practice had a direct impact on students' attitudes and their engagement with CS (McGee et al., 2019). Similarly, Rich et al. (2021) documented positive changes in teachers' learning as a result of participation in continuous PD associated with the BootUp K-8 program (https://bootuppd.org/professional-development/). The BootUp PD program extended over the period of one year and focused on coding and computational thinking, both of which are key aspects of CS. Results indicated that participation in the program increased confidence, self-efficacy, and comfort level with coding and computational thinking.

Findings from Virtual PD Efforts

Although these results are promising, the Covid-19 pandemic has forced a number of well-established face-to-face PD efforts to transition online. While there is an extensive body of literature focusing on online PD in other content areas (e.g., Parsons et al., 2019), such work is only beginning to emerge in the field of CS. For instance, Goode et al. (2020), transitioned the well-established ECS PD sessions from face-to-face to online focusing on a learning community model to support continuous participation while addressing barriers related to travel, time, geography, and cost. Findings indicated that the learning community approach can engage participants in learning that supports inquiry and equity-based CS instruction while breaking down the isolation of CS teachers (Goode et al., 2020).

Other efforts to leverage online PD are also reported from scholars outside the U.S. (e.g., Ireland, Australia). For instance, McGarr et al. (in press) describe a PD framework focusing on a community of practice approach to support nationwide CS efforts in Ireland. This approach included a combination of face-to-face and virtual just-in-time support delivered through collaborative platforms (e.g., SLACK), as well as online MOOCs focused on the development of fundamental CS skills. Similarly, Falkner et al. (2018) describe an ecosystem approach to supporting teacher PD in Australia, which includes a collection of MOOCs providing self-paced content and pedagogical learning, development of lead teachers, face-to-face PD events, and a lending library of CS technologies to facilitate integration. Evaluation results indicated that an ecosystem approach can support teacher learning and engagement with CS while strengthening the broader CS community goals.

Findings from Emergency Virtual PD Efforts

While the above studies examined the effectiveness of PD deliberately designed and delivered online, the Covid-19 pandemic necessitated emergency remote instruction, which is meaningfully different (Hodges et al., 2020). Ferdig et al. (2020) provided numerous examples of how educators working with both pre-service and in-service teachers pivoted to online instruction as a result of the pandemic, but few of these studies focused on CS PD specifically (e.g., Albert et al., 2020).

Examining their transition to virtual PD specifically in response to Covid-19, Jocius et al. (2021) describe how they shifted an established PD program to online focusing on the integration of computational thinking with English Language Arts (ELA). This program included face-to-face summer workshops and follow-up academic year activities modeled after the 3C approach (code, connect, and create). The online transition used a combination of synchronous and asynchronous activities delivered through various technologies (e.g., video conference, chat), online materials mirroring the 3C approach, opportunities for collaborative activities that supported engagement with content, and focus on practice and reflection. Follow-up activities were provided through monthly webinars focusing on CS content. Findings indicated that teachers infused CT into existing ELA curricula, albeit they differed in the manner in which they appropriated and adapted pedagogical tools for CT infusion.

In this work, we describe how we transitioned our face-to-face PD program to online delivery at a time where both the emotional strain and workload of teachers made it challenging to focus on CS in light of pressing educational challenges (Code.org et al., 2021; Jocius et al., 2021). We subsequently examine the impact of the virtual PD institute on teachers' knowledge and confidence in teaching CS, application of CS in lesson designs, and PD features most valued by teachers.

Methods

Context of this Work

This work is situated in a larger effort to improve the teaching of computing in the U.S. through effective PD. Our PD program incorporates a three-tiered approach, which typically includes: (a) an annual face-to-face week-long summer institute, (b) a college field-experience course in which undergraduate students with background in CS assist teachers in developing and implementing CS lessons back in their classrooms, and (c) sustainable partnerships with K-12 schools (Pollock et al., 2015). The week-long summer institute focuses on preparing teachers in grades 5-12 to teach standalone CS courses or integrate foundational CS principles into existing content area instruction. These principles include creativity, abstraction, data, algorithms, programming, Internet, and impacts of computing (College Board, 2017).

The design of the institute is grounded in the framework of technological pedagogical content knowledge (TPACK), which describes the knowledge needed for the effective integration of technology in teaching (Mishra & Koehler, 2006). It includes explicit attention on CS resources and tools (technology), CS principles (content), pedagogy for teaching CS (pedagogy), and culturally responsive strategies for broadening participation in computing (Codding et al., 2021; Scott et al., 2010).

In spring 2020, as Covid-19 forced school closures, we quickly realized that we would not be able to offer our annual face-to-face summer institute. Following discussions with our local chapter of the Computer Science Teachers' Association (CSTA), we decided to shift our summer institute to an institute series of bi-weekly 2-hour virtual workshops, delivered synchronously via Zoom, in fall 2020. A total of five workshops were offered between October - December for a total of 10 instructional hours. Similarly, as summer 2021 approached it again became clear that a face-to-face institute would not be possible. Therefore, we pivoted to a virtual summer institute offered over a period of four days. The design of the summer institute was informed by feedback on the content, duration, and instructional activities provided by fall participants. Each day, teachers met synchronously via Zoom from 9 a.m. – 12 p.m. for a total of 12 instructional hours. All sessions were delivered by members of our research team or teachers who had developed leadership and expertise through their participation in our prior PD efforts.

In addition, teachers participating in both the fall and summer virtual institutes devoted independent time for examining resources and tools and designing lessons that integrated CS with core curriculum content. This approach allowed us to capitalize on a combination of synchronous activities focused on new content and peer interactions with asynchronous activities focused on practice and reflection (Jocius et al., 2021). Further, technical assistance was available throughout each day of the summer institute in a separate Zoom room, which allowed teachers to receive one-to-one support or ask questions at any time. Such assistance was provided on-demand in the fall.

