Early Childhood Education Journal
https://doi.org/10.1007/s10643-025-01961-4

From Teacher Training To Student Growth: Virtual Professional Development Enhances K-2 Computer Science Education

Ghaida S. Alrawashdeh[1] · Abigail J. Bergman[2] · Marina U. Bers[2]

Accepted: 30 June 2025

Abstract

This randomized controlled trial examines the impact of two virtual professional development (PD) models (expert-led vs. peer-led) on teachers' skills and self-efficacy and subsequent effects on students' coding growth. Over two years, 81 teachers and 1,623 students participated in a coding curriculum for K-2. Data collection included coding assessments, surveys, and focus groups. Results show significant improvements in teachers' coding skills, particularly with peer-led models, and a downstream effect on students' coding growth. No significant change was observed in teachers' self-efficacy. Implications include exploring tailored PD for diverse backgrounds and improving PD design to address teachers' concerns, informing effective, sustainable PD strategies for CS education.

Keywords Elementary Education · Mixed Methods · Professional Development · Inservice Education · Online Teacher Learning

Introduction

Integrating coding and computational thinking (CT) into early childhood education is becoming essential for preparing children aged 5–8 for a future driven by digital innovation (Alrawashdeh et al., 2024; Bers, 2017). This shift is underscored by a growing body of research that, while historically focused on older students, now emphasizes the importance of teaching these skills to younger children (Tang et al., 2020; Wakil et al., 2019). In response, educational frameworks worldwide are adapting. In Europe, initiatives like the Informatics for All coalition advocate for integrating computer science (CS) and informatics into schools, guided by the International Society for Technology in Education's Computational Thinking Competencies[1]. Similarly, the United States is incorporating CT into curricula, as evidenced by the Next Generation Science Standards (NGSS Lead States, 2013).

However, integrating CS into early childhood education faces a significant challenge, as few elementary school teachers receive formal training in this area (Bower & Falkner, 2015; Broley et al., 2023; Corradini et al., 2017; Kaya et al., 2019; Mouza et al., 2022). This struggle is often rooted in a lack of pedagogical and content knowledge (Codding et al., 2021; Kong & Wong, 2017; Mouza et al., 2022; Wang et al., 2020), impacting teachers' confidence (Singh, 2018). Professional development (PD) involving active participation is highly effective in enhancing teachers' self-efficacy (TSE), attitudes, and knowledge (Mason & Rich, 2019). While emerging evidence suggests a positive correlation between teachers' competence and confidence and student achievement (Mok & Moore, 2019; Swarnalatha, 2019), research in this area remains limited, highlighting a gap not only in CS education but also in the broader educational

✉ Ghaida S. Alrawashdeh
galrawashdeh@oakland.edu

Abigail J. Bergman
bergmaad@bc.edu

Marina U. Bers
marina.bers@bc.edu

1 Department of Teaching and Learning at School of Education and Human Services, Oakland University, 48309-448 Rochester, MI, USA

2 The Lynch School of Education and Human Development, Boston College, 140 Commonwealth Avenue, Chestnut Hill, MA 02467, USA

1 To learn more visit https://iste.org/standards/computational-thinking-competencies and https://www.informaticsforall.org/#:~:text=Informatics%20for%20All%20is%20a,a%20foundational%20discipline%20in%20schools.

Published online: 20 September 2025

landscape (Copur-Gencturk et al., 2024; Lauermann & ten Hagen, 2021).

Further, compared to other subjects, there has been relatively less focus on PD efforts and studies aimed at introducing CS concepts to elementary classrooms (Mcinerney, 2021). Recent initiatives have predominantly concentrated on preparing pre-service teachers, who are typically more technologically adept, leaving a notable gap in addressing the needs of in-service teachers (Rich, Mason et al., 2021b). Further, in-service teachers face various challenges in attending PD sessions, including high workloads, financial constraints, scheduling challenges, and difficulties organizing substitute classes (Butt et al., 2021; Krille, 2020). Virtual PD emerges as a promising solution, providing flexible and sustainable opportunities, especially during the COVID-19 pandemic (Bragg et al., 2021). One effective approach to virtual PD is the train-the-trainer model. This involves training a small group of educators who then teach their peers, which not only reduces costs but also supports the long-term sustainability of PD programs (Hassler et al., 2018; Campbell, 2014).

Acknowledging these gaps, our research aims to provide insights into the effects of integrating CS in early childhood education. Specifically, our evaluation centers on the effects of two virtual training (expert-led vs. peer-led; train-the-trainer) aimed at enhancing teachers' pedagogical and content knowledge, alongside their self-efficacy in implementing a coding curriculum named Coding As Another Language (CAL; see Appendix for curriculum details). Additionally, we assess the impact of the intervention (training vs. no training) on students' coding skills to inform the creation of effective and universally applicable training programs for foundational CS. This evaluation also incorporates teachers' perceptions of the training and concerns about curriculum implementation, as revealed through focus group discussions. Our investigation employs a randomized controlled trial, with teachers in the treatment condition receiving the training (expert-led vs. peer-led), guided by three research questions:

- **RQ1.** What are the effects of two virtual CS PD models on K-2 teachers' self-efficacy and coding skills?
- **RQ2.** Does improvement in teachers' coding skills as a result of the training impact students' coding proficiency?
- **RQ3.** How do insights from K-2 teachers' focus groups provide explanations for the effects of the PD training on teachers' self-efficacy, coding skills, and their impact on students' coding proficiency?

Literature Review

Computer Science in Early Childhood Education

While CS encompasses a broad range of topics in elementary education, this paper specifically focuses on CT and coding skills. CT involves the thought processes of framing problems and devising solutions that can be executed by a computer, human or machine (2006, 2010, 2019). This encompasses skills like deconstruction, abstraction, pattern recognition, and algorithmic thinking (Hudin, 2023; Montuori et al., 2024; Wing, 2011). Coding, on the other hand, involves creating instructions for a computer-based entity to follow (Egbert et al., 2021).

Integrating both coding and CT in learning environments not only fosters logical reasoning, sequencing, and structuring, but also enhances creativity and critical thinking across subjects (Blake-West et al., 2024; Fraillon et al., 2020; McCormick & Hall, 2022; Papadakis, 2020; Román-González et al., 2017). For instance, emerging evidence shows that these skills positively impact reading and writing, improving students' ability to apply logical reasoning in their written work (Delacruz, 2020; Thompson & Childers, 2021). In response, the revision of CT standards by the Computer Science Teachers Association and International Society for Technology in Education now emphasizes the integration of CT across diverse subject areas, moving beyond its traditional focus within CS education (CSTA, 2017).

Early integration of CT and coding activities also fosters a deep interest in STEM and influences learning trajectories (Aivalogou & Hermans, 2019; Burack et al., 2019; Master et al., 2017; Mihm, 2021; Rosson et al., 2011; Tai et al., 2006). Substantial research shows that this early integration helps reduce disparities in STEM interest and self-efficacy, particularly among historically underrepresented groups (Master et al., 2017; Miller et al., 2018; Witherspoon et al., 2017), highlighting the significance of introducing coding activities early in education (Montuori et al., 2024).

Teacher Self-efficacy

The integration of coding and CT into curricula highlights their growing importance in education (Mills et al., 2024). However, a shortage of K–12 educators proficient in these subjects persists (Mason & Rich, 2019). Key challenges include gaps in subject knowledge among elementary teachers, who often teach CS concepts without formal qualifications (Mcinerney, 2021; Menekse, 2015), limited PD opportunities, and inadequate post-training support (Mason & Rich, 2019; Sentence & Csizmadia, 2017; Yadav et al., 2016). Even trained teachers frequently lack the self-efficacy to teach CS effectively (Rich, Larsen et al., 2021a).

Self-efficacy, rooted in Social Cognitive Theory, denotes an individual's belief in their ability to perform a specific action or achieve a particular outcome in a given situation (Bandura, 1985). This belief, shaped by perceptions of skills, knowledge, and past experiences, can influence academic success, motivation, and resilience (Bandura, 2008). Research underscores the significant impact of self-efficacy on the selection of learning strategies and academic performance across various contexts and age groups, with a pronounced effect observed in STEM subjects (Huang et al., 2022; Nissen & Shemwell, 2016; Skaalvik et al., 2015; Usher et al., 2019).

Teacher self-efficacy refers to a teacher's belief in their ability to achieve desired outcomes, including student engagement and learning (Yeşilyurt et al., 2016). Research across various STEM disciplines indicates that TSE holds equal importance to their actual knowledge and skills, influencing teaching practices and student outcomes (Copur-Gencturk et al., 2024; Ertmer et al., 2012; Tschannen-Moran & Hoy, 2007; Zhou et al., 2020). Rich et al. (2021a, b) emphasized the crucial role of TSE in effective coding instruction, noting that a lack of TSE can hinder elementary teachers' ability to teach coding effectively. In contrast, teachers at the middle and high school levels typically demonstrate higher self-efficacy in CS instruction compared to their elementary counterparts (Schwarzhaupt et al., 2021).

Drawing upon mastery experiences can enhance self-efficacy (Bandura, 1985, 2008; Tschannen-Moran & Hoy, 2007). Zhou et al. (2023), for instance, found a strong overall effect size ($g = 0.64$, $p < .01$) of a STEM PD training on K-12 TSE. Additionally, observing competent demonstrations, particularly from relatable peers, can positively influence self-efficacy, as individuals tend to identify with those who reflect aspects of their own identity (Bandura, 1997). In a study by Kelley et al. (2020), science teachers experienced an increase in self-efficacy after participating in PD sessions held within a community of practice. However, despite its significance, there remains a limited body of research specifically focusing on TSE in CS education (Zheng et al., 2019).

Virtual Professional Development

The shortage of formal CS qualifications among elementary educators (Mcinerney, 2021; Menekse, 2015) has highlighted the need for targeted PD to boost their confidence in teaching computing and engineering concepts (Rich et al., 2017). Well-designed PD programs have proven to increase TSE and improve teaching practices. For example, Mcinerney et al. (2020) observed increased confidence in using CS tools after a six-day training, while Hußner et al. (2024) found that hands-on learning during PD improved both TSE and teaching practices.

However, traditional PD formats, while valuable, often present challenges for in-service teachers, who face obstacles like scheduling conflicts, heavy workloads, and tight budgets (Butt et al., 2021; Krille, 2020). Virtual PD, by contrast, offers a flexible and accessible solution that allows teachers to engage in meaningful professional learning without these barriers (Alrawashdeh et al., 2024). This approach aligns well with the growing prominence of remote learning (Bragg et al., 2021), providing an opportunity for teachers to enhance their knowledge, attitudes, and teaching effectiveness (Chen & Cao, 2022; Lawrence & Ogundolire, 2022).