Table 1 Content, delivery, and rationale of transition to online PD

Face-to-face component	Reasoning/Literature support	Changes for virtual PD
Opening Session Time: daily 30-minute sessions. Content: Introductions and Logistics	Introductions Logistics Overview	Virtual Opening Session (Synchronous) Time: 30-minute session Content: Introductions and Logistics
CS Unplugged Group Activities Time: daily 30-minute sessions. Content: CS Unplugged activities that connected with the CS Big Idea of the day.	Introducing CT concepts using metaphors to create connections Bell & Vahrenhold, 2018).	Virtual CS Unplugged (Synchronous) Time: two 30-minute sessions Content: CS Unplugged activities that can be delivered in an online environment: Cryptography and Product Code Magic (Fall 2020) / Cryptography and Info hiding protocols (Summer 2021).
Culturally Responsive Pedagogy Time: daily sessions, ranging from 30-75 minutes. Content: Activities centered on promoting diversity, self-reflection, centering equity, and implementation.	Expanding access, interest, and identification in CS Broadening participation in computing (Alvarado et al., 2012; Ladson-Billings, 1995; Pollock, 2008; Scott et al., 2010).	Culturally Responsive CS Pedagogy (Synchronous) Time: one 45-minute session (Fall 2020), two 45-minute sessions (Summer 2021) Content: culturally responsive computing, activities focused on designing supportive learning environments, student-centered instruction, structural inequities, and CS impacts (Fall 2020 & Summer 2021) Content: CS careers and impacts of broadening participation in computing (Summer 2021).
Programming Time: two 60-minute sessions. Content: Introductory and Advanced programming exercises using Scratch.	Teachers learn programming skills (e.g., content knowledge) Teachers learn pedagogical approaches to teaching programming (e.g., scaffolding), including equity pedagogies (e.g., pair programming; Madkins et al., 2020).	Programming (Synchronous) Time: two 60-minute sessions Content: Beginner, Intermediate, Advanced Sessions facilitated by teacher leaders using breakout rooms in Zoom (Fall 2020) Content: Choice of Scratch or Creating with Artificial Intelligence (Summer 2021).
Introduction to Various CS Tools Time: two 90-minute sessions. Content: Beebots, Micro:bits, Ozobots, MakeyMakey.	Not all teachers have access to the same funding and/or tools. Providing a broad range of examples of tools allows for teachers to have a greater understanding of what is out there and what they might be able to use.	CS Tools with Online Components (Synchronous) Time: one 2-hour session (Fall 2020) / 3 45-minute sessions (Summer 2021) Content: Micro:bits and Twine facilitated by teacher leaders using breakout rooms in Zoom (Fall 2020) Content: Micro:bits (pocket-size computers), Web Design, Coding with music (Summer 2021).
Incorporating CS Standards into Core Lessons Time: one standalone session and opportunities for discussion throughout the week. Content: Connecting CS to content-area instruction.	Providing connections to teaching expectations (e.g., Coherence).	Examining Standards and Examples of CT (Synchronous & Asynchronous) Time: one synchronous 60-minute session and asynchronous independent work on lesson development. Content: CS standards developed by CSTA (2017), content area standards, and CS lessons appropriate for different grade levels.
Collaborative Lesson Planning Time: one stand-alone session and independent work throughout the week. Content: Development of lessons that integrate CS with content-area instruction within one's own context.	Connecting CS with existing content-area instruction (Yadav et al., 2016) to demonstrate coherence and engage in active learning.	Offline Lesson Planning Individually or in Teams - (Asynchronous) Time: Independent, asynchronous work - approximately 1 hour per PD session Content: Development of lessons that integrate CS with content-area instruction within one's own context.

Table 2 PD participant demographics

		Number	Percentage
Gender	Female	33	82.5
	Male	7	17.5
Race	Asian	1	2.5
	Black/African American	10	25.0
	White	27	67.5
	Prefer not to answer	2	5.0
Ethnicity	Hispanic or Latinx	1	2.5
	Prefer not to answer	1	2.5
	Other	38	95
Teaching Experience	0-3 Years	3	4.5
	4-5 Years	2	5.0
	6-10 Years	10	25.0
	11-15 Years	7	17.5
	16+ Years	18	45.0
CS Experience	0 Years	15	37.5
	1 Year	4	10.0
	2-3 Years	11	27.5
	4-5 Years	5	12.5
	6+ Years	5	12.5
Primary Discipline[a]	Computer Science	13	33.3
	Business	6	15.4
	Career/Technical Ed	5	12.8
	Science/Math	2	5.1
	Other	13	33.3

N=40 except [a] *N*=39

Table 1 provides a snapshot of the content, delivery, and rationale of our virtual institute in fall 2020 and summer 2021. Unlike our face-to-face PD program, no follow-up classroom support was provided. Due to our university classes also moving online, we were unable to offer the field-experience course.

Participants

Participants included 15 teachers who attended the fall virtual institute and 27 teachers who attended the summer virtual institute, for a total of 42 teachers. Of those, 21 (50%) teachers had previously attended our face-to-face PD programs. Those teachers taught in a variety of grade levels and schools throughout a U.S. state. Table 2 presents demographic information on participating teachers. Participants[1] were fairly seasoned teachers, with the majority of them having 16+ years of experience. However, as shown on Table 2, their experience in CS was much shorter. Only five teachers had 6+ years of CS experience. Further, only thirteen participants taught CS as their primary discipline while the rest were interested in integrating CS across other areas. All teachers voluntarily participated and were compensated for their time.

Design and Data Collection

The study used a convergent parallel mixed methods design (Cresswell & Plano Clark, 2017). Data for each research question were collected from different sources including pre and post surveys, lesson plans submitted at the end of each institute, and interviews. Qualitative and quantitative collection and analysis proceeded separately, with integration happening at the point of analysis.

Surveys Data were collected using both quantitative and qualitative procedures. Participants completed electronic surveys, administered through Qualtrics, prior to the first PD session and after the last session. These surveys were designed to gauge participants' level of knowledge and confidence in teaching CS before and after each institute. The post-survey also included both closed- and open-ended questions to measure participant satisfaction and perceptions of the PD, as well as what they had learned in it. Survey items were based on research discussed above about effective PD (e.g., *"This PD workshop was tailored to my needs as a learner"* or *"Sufficient time was provided for guided practice and tasks"*). There were 15 responses to the pre-survey and 8 to the post- survey in fall 2020. There were 25 responses to the pre- survey and 22 to the post- survey in summer 2021.