Synchronous virtual training sessions led by experts offer the advantage of eliminating travel time while enabling teachers from different locations to participate without disrupting their schedules (Alrawashdeh et al., 2024). When peer teachers lead synchronous or asynchronous virtual PD, this model becomes even more flexible. This approach empowers a teacher to guide their colleagues, fostering collaboration while leveraging the expertise within the school or district. As noted by Hassler et al. (2018) and Campbell (2014), this train-the-trainer approach is not only cost-effective but also supports the long-term sustainability of PD programs. However, there is limited research comparing the effectiveness of these virtual models in enhancing coding competence and self-efficacy among elementary educators.

Similarly, the connection between TSE and student achievement remains complex, with some studies suggesting a positive correlation (Mok & Moore, 2019; Swarnalatha, 2019) and others finding no direct link (Jerrim et al., 2023). Interestingly, Li et al. (2022) found that subject-specific PD, such as science-related programs, combined with attendance can positively impact student outcomes. Copur-Gencturk et al. (2024) also found significant gains in student learning after an asynchronous PD program for math teachers. Our study seeks to address these gaps by employing two synchronous virtual training options and examining the downstream effects of teacher PD on student outcomes.

PD Conceptual Framework

The PD program in this study focused on content knowledge (CK) and pedagogical content knowledge (PCK; Loewenberg Ball et al., 2008), targeting teachers' expertise in subject matter, teaching methods, and selecting appropriate CS instructional strategies (i.e., heuristic strategies; Hill & Loewenberg Ball, 2004).

For CK, we emphasized teachers' comprehension of the conceptual foundations of the content and their ability to problem-solve through reasoning and evaluation. PCK drew on Shulman (1986) and Copur-Gencturk and Tolar (2022) work to encompass understanding student challenges, learning patterns, and using instructional tools and methods effectively. While CK and PCK are essential for instructional quality and student performance across subjects (Gess-Newsome et al., 2019; Iserbyt et al., 2017; König et al., 2021; Kulgemeyer & Riese, 2018), novice teachers often prioritize PCK, whereas experienced educators integrate technological knowledge with pedagogy (Zheng et al., 2019). Research on online PD for elementary teachers in China, for instance, reveals disparities in practical knowledge between experts and novices (Sun et al., 2023). This suggests that future PD programs should address the varying needs and interests of teachers at different career stages.

While the participating teachers in this study had an average of 14.5 years of experience, only 19% had post-secondary STEM backgrounds, and 41% had some experience with coding, highlighting the need for CS-specific PD for this population. In Zhou et al.'s study (2020), teachers who engaged in a nine-week hybrid PD program saw a significant increase in both CK and PCK. This underscores the potential benefits of targeted PD in enhancing educators' skills and readiness to teach CS effectively.

As such, our PD program followed Darling-Hammond's (2017) design, incorporating elements to transform teacher knowledge, practices, and beliefs through hands-on coding and pedagogical approaches (Mason & Rich, 2019). The content-specific sessions addressed both student and teacher needs, introducing key subject matter and technology through hands-on activities and facilitator modeling. This involved gaining familiarity with the block-based programming language ScratchJr [2] to bridge the gap between theoretical knowledge and hands-on skills. Teachers were also introduced to the four pedagogical frameworks that form the foundation of the CAL[3] curriculum used in the participating schools.

The training also placed a strong emphasis on enhancing teachers' understanding of common student struggles through reflections, recognizing patterns in their learning, and the ability to select or adapt activities and respond to students' needs based on their level of understanding. Additionally, participating teachers actively engaged in reflective practices concerning lesson implementation in their classrooms, sharing valuable insights about general implementation strategies. The training allowed flexibility for teachers to choose sessions based on their preferences, fostering collaboration and idea exchange among colleagues who attended sessions on the same days. Learning was sustained through weekly check-ins following the conclusion of the PD. Figure A1 in the appendix offers an overview of the topics covered during the training sessions.

The Virtual PD Training Models

As previously mentioned, we delivered the PD program virtually and synchronously, using two distinct models, each informed by key studies. Model 1, expert-led (4 h), focused on building foundational CK and PCK. Model 2, peer-led (up to 6 h), built on the first model by having participants from Model 1 lead the PD the following year after receiving training. This 'train the trainer' model emphasized the power of observing competent demonstrations from relatable peers. Bandura (1997) highlights how this approach can positively influence self-efficacy.

Model 2 expanded on Model 1 by offering additional support and networking opportunities, fostering a strong professional learning community. We hypothesized that allowing time for sharing teaching strategies and resource, as well as networking opportunities would positively impact TSE and other affective states, subsequently influencing student outcomes (Copur-Gencturk et al., 2024; Eells, 2011; Hill & Loewenberg Ball, 2004; Warner et al., 2019). For instance, in Frumin et al.'s study (2018), in-service teachers engaged in an online PD community for STEM-related subjects reported that the provision of networking opportunities, resource sharing, and strategy exchange helped mitigate feelings of isolation.

Model 2 is similar to an Australian PD program where teachers co-taught with computing professionals, gradually gaining independence (Williams et al., 2020). In the first term, the expert delivered content, modeling coding concepts, while in the second term, the teacher led with the professional assisting. Teachers reported increased competence and confidence as they gradually took charge.

Both models align with Darling-Hammond's (2017) principles for effective training: collective participation, active learning, coherence, and content focus (Avalos, 2011; Darling-Hammond et al., 2017; Desimone, 2009; Guskey & Yoon, 2009; Odden & Picus, 2014). Additionally, research on equity-focused PD (e.g., Codding et al., 2021) reinforces the importance of promoting diverse instructional practices, which resonates with the overall PD and curriculum goals. The CAL curriculum actively promotes gender equity and representation of minorities in STEM. This is achieved through featuring diverse role models in the lessons, and highlighting the contributions of women and minorities to STEM advancements. By promoting CS experiences and gender equity, our program reflects research advocating for

[2] ScratchJr has been found to be effective in cultivating both CT and coding proficiencies, even among preschool students (e.g., Louka & Papadakis, 2024).

[3] These are: coding as a playground, coding as another language, coding as a bridge, and coding as a palette of virtues.

these strategies to empower women in STEM (Hinckle et al., 2020).

The training duration of 4–6 h was determined based on stakeholder preferences. This decision was rooted in the recognition that educators often have limited time availability and competing demands on their schedules. By condensing the training into focused sessions, we aimed to maximize participant engagement and retention of key concepts while accommodating educators' busy schedules. Studies evaluating brief-duration PD sessions have shown their effectiveness in achieving learning objectives when appropriately designed (Bonner et al., 2019; Codding et al., 2021; Lauer et al., 2014; Moè, 2021; Simmonds et al., 2021).

Methods

The two synchronous PD training models were implemented over two consecutive years, spanning from 2021 to 2023. The study was conducted in public schools across two New England states in the US, denoted as State A and State B. This study is part of a larger investigation where eligible schools from both states were randomly assigned to either the treatment group (76% of schools), receiving teacher training and implementing the curriculum in the 2021–2022 school year, or the delayed treatment group (24%), receiving training in the subsequent year (2022–2023 school year). Details regarding the randomization and eligibility criteria are outlined in the Appendix.

A comprehensive analysis of the trial's effect on student outcomes, including numeracy and literacy accounting for demographic variables is presented in previous (masked for review) and upcoming publications. The present study focuses solely on the impact of the PD training on teachers' skills and perceptions and how this, in turn, affects students' coding abilities. While the control group teachers received the treatment in the second year, we did not collect data from this group during that time. Instead, we collected data from teachers in the same control schools who had not participate in the first year. These new teachers were allocated to the treatment condition. This approach ensured an exclusive allocation of participants. Approval for the study was obtained from the Institutional Review Board at Boston College [protocol number #23.063.01].

Model 1, conducted in the first year, involved sessions led by an experienced trainer from the research team, comprising two two-hour sessions via Zoom (4 h in total). In the second year, Model 2 was introduced, facilitated by teachers trained as CAL trainers. Model 2 extended the duration up to six hours to allow time for more discussions among attending teachers on teaching strategies, resource sharing, and networking opportunities. CAL trainers collaborated with technical and project coordinators to develop supplementary materials such as lesson slide decks, instructions, and tips. These resources were then distributed to participating teachers by the coordinating teams at each research site. Continuous support was also provided by the research team and a district-level project coordinator throughout the implementation phase. This support included regular email communications and weekly drop-in support sessions. Figure 1 offers an overview of the distinctions between the training models.

Tools

The validated Coding Stages Assessment (CSA; de Ruiter & Bers, 2022) was used to measure both teacher and student knowledge of coding in ScratchJr (reliability: Guttman's $\lambda 6 = 0.94$). This 25-question assessment (details in Appendix) was administered before and after the intervention for both the treatment and control groups. To administer the CSA assessment, independent research assistants, trained separately and blinded to the study design, conducted one-on-one Zoom assessments. An external evaluation group collaborated with school coordinators to ensure unbiased selection of these assistants. Fleiss' Kappa was calculated to assess the inter-rater reliability among the assistants, indicating substantial agreement between the assistants' judgments, $\kappa = 0.656$, $z = 132$, $p < .001$.

TSE in teaching coding was gauged using a pre- and post-training survey (6 items, 5-point Likert scale). The Appendix contains a list of these items and their respective sources. Demographic information (gender, years of teaching experience, coding/STEM background, etc.) was also collected. To gain deeper insights, semi-structured focus groups (45 min) were conducted after training and implementation to explore teachers' perceptions of the training and curriculum (refer to Appendix for the protocol). A member of the research team facilitated these discussions. Finally, lesson logs were used to track fidelity of implementation. Figure 2 summarizes the data collection instruments used for both teachers and students.