Lesson plans As a culminating activity in both the fall 2020 and summer 2021 institutes, participants developed conceptual lesson plans integrating CS with content area instruction (N=22). Conceptual lesson plans refer to lessons that teachers *planned* to use in their own classrooms following their participation in PD (Codding et al., 2021). Participants developed the lesson plans independently (N = 18) or with a partner (N = 4). These plans were used as artifacts to demonstrate participant learning and application of PD concepts. Specifically, teachers were asked to indicate the target audience (e.g., grade level), content area, lesson goals, CS standards, required technologies, and learning assessment. Teachers designed lesson plans that could be used for multiple content areas including standalone CS courses (N=15), world languages (N=3), mathematics (N=2), social studies (N=1), and ELA (N=1). Lesson plans covered a wide range of CS topics ranging from block-based coding to create stories on Scratch to writing algorithms for playing rock, paper, scissors as an unplugged activity.

---

[1] Not all participants completed all data collection instruments, therefore the sample size for each data source varies.

Table 3 Codebook for lesson plan analysis

Code	Criteria
CS Principles (CS for All Teachers, n/d)	
Algorithms	Activities use a sequence of instructions for a process that can be executed using tools and devices to solve problems.
Programming	Activities involve a creative process that enables problem solving, human expression, and creation of knowledge. Activities use mathematical and logical concepts and are facilitated by appropriate abstractions.
Abstraction	Activities use models and simulations to raise and answer questions.
Creativity	Activities foster the creation of artifacts and creative expression.
Data	Activities use computer programs to process information to gain insight and knowledge. Activities use computing facilities to explore and discover connections in information through data manipulation.
Internet	Activities use the Internet, a network of autonomous systems, and the systems built on it.
Impacts of CS	Activities explore the ways computing affects communication, interaction, and cognition. Activities explore computing within economic, social, and cultural contexts.
Pedagogy	
Collaborative Learning	Students are configured into small groups or partners.
CS Inquiry	Activities are characterized as open ended tasks that are student centered and encourage discovery. Students analyze and think critically between various solutions and approaches for completing tasks.
CS Unplugged	Activities engage students with ideas from computer science without having to learn programming or use a digital device.
Scaffolding - Use	Activities are limited to "using" a given artifact such as "inspecting a code and running existing models". Students act as "consumers of someone else's creation."
Scaffolding – Modify	Activities involve "using" and "modifying" an existing artifact. "Students go from solely using existing code to changing the code to suit their intended desires as designers."
Scaffolding - Create	Activities move away from "using" and "modifying" existing artifacts to producing new artifacts. "Students end up in a state where they have created a completely new model, having full ownership of the design and agency over its development."

Interviews Interview data helped obtain in-depth information on teachers' experiences. All teachers who attended the fall 2020 institute were invited to an interview. Of those, 8 participants (53.3% response rate) responded and were interviewed in spring 2021. Of the teachers attending the summer 2021 institute, a purposeful sample of 14 new and returning participants from diverse school settings was invited for an interview after the PD. Out of those invited, 9 responded (64.3% response rate) and were interviewed at the end of the institute. Therefore, a total of 17 interviews were conducted. The interview protocol explored how teachers responded to the PD, what they had learned in it, and whether and how they had implemented or planned to implement strategies, practices, or resources shared in the PD. Interviews took place on Zoom and were recorded and transcribed. All interviews were conducted by an external evaluator associated with a research center responsible for the overall PD evaluation. This allowed participants to freely express their perspectives of the PD without trying to please PD providers.

Data Analysis

Both qualitative and quantitative analysis were conducted depending on each data source.

Surveys Frequencies were calculated for all closed-ended items and means and standard deviations were also calculated for Likert scale ratings. Paired sample t-tests were conducted on the 21 items that appeared on both the pre- and post-PD surveys. There were a total of 28 matched pairs (i.e., where the individual completed both pre- and post-PD surveys) used in the analysis. Open-ended survey responses were coded by question, using an inductive, grounded theory approach (Saldaña, 2021; Corbin & Strauss, 2008).

Lesson Plans Conceptual lesson plans were analyzed to examine the ways CS content and pedagogical practices were infused in the conceptual lesson plans after participating in the virtual PD. For this analysis, we developed a codebook drawing from two sources (CS for all, n.d.; Lytle et al., 2019). Codes were grouped into two categories that addressed CS principles and CS pedagogy, including specific scaffolding activities modeled after the use-modify-create framework (Lytle et al., 2019; see Table 3).

Interviews Interview transcripts were coded in Dedoose (Socio-Cultural Research Consultants, 2021), using a mixture of a priori codes based on previous work (see Codding et al., 2021) and codes that emerged inductively. The goal was to identify themes salient across participants.

Table 4 Ratings of knowledge and skills before and after the workshop

	Before workshop		After workshop	
	Number	Percent	Number	Percent
Poor	1	3.40	0	0.00
Below average	6	20.70	0	0.00
Average	13	44.80	8	27.60
Above average	9	31.00	19	65.50
Excellent	0	0.00	2	6.90
Mean (S.D.)	3.03 (0.82)		3.79 (0.56)	

N=29

Findings

Findings indicated that participation in a virtual PD institute positively influenced teachers' knowledge and confidence as well as application of learning into practice. Further, teachers valued many aspects of the virtual PD.

Impact of Virtual PD on Teacher Knowledge and Confidence

Survey data provide insight into teachers' knowledge, skills, and confidence in CS. An item on the post survey asked participants to rate their knowledge and skill in CS both before and after participation in the virtual PD institute (Table 4). Results indicated self-reports of growth. The mean rating *before* the PD was 3.03 (SD = 0.82) compared to 3.79 *after* the PD institute (SD = 0.56). A paired samples t-test indicates that this difference is significant, t (28) = -5.93, p(two-tailed)<0.001. After participation in the virtual PD institute, every participant rated themselves at least average with the largest group (65.5%) selecting "above average."

Post-survey responses also indicated higher confidence teaching CS principles (Table 5). Specifically, participants' ratings of their confidence in teaching CS skills overall, as well as teaching skills related to each CS principle, were higher on the post-surveys compared to the pre-surveys. All increases were statistically significant. The largest gain was seen for the principle of programming, and the highest ratings both before and after participation in PD were for creativity.

Surveys and interviews also provide some qualitative insight into how participants gained knowledge, skills, or confidence related to teaching CS. Specifically, as they described their PD experience, participants often indicated how their own understanding had grown or how they now felt more confident or equipped to bring CS concepts into the classroom. For example, one teacher contrasted the summer institute to a previously attended PD opportunity:

> [In another workshop] I feel like I didn't learn anything. I was just sitting there. And this time I did learn things. I do have materials for planning my entire year next year. And I feel like I have a foundation to start with. When [cybersecurity presenter] shared resources with us, I told her, 'I've been a year researching...and I couldn't find anything I really liked.'