Sample

Following an initial eligibility screening of 44 schools, 140 K-2 teachers from 33 schools were invited to join the study by the state department of education. Using stratified random sampling to minimize within-school contamination, ten schools from State A and six from State B were assigned to the treatment group. However, two schools withdrew before implementation began. Of the 114 teachers who initially consented to participate, 10 did not take the baseline coding assessment, and 14 did not complete the endline assessment,

PD Model	% of Participants	Project Year	Total Duration	Facilitator
Model 0 Control	23% (State A = 42%)	Year 1	NA	NA
Model 1 Treatment	46% (State A = 54%)	Year 1	Four hours	Expert in early childhood CS
Model 2 Treatment	31% (State A = 24%)	Year 2	Up to Six hours (first four hours consisted of same activities as models 1 and 2; last 2 hours provided extra time for support, logistics, and networking)	CAL trainer (Year 1 participating teachers who received special training to conduct the PD at a higher education institution)

Fig. 1 An Overview of the Synchronous PD Training Models

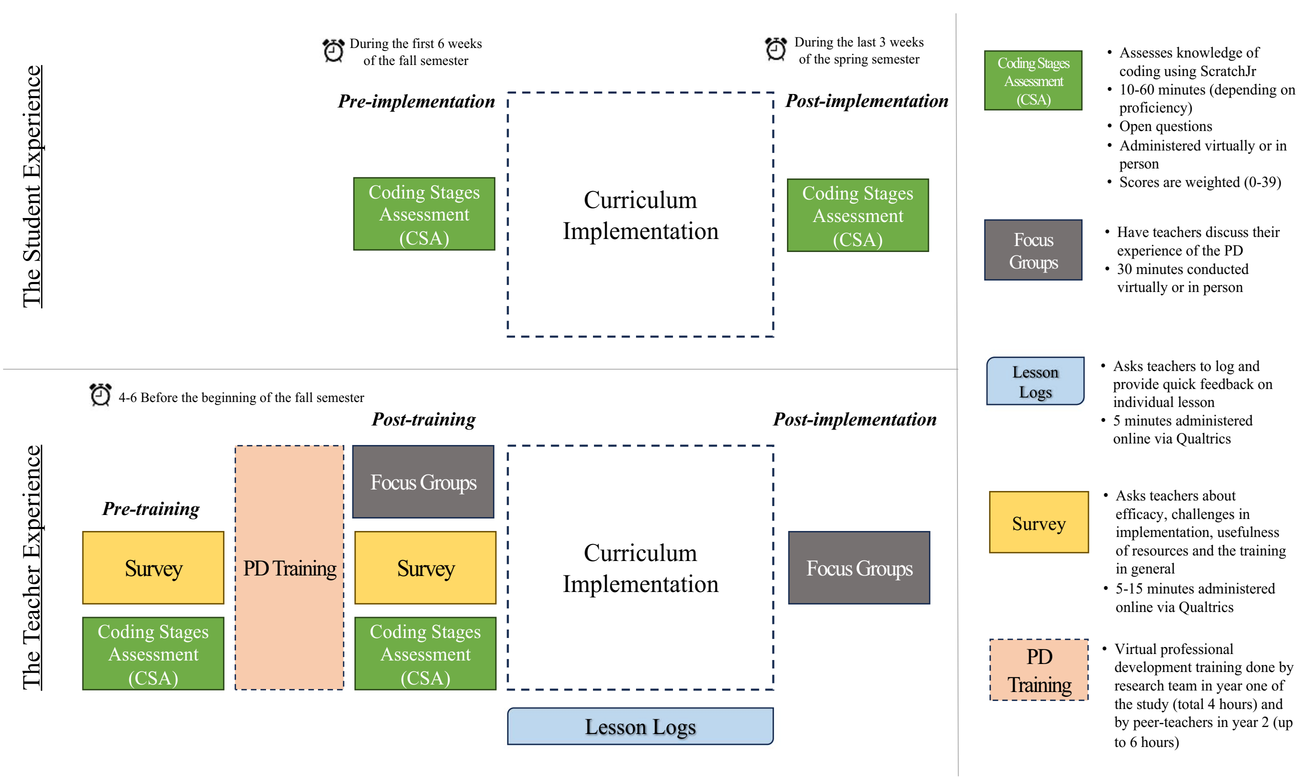

Fig. 2 An Overview of the Larger Project and Data Collection Instruments and Timeline

resulting in a 13.5% attrition rate. This attrition was primarily due to teacher turnover, COVID-related disruptions, and time constraints. Additionally, in some schools, curriculum implementation was taken over by specialists, either technology ($n=5$) or inclusion ($n=4$), who assisted the trained teachers. Analysis of baseline coding scores revealed no significant difference between teachers who completed the study and those who withdrew, $t(111)=0.78$, $p=.436$.

Of the 2,196 students who consented to participate, 66 did not complete the baseline assessment, and 434 students (whose teachers dropped out) did not complete the endline assessment, resulting in an attrition rate of 20.6%. Baseline coding scores showed no significant difference between students who remained and those who withdrew, $t(2110)=-0.84$, $p=.400$.

Table 1 Frequency table for teachers ($n=81$) and Students ($n=1{,}623$)

		Participants			
		Teachers		Students	
Variable Level		n	%	n	%
Project Year	Year 1	42	52	891	55
	Year 2	39	48	733	45
Condition	Treatment	62	76	1224	75
	Control	19	24	399	25
Site	State A	34	42	625	39
	State B	47	58	998	61
Gender	Woman	76	94	740	51
	Man	2	2	703	49
	Unreported	3	4	0	0
Race/Ethnicity	White	57	70	839	58
	Hispanic	6	7	309	21
	Black or African American	8	10	143	10
	Mixed	4	5	79	5
	Asian	3	4	65	5
	Native American/ Alaska Native	–	–	8	1
	Unreported	3	4	–	–
Teacher PD Model	Model 1	38	47	791	49
	Model 2	31	38	569	35
	Model 3	12	15	263	16
Teacher Post-secondary STEM Background	No	63	81	–	–
	Yes	15	19	–	–
Teacher Prior Coding Experience	No	46	59	–	–
	Yes	32	41	–	–
Student Grade	K	–	–	495	30
	1	–	–	470	29
	2	–	–	659	41
Student SES (Low)	No	–	–	993	69
	Yes	–	–	450	31
Student ELL	No	–	–	1226	85
	Yes	–	–	217	15
Student LEP	No	–	–	1201	83
	Yes	–	–	242	17