Participants brought varied prior experience related to CS (see Table 2) but the data suggest that both novices and veterans reported growth in their knowledge and confidence. For instance, survey comments focused on CS content as the best aspect of the institute came from someone calling herself "a VERY NEW beginner in the field" and from someone who appreciated "understanding advanced computer science information." Some participants identified specific CS content or topics they appreciated learning, especially artificial intelligence and cybersecurity, as well as programming with specific tools, such as Micro.bits and Scratch.

Table 5 Confidence before and after PD institute - results of paired sample T test

Items	Pre Mean	Pre SD	Post Mean	Post SD	t	*df*	P (two-tailed)
I feel confident teaching . . .							
CS Skills Overall	3.25	1.11	4.04	0.64	-5.28	27	<0.001
I feel confident teaching CS skills related to . . .							
Creativity	3.39	1.13	4.21	0.57	-4.12	27	<0.001
Abstraction	2.82	1.06	3.68	0.67	-4.34	27	<0.001
Data	2.86	1.01	3.68	0.72	-4.80	27	<0.001
Algorithms	2.71	1.15	3.64	0.95	-5.23	27	<0.001
Programming	2.82	1.09	3.89	0.96	-5.39	27	<0.001
Internet[a]	3.22	1.05	4.11	0.80	-5.45	26	<0.001
Impacts of CS	3.14	1.30	4.11	0.83	-5.31	27	<0.001

N=28 except [a](*N*=27)

1 = strongly disagree to 5 = strongly agree

Table 6 Code distribution in lesson plans

Code	Lesson Plan Applications	
	N	%
CS Principles		
Algorithms	15	68.18
Programming	13	59.09
Abstraction	7	31.82
Creativity	7	31.82
Data	4	18.18
Internet	3	13.64
Impact of CS	2	9.09
Pedagogy		
Collaborative Learning	12	54.55
CS Inquiry	11	50.00
CS Unplugged	6	27.27
Scaffolding - Use	3	13.64
Scaffolding - Modify	3	13.64
Scaffolding - Create	8	36.36

Application of CS Content and Pedagogy in Conceptual Lesson Designs

Participation in a virtual PD institute helped teachers form a conceptual understanding of CS and take steps toward infusing CS in their lesson plans. The construction of lesson plans was guided by the CS standards teachers selected based on relevance to their learning goals and outcomes. As shown on Table 6, teachers were able to articulate their understanding of CS by integrating CS principles and CS pedagogical strategies into their conceptual lesson plans.

CS Principles

Teachers incorporated all seven CS principles into their lesson plans. The most frequently used principles were programming (N=13), algorithms (N=15), abstraction (N=7), and creativity (N=7). Jane and Emma, who taught at the same public elementary school, planned to use Scratch to promote programming and student creativity. Specifically, after having students investigate the various functions of Scratch, they planned to have them program an animation to creatively introduce themselves. In another lesson, Lena utilized algorithms as a way of guiding students through cryptography. Lena's plan placed students in small groups to encrypt and decrypt messages based on codes they created. Kara's lesson utilized algorithms through CS unplugged. In this lesson, students were to write instructions for their classmates related to making a paper plane, which they tested and modified. The least frequently used principles were data (N=4), Internet (N=3), and impacts of computing (N=2). Bianca, for instance, merged social studies with CS to create a lesson plan about the importance of data and how to manipulate them. The goal was for students to clean, filter, and slice a dataset about world happiness on Google Sheets to create visuals based on student-generated questions.

CS Pedagogy

Analysis revealed the integration of four CS pedagogical strategies into lessons – collaborative learning, CS inquiry, scaffolding, and CS unplugged. These strategies are critical for addressing equity concerns (Goode et al., 2014). Collaborative learning was the most commonly used strategy (N=12). Specifically, teachers planned to use strategies such as pair programming or group work to promote collaboration. For instance, Sue's lesson configured learners in pairs to create a skit on Scratch using world languages. Similarly, CS inquiry was also commonly found in lesson plans (N=11). Teachers' lessons frequently encouraged students to experiment with programming through open-ended tasks. Sam's lesson, for example, engaged students in reviewing programming concepts as a way of developing sequences and simple loops to address problems using java script. The goal was for students to engage with a video-game task in order to navigate a hero character through various challenges using efficient strategies.

Results indicated that teachers (N=14) integrated three types of scaffolding in their lesson plans to support CS instruction, reflecting the use-modify-create framework (Lytle et al., 2019). Specifically, some lessons started with *using* pre-existing artifacts, such as an algorithm or CS resource (N=3). Emma, for instance, designed a cybersecurity lesson that engaged students in decoding hieroglyphic messages as a way of understanding how to use code to write and send messages. The next scaffolding phase involved *using* and *modifying* pre-existing artifacts to suit students' needs as designers (N=3). In this phase, Emma's lesson asked students to practice writing and modifying hieroglyphics to produce coded messages. The final scaffolding phase, *create*, moved students away from using and modifying existing artifacts to *creating* new ones (N=8). In this phase, Emma's lesson engaged students in creating strong Internet passwords based on lessons learned from encrypting and decrypting hieroglyphic messages with their peers.

CS unplugged was the least incorporated strategy (N=6). While some lessons were built entirely around CS unplugged, others incorporated unplugged activities as a way to introduce the lesson and relevant concepts without the use of technology. For example, Holly created a programming lesson using Micro:bits. Prior to coding, Holly's lesson had students work in pairs to explain the rules for playing the popular game of rock, paper, scissors following a step-by-step approach (i.e., algorithm). Students were to compare and troubleshoot any missing or confusing steps, replicating the Micro:bit activity. Using pair-programming, students were to subsequently program their devices to play *rock, paper, scissors* and keep track of scores on Google Sheets.

Teachers' Use/Plans to Use New Resources and Strategies from the Virtual PD

In interviews, teachers expanded on classroom implementation of resources and strategies they had learned in the virtual institute. Of the 17 individuals interviewed, 14 could describe an activity that they had implemented or planned to implement with their students or a strategy or resource they had discovered. A difference between the cohorts is relevant here. Interviews for the fall 2020 cohort occurred in spring 2021, allowing more time for actual implementation, while interviews for the summer 2021 cohort occurred shortly after the PD institute. Thus, it is not yet known whether their plans for implementation were carried out.

Most frequently, what had been implemented entailed projects using the Scratch programming language: "[Scratch] was the only thing I actually used in practice. We used Scratch and they had to tell a story about themselves and how the pandemic was affecting them personally." One participant described implementing a CS unplugged "barcode activity for teaching algorithms." This activity, introduced during both institutes, helps teachers understand how the barcode checkout system works. Another teacher described learning strategies for successfully implementing Code HS curricula (although they noted these ideas came from informal discussion, not planned agenda items): "There were some other teachers that were using it, and that sharing of 'how did you make this work? How do you work around that?'"

When summer 2021 participants described concepts they planned to bring into the classroom in 2021-22, the most common were artificial intelligence (AI), programming with Micro:bits, and cybersecurity. There were many mentions of CS curriculum resources including Code.org CS unplugged activities. Two differences were observed in teachers' narratives. First, some participants simply listed topics (e.g., "well, Micro:bits are cool") without providing adequate context related to their implementation, while others laid out concrete and elaborated plans: "we're going to go over the negatives and the positives of AI. We're going to teach AI and [their final project] will be to create a presentation choosing something in real life that could be addressed with AI."

Second, interviews included a mix of teachers who were new to the virtual PD institute and those with experience in prior years (face-to-face). Newcomers were more likely to identify model resources or strategies while veterans usually spoke in terms of enhancing or deepening existing practices, including ideas they had previously gleaned from the institute. For instance, a first-time participant said, "some of the activities with different programs that were shared, I can utilize and change up ways in which I do certain things to make it fun for students. So, utilizing Scratch for instance, to help my students introduce themselves." A veteran planned to "continue using the Explore model that Code.org uses for having students explore some of the program..[and] practices that we're already using as far as pair programming" but to also strengthen another pedagogy "really taking the time to sit down and give the students the opportunity to summarize what they're doing."

Although it was not an explicit question, many teachers raised the pandemic context. Most fall participants described how Covid-19 had constrained their classrooms and their prospects for implementation of ideas from the PD. These challenges included virtual or hybrid environments, shortened or canceled CS class time, and limitations on group work or shared supplies including technology. Summer 2021 interviewees were generally hopeful about the return to in-person learning but were unsure about what protocols would be in place and the implications for CS instruction.

Valuable Elements of Virtual PD

Participants' ratings of the overall institute quality were strong. Everyone who completed a post-survey (N=30) rated the institute "average" or better (M=4.45, SD:0.63), and over half (51.7%) called it "excellent." Further, participants rated almost every aspect of the PD positively (see Table 7). Out of 18 statements, 16 had average ratings of at least a 4.00, meaning an overall level of agreement. Only ratings of facilities and time did not (likely reflecting limitations of the online environment, discussed below). The highest-rated items pertained to relevance and learning.

When teachers identified what they found most valuable about the PD on the post-survey, 6 common themes emerged (see Table 8). Many recurred in interviews, with participants emphasizing the importance of (a) exchanging curriculum ideas, tools, and resources; (b) professional collaboration and networking; and (c) learning CS content. These themes were highly interwoven.

Teachers reported gaining insight into new resources and tools, as well as learning "practical tips" for teaching CS. One wrote, "Every workshop offered a real opportunity to implement what we learned in our classroom." An interviewee pointed to the sessions on cybersecurity and artificial intelligence, saying both facilitators "had very useful resources, so I could just pick it up and I could start teaching it." Another — also referencing cybersecurity—commented: "I didn't know about that, and I'm like, 'oh, I want that!' Because that will get the kids to understand!" Generally,

Table 7 Virtual PD institutes feedback

Statement	Mean	SD
I can use this training to positively impact the achievement of my students.	4.63	0.49
The intent of the workshop is relevant to my professional responsibilities.	4.60	0.50
The facilitators helped me understand how to implement my learning.	4.50	0.51
This workshop will extend my knowledge, skills, and performances.	4.57	0.50
This workshop was tailored to meet my needs as a learner.	4.40	0.67
The facilities were appropriate for the activities.	4.03	0.81
The facilities were conducive to learning.	3.97	0.81
The workshop was supported by effective/appropriate use of technology.	4.47	0.51
New practices were modeled and thoroughly explained.	4.23	0.73
Sufficient time was provided for guided practice and tasks.	3.90	0.80
The facilitators were knowledgeable and helpful.	4.57	0.50
The facilitators were well prepared.	4.43	0.57
The instructional techniques used facilitated my learning.	4.53	0.57
The materials used were accessible and enhanced my learning.	4.50	0.57
The workshop's activities were carefully planned and organized.	4.43	0.57
The workshop's goals and objectives were clearly specified.	4.43	0.57
The workshop included a variety of learning activities relevant to the topic.	4.53	0.57
Time was used efficiently and effectively.	4.47	0.51

N=30

1 = strongly disagree to 5 = strongly agree

Table 8 Most valued aspects of the virtual PD institutes

Post-Survey Theme	Number	Percent
Learning new resources, tools or strategies	13	48.2
Peer support/network	12	44.4
Computer Science content (in general or specific topic)	11	40.7
Facilitators	4	14.8
Chance to model or explore lessons	3	11.1
Breakout rooms	2	7.4

N=27. Some responses mentioned more than one theme

participants found a strong match between the institute's resources and their students' needs.

Sharing resources in this way also nurtured collaboration and professional relationships. One teacher wrote, "Hearing other teachers' ideas and experiences helped to give me a better understanding of ways I could use certain activities/ideas in my classroom." Another explained how the institute helped break down the isolation often experienced by CS educators:

> I love being able to communicate with other educators that are in the same computer science field because often you are kind of alone when you're in these schools. You know, [if you teach] math, you can talk to all the other teachers that are teaching math . . . but technology [cannot do that].

The challenging context of Covid-19 made peer support especially important. For instance, one teacher valued "listening to other educators and how they are teaching during a pandemic."

Breakout rooms allowed teachers to deepen their CS content knowledge and work with those with similar teaching contexts or interests. For example, one high school teacher valued being able to "sit down and actually practice some of the lessons that we're getting ready to do in the AP Computer Science Principles." Parallel sessions also offered variety: "It wasn't like you just had one track to follow. You could bounce around between different rooms, and find out, get more information." This ability to "bounce around" with a single click was one affordance of virtual PD. The virtual format also increased access, especially for geographically far-flung participants, and some found it convenient and productive. One teacher who had attended both formats found online "more focused with fewer distractions. It seems, when we're all in a room together, not that I don't want other people to speak, but sometimes we go off the rails."