Given the focus of this paper, the analytical sample includes data from teachers who implemented the curriculum themselves ($n=81$), and their students ($n=1{,}623$). As a result, the analytical sample is less balanced than originally intended. Among these 81 teachers, 5 from State A and 11 from State B participated in focus groups post-training, while 4 from State A and 13 from State B participated post-implementation. Detailed information on the sampling process, allocation to treatment conditions, and handling of missing data is available in the appendix.

The majority of the 81 teachers were identified as women (94%), White (70%), and from State B (58%), as detailed in Table 1. Similarly, student demographics, also shown in Table 1, revealed a White majority (58%) from State B (62%) with high socioeconomic backgrounds (69% not qualified for free/reduced lunch). 15% were English language learners (ELL) and had individualized educational plans (IEP). Importantly, students were unaware of their assigned condition (treatment or control).

Analysis

To address the research questions, we conducted a multi-step analysis. Factor analysis with Promax rotation validated the self-efficacy construct. Assumptions, analysis and results are available in the Appendix. We then normalized the coding and TSE composite scores to a 0-100 scale to facilitate meaningful comparisons and analyses. To quantify the improvement in coding skills and self-efficacy from baseline to endline, we calculated the score difference for each participant by subtracting the pre-assessment score from the post-assessment score. We refer to the students' coding improvement as SCI, teachers' coding improvement as TCI, and teachers' self-efficacy improvement as TSE.

Following this data preparation, we ran a path model to explore the effects of the PD training on TCI and TSE (RQ1). We also investigated the relationship between TCI and SCI (RQ2). Mediation tests were conducted to evaluate these effects, considering both full and partial mediation, with a significance level (alpha value) set at 0.05 (Gunzler et al., 2013; Preacher & Hayes, 2004; Zhao et al., 2010). To account for initial abilities, we included baseline scores in the model as covariates. For a visual representation of the path model, refer to Fig. 4 and see appendix for assumptions.

To determine the standard errors for the path parameter estimates, we conducted Bootstrapping with 100 iterations, obtaining comparable results in sensitivity analysis with Maximum Likelihood estimation. To assess the reliability of the path model, we considered the $N{:}q$ ratio. Our analysis,

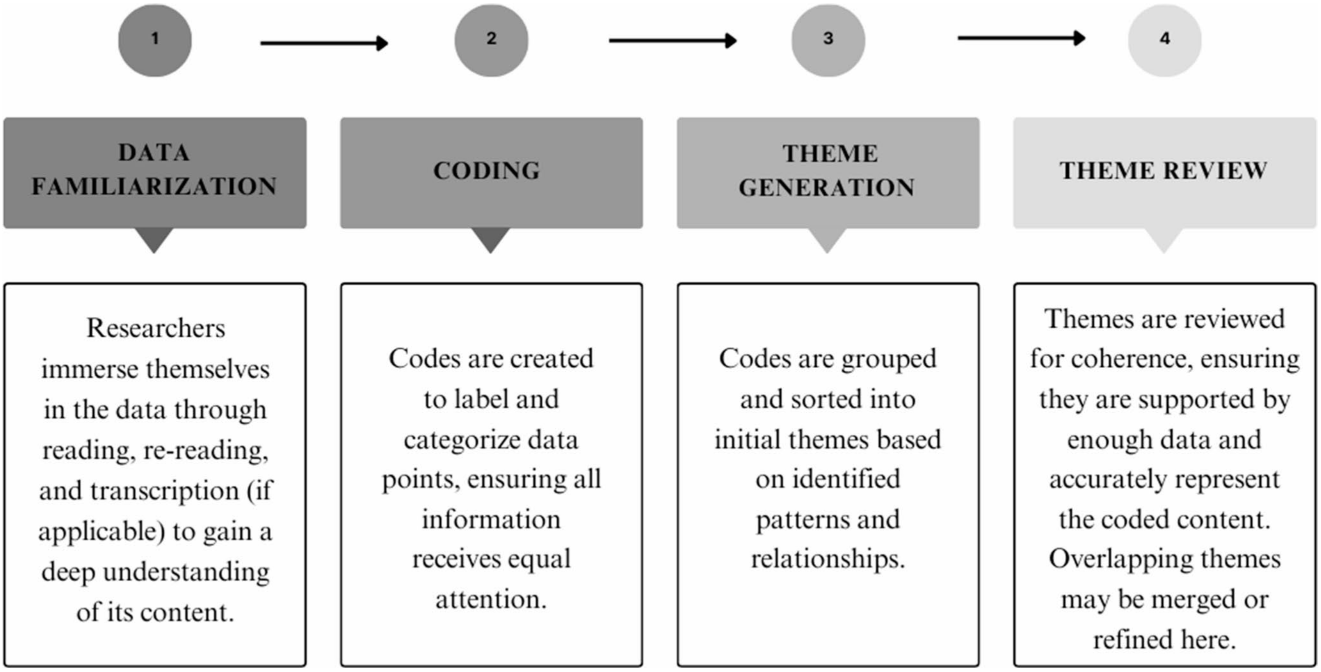

Fig. 3 Reflexive Thematic Analysis Steps for the Focus Groups

Table 2 Descriptive statistics for teachers ($n=83$) and students' ($n=1623$) Coding and TSE scores by condition and time