Yet the data also reveal the constraints of online PD. Some teachers were simply "sick of [virtual]" or felt that "obviously in person is better." Others found learning CS concepts through a screen challenging and felt that in "face-to-face...you can get better understanding." Finally, virtual PD was by necessity shorter in scope and duration than the face-to-face institute. Some participants expressed the need for more time or a slower pace.

Discussion and Implications

Despite growing attention to PD, we continue to know little about the CS concepts that teachers succeed or struggle with, the pedagogies they engage with, their self-efficacy, and the manner in which they apply PD learning into their practice (Pollock et al., 2017; Rich et al., 2021). Findings from this work indicated that the virtual PD institutes achieved several important successes, despite highly demanding circumstances and the complexities of the pandemic. Specifically, teachers reported increased confidence and preparation to teach CS, particularly regarding programming and algorithms. These findings could be attributed to the increased attention of the virtual institutes on foundational concepts related to programming and algorithms. Building teachers' confidence in foundational CS concepts, such as programming and algorithms, is key to their future success; literature suggests that the most common challenges noted by CS teachers is their own CS subject matter knowledge (Sadik et al., 2020; Sentance & Humphreys, 2018).

Further, analysis of conceptual lesson plans indicated that teachers incorporated CS principles in their curricula. The most widely utilized principles again included programming and algorithms while CS principles related to abstraction and creativity were more limited. These findings are not surprising since these concepts received less attention during the virtual institutes. Further, extensive practice with programming may be needed before teachers feel confident implementing projects that rely on abstraction (Rich et al., 2021). Data, Internet, and impacts of computing received even less attention. However, impacts of computing were typically woven in lessons in the form of pedagogy. In fact, lesson analysis indicated that all teachers included CS pedagogical strategies including collaboration, inquiry, CS unplugged, and scaffolding. Research indicates that such practices are critical for addressing equity concerns, engaging students with real CS learning, and broadening participation in computing (Goode et al., 2014).

Similarly, interview data revealed that several teachers applied content, pedagogy, and technology modeled during the PD in their classrooms following their participation in the virtual institutes. This finding is important given the increased focus on core content area instruction during the pandemic. Some teachers, however, described implementation challenges associated with the pandemic, including time constraints. This finding is consistent with other reports which indicate that teachers temporarily suspended CS instruction during the pandemic (Code.org et al., 2021). Of those teachers who applied PD learning into practice, novice teachers utilized more resources or strategies shared during the institutes while more experienced teachers focused on deepening existing practices. This finding is consistent with research indicating that teachers typically adopt, use, and modify PD content and pedagogy depending on their cultural and social contexts (Grossman et al., 1999). Specific to CS, findings are consistent with prior work indicating different approaches to the uptake of CS content and pedagogy introduced during the PD (e.g., Jocius et al., 2021; Rich et al., 2021).

Finally, results indicated that participant satisfaction remained generally high and all data sources point to the particular value that participants found in the PD community of CS educators. The opportunity to collaborate and share resources and ideas was also important. Some veterans expressed gratitude that the community was able to reconvene during the pandemic. These findings are consistent with prior research which indicates that teachers find value in collaborating and networking with peers (e.g., Goode et al., 2020). Further, the opportunity to learn new CS content was highlighted by participating teachers , both new and returning, indicating the need to differentiate instruction through multiple tracks or break out sessions that allow teachers to continue building their expertise.

Implications for Practice

Findings from this work have implications and can inform the design of effective online CS PD programs aimed at broadening participation in computing. Specifically, two implications for practice are evident.

Differentiate Instruction Findings indicate that participants expressed greater interest for breakout sessions and curriculum resources that closely match their teaching context and level of experience. If participants are welcome to return to the PD for multiple years, it is important that they are able to engage in continuous and sustained learning and exploration of new materials, tools, and resources. Similarly, novice teachers should be allowed the time and space to learn foundational skills and feel confident about their ability to engage diverse students in CS. Research indicates that teachers who are confident in their ability to teach CS are better equipped to implement PD learning into practice and spark their students' interest in CS (Code.org, 2018; Google Inc. & Gallup Inc. 2016). Towards this end, providing choices through parallel sessions that help teachers deepen their understanding of both content and pedagogy is important. In this work, we created different tracks to continue supporting experienced teachers. Further, our help desk in the summer of 2021 was intended to provide a resource for those teachers who needed extra support throughout the PD.

Capitalize on the Strengths of Online Instruction and Strategize about its Constraints Like other scholars (e.g., Jocius et al., 2021), we were initially concerned that transitioning our face-to-face institute into a virtual format would limit participants' opportunities to learn and openly share practices. However, findings indicated that teachers valued several online components, including the flexibility to move from one breakout room to another as well as share and collaborate with other teachers. Further, the online format enabled participants from different regions of the state to attend and enriched the diversity of experiences shared within the group. It was reassuring that almost 50% of teachers reported peer collaboration as their most valued aspect of the virtual institutes. Nonetheless, it is important to note that many participants met each other in prior face-to-face PD offerings and had the opportunity to build community. Therefore, just like the PD reported by Goode et al. (2020), the social presence created throughout the institutes had its foundations in the face-to-face meetings.

Despite the successes, engaging teachers in hands-on programming activities was more challenging online and it frequently took longer to troubleshoot and support novices. Towards this end, just-in time support should be readily available through private meeting rooms that make it easier for novices to voice their questions. Further, online videos and other code visualization tools could be useful for building teacher knowledge of foundational CS concepts (Qian et al., 2018).

Implications for Research

As noted, the background and CS preparation of teachers participating in the virtual institutes varied and our findings did not differentiate based on teacher characteristics. Future research should examine how teacher background characteristics, such as CS knowledge, confidence in teaching CS, prior PD experiences, and initial preparation, may influence their PD experiences. Given that most CS teachers come from different disciplines and backgrounds, identifying how PD experiences influence diverse sets of teachers is key. Similarly, examining whether PD experiences differ based on whether teachers teach standalone CS courses compared to integrating CS in other disciplines will also be important. Furthermore, future research should examine the manner in which teachers apply sound pedagogical strategies in their classrooms, including strategies that connect with the realities of the pandemic. Many of the examples provided by teachers regarding applications of computing to address impacts and community-based issues pertained to the pandemic (e.g., computing applications that facilitate contact tracing). Therefore, to sustain student CS engagement in the immediate future it may be valuable to identify ways in which the teaching of CS can connect to community issues associated with the ongoing Covid-19 pandemic.