Participant	Variable	Condition	Baseline			Endline		
			M	*SD*	SE_M	*M*	*SD*	SE_M
Teacher	Coding	Treatment	32.56	26.70	3.39	57.54	30.87	3.92
		Control	26.85	29.53	6.78	42.96	37.58	8.62
Teacher	TSE	Treatment	43.55	27.53	3.50	72.45	14.10	1.79
		Control	45.22	27.21	6.24	72.37	14.91	3.42
Student	Coding	Treatment	21.35	15.30	0.44	59.36	26.63	0.76
		Control	29.08	17.48	0.88	59.67	25.23	1.26

M mean; *SD* Standard deviation; SE_M Standard Error of the Mean

with a participant-to-item ratio of approximately 135 to 1, involving a sample size of 1623 and 12 variables, met the recommended *N*:*q* ratio (Kline, 2016; Schreiber et al., 2006). While the Chi-square statistic is a commonly used measure for path model fit, it is sensitive to sample size (Hooper et al., 2008). Therefore, additional fit indices were considered alongside the Chi-square test, including RMSEA, CFI, and SRMR. Data analysis was performed in R Statistical Software (Version 2023.06.1 + 524).

To gain deeper insights from the focus group discussions regarding the PD training (RQ3), we employed a Reflexive Thematic Analysis (RTA) approach (Braun & Clarke, 2006) after transcribing the interviews. RTA's strength lies in its theoretical flexibility, allowing a wider range of themes to emerge organically, aligning with our study's goal of generating themes for focused discussion of the quantitative results. This iterative process involved systematically reviewing transcripts to identify patterns and insights relevant to our research questions.

We used an Excel spreadsheet to organize and categorize the data for efficient analysis. Each transcript was carefully reviewed multiple times to ensure comprehensive understanding of the participants' perspectives and experiences. Through this iterative process (see Fig. 3), emergent themes were identified and refined. The main author conducted this analysis, with support from an external evaluation group involved in the broader project.

Results

In examining teachers' mean coding scores at baseline (Table 2), no significant differences were observed between the Treatment ($M=32.56$, $SD=26.70$) and Control ($M=26.85$, $SD=29.53$) conditions, $t(79)=0.80$, $p=.428$. Similarly, for TSE, no significant differences at baseline were observed between the Treatment ($M=43.55$, $SD=27.53$) and Control ($M=45.22$, $SD=27.21$) conditions, $t(79) = -0.23$, $p=.817$.

However, significant differences were found at baseline between the mean scores of students assigned to the Treatment and the control conditions, $t(609.03) = -7.90$, $p < .001$, with the students in the Control condition having higher initial scores (Table 2).

The path analysis results are presented in Fig. 4; Table 3. The Chi-square goodness of fit test was significant, $\chi^2(3) = 11.37$, $p = .010$. However, when considering other fit indices, the model appeared to provide a good fit, as shown in Table 3 (Hooper et al., 2008).

Effect of the Intervention on Teachers' Outcomes (RQ1)

The path analysis results revealed that the PD model had a significant positive effect on TCI, $B = 7.10$, $z = 6.84$, $p < .001$. Both expert-led and peer-led PD models effectively improved teachers' coding skills, with the effect size being moderate. The PD model, however, did not significantly impact TSE, $B = -0.91$, $z = -1.08$, $p = .279$, indicating no meaningful change in teachers' self-efficacy across training types.

Further analyses using multiple regression identified several demographic factors as significant predictors of TCI, even when controlling for the intervention. TCI significant predictors included the site, grade level taught, race/ethnicity, and STEM background. Specifically, teachers who taught lower grades, were from State B, or had a STEM background demonstrated greater coding improvement. Conversely, teachers identified as Black/African American or mixed race showed lower improvement compared to White teachers. Complete results are presented in the appendix.

Effect of Teachers' Coding Skills on Students' Coding Skills (RQ2)

The path analysis indicated that TCI had a small but significant positive effect on SCI, $B = 0.05$, $z = 2.59$, $p = .010$. The PD model directly impacted SCI, $B = 2.77$, $z = 3.05$, $p = .002$. Since the direct effect between PD Model and SCI was significant, full mediation by TCI did not occur. Partial mediation was examined using the indirect and total effects of TCI on the relationship between Model and SCI (Gunzler et al., 2013; Zhao et al., 2010; Preacher & Hayes, 2004). The indirect effect of TCI on the relationship of SCI regressed on Model was significant, $B = 0.39$, $z = 2.42$, $p = .016$. The total effect of Model on SCI was also significant, $B = 3.16$,