Limitations

There are three limitations associated with this work; one related to the PD and two related to the research itself.

PD Limitation We recognize that the virtual institutes provided a marginally sufficient number of hours recommended by best practices in teacher PD. Specifically, research indicates that short PD experiences lasting less than 14 hours show no statistically significant effect on student learning (Darling-Hammond et al., 2009). Further, PD associated with increased student achievement typically spreads out over 6-12 months. While our virtual institutes included approximately 15-20 hours of synchronous, asynchronous, and independent work, these hours were not spread out over time and could not include the year-long follow-up support available during the face-to-face PD institute. Although we documented positive outcomes as a result of teachers' participation in a virtual PD institute, at this time we do not have evidence of sustained implementation of CS in practice. However, it is notable that 21 participants engaged in continuous PD through their participation in our prior face-to-face institutes, in some cases over multiple years.

Research Limitations Participating teachers included a combination of experienced and novice CS teachers. Due to the small sample size our analysis did not disentangle findings based on prior experience or PD participation. Additionally, interviews for summer 2021 participants were completed at the end of the institute. Therefore, it is not known whether participants applied learning from PD in their classrooms following their participation in the summer institute.

Conclusion

As the interest in CS education is growing, there is enormous pressure among K-12 systems to offer CS coursework. However, there is tremendous need for K-12 teachers with the background, preparation, and experience needed to teach CS using pedagogical practices found to broaden participation in computing. While research on face-to-face PD programs with the potential to impact teacher learning and practice has recently emerged, the Covid-19 pandemic necessitated changes in the manner in which such programs were delivered. With the continuous challenges and uncertainty of the pandemic, it becomes essential to examine how online PD approaches can leverage the affordances of technology to prepare CS teachers. In this paper, we examined teacher outcomes as a result of their participation in virtual PD institutes. Findings indicated that teachers benefited from participation in the virtual institutes improving their confidence in teaching CS, application of CS in lesson designs, and implementation in practice immediately following their participation. Further, findings point to implications for the effective design of online PD which helps build teachers' understanding of CS content, pedagogy, and technology.

Funding This work was supported by the National Science Foundation under Award #: 1639649

Declarations

All procedures performed in this study were conducted in accordance with the ethical standards of the institutional review board (IRB) and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards. The methodology and research instruments for this study were approved by the Institutional Review Board at the University of Delaware.

References

Albert, J., Jocius, R., Barnes, T., Joshi, D., Catete, V., Robinson, R., O'Byrne, I., & Andrews, A. (2020). Research-based design recommendations for transitioning a computational thinking integration summer professional development to a virtual format. In R. Ferdig, E. Baumgartner, R. Hartshorne, R. Kaplan-Rakowski, & C. Mouza (Eds.), *Teaching, technology, and teacher education during the covid-19 pandemic: Stories from the field* (pp. 59–64). AACE.

Alvarado, C., Dodds, Z., & Libeskind-Hadas, R. (2012). Increasing women's participation in computing at Harvey Mudd College. *ACM Inroads, 3*(4), 55–64.

Bandura, A. (1977). Self-efficacy: toward a unifying theory of behavioral change. *Psychological Review, 84*, 191–215. https://doi.org/10.1037/0033-295X.84.2.191

Bell, B., & Vahrenhold, J. (2018). CS Unplugged – How is it used, and does it work? In J. J. Bockenhauer, K. Dennis, & W. Unger (Eds.), *Adventures between lower bounds and higher altitudes* (pp. 497–521). Springer.

Borrero, N., Ziauddin, A., & Ahn, A. (2018). Teaching for change: New teachers' experiences with and visions for culturally relevant pedagogy. *Critical Questions in Education, 9*(1), 22–39.

Codding, D., Alkhateeb, B., Mouza, C., & Pollock, L. (2021). From professional development to pedagogy: An examination of computer science teachers' culturally responsive instructional practices. *Journal of Technology and Teacher Education, 29*(4), 497–532.

Code.org Advocacy Coalition (2018). *2018 State of Computer Science Education*.

Code.org, CSTA, & ECEP Alliance (2021). *2021 State of computer science education: Accelerating action through advocacy*. Retrieved from https://advocacy.code.org/stateofcs

College Board. (2017). *AP computer science principles*. College Board.

Computer Science Teachers Association (2017). https://www.csteachers.org/page/standards

Corbin, J., & Strauss, A. (2008). *Basics of qualitative research: Techniques and procedures for developing grounded theory* (3rd ed.). SAGE Publications.

Cresswell, J. W., & Plano Clark, V. L. (2017). *Designing and conducting mixed methods research* (3rd ed.). SAGE Publications.

CS for All (n.d.). https://www.blueprint.cs4all.nyc/what-is-cs/

Darling-Hammond, L., Wei, R.C., Andree, R., Richardson, N., & Orphanos, S. (2009). *Professional learning in the learning profession: A status report on teacher development in the United States and abroad*. Online: https://edpolicy.stanford.edu/sites/default/files/publications/professional-learning-learning-profession-status-reportteacher-development-us-and-abroad.pdf

Dedoose Version 9.0.17, web application for managing, analyzing, and presenting qualitative and mixed method research data (2021). Socio Cultural Research Consultants, LLC www.dedoose.com.

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

Falkner, K., Vivian, R., & Williams, S. A. (2018). An ecosystem approach to teacher professional development within computer science. *Computer Science Education, 28*(4), 303–344. https://doi.org/10.1080/08993408.2018.1522858

Fancsali, C., Mark, J., & DeLyser, L.A. (2020). NYC CS4All: An early look at teacher implementation in one districtwide initiative. In *2020 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, Portland, OR, 2020, pp. 1-8. https://doi.org/10.1109/RESPECT49803.2020.9272418

Ferdig, R., Baumgartner, E., Hartshorne, R., Kaplan-Rakowski, R., & Mouza, C. (Eds.). (2020). *Teaching, technology, and teacher education during the covid-19 pandemic: Stories from the field*. AACE.

Gay, G. (2018). *Culturally responsive teaching: Theory, research, and practice*. Teachers College Press.

Goode, J., Malyn-Smith, J., Peterson, K., & Chapman, G. (2020). Features that support an equity-based professional learning community. *Computing in Science & Engineering, 22*(5), 51–59.

Goode, J., Margolis, J., & Chapman, G. (2014). Curriculum is not enough: The educational theory and research foundation of the exploring computer science professional development model. *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, SIGCSE'14, March 5-8. ACM.

Google Inc. & Gallup Inc. (2016). *Diversity gaps in computer science: Exploring the underrepresentation of girls, Blacks and Hispanics*. Google Inc. & Gallup Inc..

Grossman, P. L., Smagorinsky, P., & Valencia, S. (1999). Appropriating tools for teaching English: A theoretical framework for research on learning to teach. *American Journal of Education, 108*(1), 1–29.

Hodges, C., Moore, S., Lockee, B., Trust, T., & Bond, A. (2020). The difference between emergency remote teaching and online learning. *Educause, Friday, March, 27*, 2020.

Jocius, R., O'Byrne, I., Blanton, M., Albert, J., Joshi, D. & Robinson, R. (2021). Leveraging virtual professional development to build computational thinking literacies in English language arts classrooms. *Contemporary Issues in Technology and Teacher Education, 21*(4). https://citejournal.org/volume-21/issue-4-21/english-language-arts/leveraging-virtual-professional-development-to-build-computational-thinking-literacies-in-english-language-arts-classrooms

Klassen, R. M., & Tze, V. M. C. (2014). Teachers' self-efficacy, personality, and teaching effectiveness: a meta-analysis. *Educational Research Review, 12*, 59–76. https://doi.org/10.1016/j.edurev.2014.06.001

Ladson-Billings, G. (1995). Toward a theory of culturally relevant pedagogy. *American Educational Research Journal, 32*(3), 465–491.

Lee, O. & Campbell, T. (2020). What science and STEM teachers can learn from COVID-19: Harnessing data science and computer science through the convergence of multiple STEM subjects. *Journal of Science Teacher Education, 31*(8), 932–944. https://doi.org/10.1080/1046560X.2020.1814980.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, *2*(1), 32-37.

Lytle, N., Catete, V., Boulden, D., Dong, Y., Houchins, J., Milliken, A., Isvik, A., Bounajum, D., Wiebe, E., & Barnes, T. (2019). Use, modify, create: Comparing computational thinking lesson progressions for STEM classes. *Annual Conference on Innovation and Technology in Computer Science Education*, 395–401.

Madkins, T. C., Howard, N. R., & Freed, N. (2020). Engaging equity pedagogies in computer science learning environments. *Journal of Computer Science Integration, 3*(2), 1–27.

McGarr, O., Goos, M., McInerney, C., Johnson, K., & Flemming, U. (in press). Implementing a professional development framework to assist the roll-out of computer science in second level schools in Ireland. In C. Mouza, A. Leftwich, & A. Yadav (Eds.), *Professional development for in-service teachers: Research and practices in computing education*. Information Age Publishing.

McGee, S., Greenberg, R.I., McGee-Tekula, R., Duck, J., Rasmussen, A.M., Dettori, L., & Reed. D.F. (2019). An examination of the correlation of Exploring Computer Science Course performance and the development of programming expertise. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, SIGCSE '19. ACM.

Menekse, M. (2015). Computer science teacher professional development in the United States: a review of studies published between 2004 and 2014. *Computer Science Education, 25*(4), 325–350.

Mishra, P., & Koehler, M. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. *Teachers College Record, 108*(6), 1017–1054.

Mouza, C., Codding, D., & Pollock, L. (2018). Learning to teach computer science: Professional development design and teacher outcomes. In E. Langran & J. Borup (Eds.), *Proceedings of Society for Information Technology & Teacher Education International Conference* (pp. 1628–1634). AACE.

Mouza, C., Pollock, L., Pusecker, K., Guidry, K., Yeh, C, Harvey, T., & Atlas, J. (2016). Implementation and outcomes on a three-pronged approach to professional development for computer science principles. SIGCSE'16: *Proceedings of 2016 ACM SIGCSE Technical Symposium on Computer Science Education*, pp. 66–71.

Parsons, S. A., Hutchison, A. C., Hall, L. A., Parsons, A. W., Ives, S. T., & Leggett, A. B. (2019). US teachers' perceptions of online professional development. *Teaching and Teacher Education, 19*(1), 33–42.

Pollock, L., Mouza, C., Atlas, J., & Harvey, T. (2015). Field experience in teaching computer science: Course organization and reflections. SIGCSE'15: *Proceedings of 2016 ACM SIGCSE Technical Symposium on Computer Science Education,* pp. 374–379.

Pollock, L., Mouza, C., Czik, A., Little, A., Coffey, D., & Buttram, J. (2017). From professional development to the classroom: Findings from CS K-12 teachers. SIGCSE'17: *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Educatio*n, pp. 477–482.

Pollock, M. (2008). From shallow to deep: Toward a thorough cultural analysis of school achievement patterns. *Anthropology & Education Quarterly, 39*(4), 369–380.

Qian, Y., Hambrusch, S., Yadav, A., & Gretter, S. (2018). Who needs what: Recommendations for designing effective online professional development for computer science teachers. *Journal of Research on Technology in Education, 50*(2), 164–181. https://doi.org/10.1080/15391523.2018.1433565

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.

Sadik, O., Ottenbreit-Leftwich, A., & Brush, T. A. (2020). Secondary computer science teachers' pedagogical needs. *International Journal of Computer Science Education in Schools, 4*(1), 1–21.

Saldaña, J. (2021). *The coding manual for qualitative researchers* (4th ed.). SAGE Publications.

Scott, K. A., Clark, K., Hayes, E., Mruczek, C., & Sheridan, K. (2010). Culturally relevant computing programs: Two examples to inform teacher professional development. In *Society for Information Technology & Teacher Education International Conference* (pp. 1269–1277). Association for the Advancement of Computing in Education (AACE).

Sentance, S., & Humphreys, S. (2018). Understanding professional learning for Computing teachers from the perspective of situated learning. *Computer Science Education, 28*(4), 345–370. https://doi.org/10.1080/08993408.2018.1525233

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends, 60*(6), 565–568.