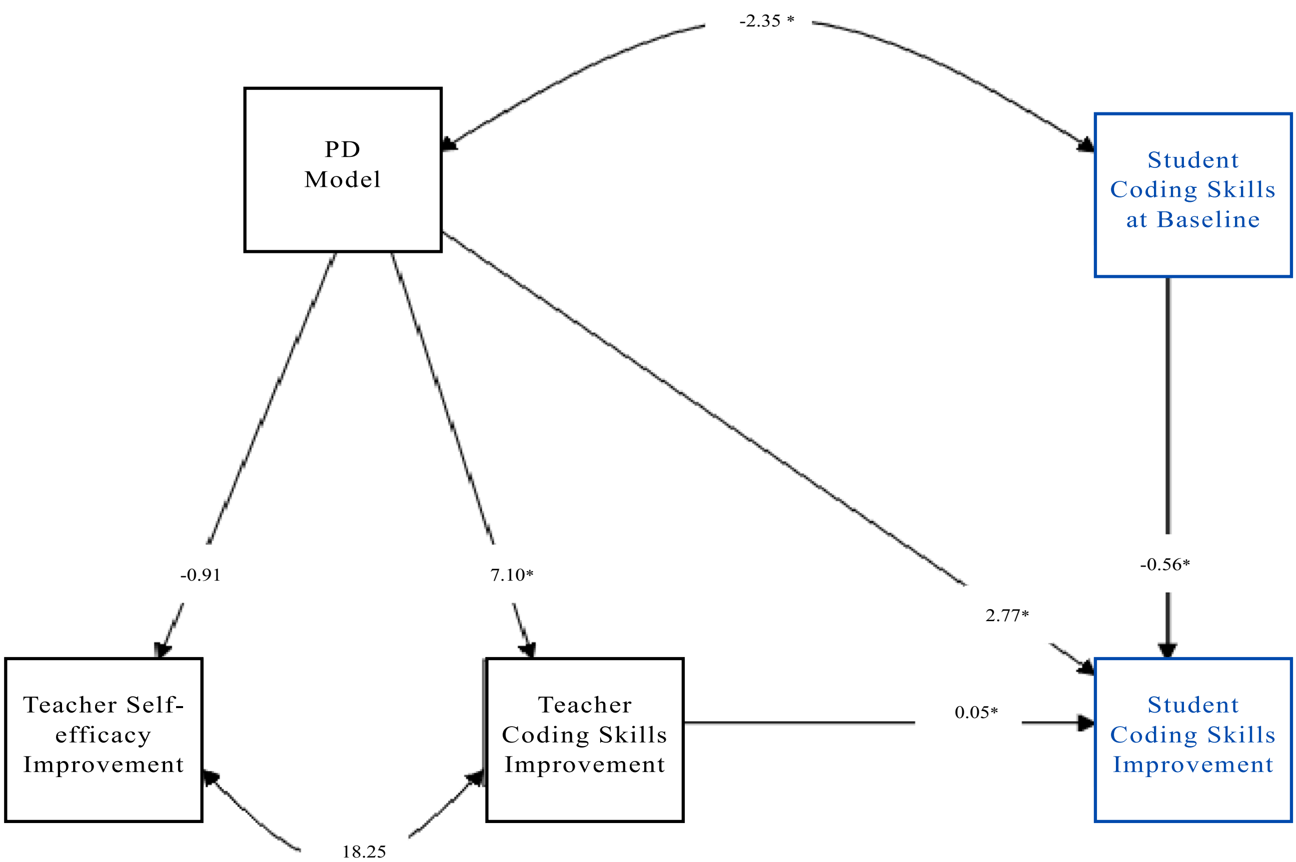

Fig. 4 Results of the Path Analysis Model. Note: To prevent overcrowding error terms are not presented in the figure. Refer to Table 3 for the complete results

Table 3 Unstandardized loadings (Standard Errors), Standardized loadings, and Significance Levels for Each Parameter in the path analysis model (n=1623)

Parameter Estimate	Unstandardized	Standardized	p
Regression: PD Model (Model 0–3)			
TSE	−0.91(0.84)	−0.03	0.279
TCI	7.10(1.04)	0.17	<0.001
SCI	2.77(0.91)	0.07	0.002
Regression: TCI			
SCI	0.05(0.02)	0.06	0.010
Regression: Student Baseline Coding			
SCI	−0.56(0.04)	−0.33	<0.001
Mediation	0.39(0.16)	0.01	0.016
Indirect Effect of SCI on Model by TCI	3.16(0.90)	0.08	<0.001
Total Effect of SCI on Model			
Covariances			
TSE and TCI	18.25(17.87)	0.03	0.307
Model and Baseline	−2.35(0.29)	−0.20	<0.001
Errors			
Error in Model			
Error in TSE	0.51(0.02)	1.00	<0.001
Error in TCI	581.41(20.41)	1.00	<0.001
Error in SCI	891.22(31.29)	0.97	<0.001
Error in Baseline	640.10(22.47)	0.87	<0.001

Note. $\chi^2(3)=11.37$, p=.010; NFI=0.97; TLI=0.92; CFI=0.98; RMSEA=0.04, 90% CI = [0.02, 0.07]; SRMR=0.02.

z=3.51, p<.001. Since the indirect and total effects were significant, partial mediation was supported by TCI. As for students' baseline coding scores, they had a significant negative relationship with their coding improvement, B = −0.56, z = −14.11, p<.001, indicating that students with higher initial coding skills showed less improvement, likely due to a ceiling effect.

Teachers Perceptions of the PD (RQ3)

Analysis of focus group interviews revealed five interconnected themes that capture teachers' experiences with the PD program and its classroom implementation: teacher confidence, teacher concerns, challenges in implementation, hands-on training, and student success. These themes reflect both the strengths of the training and persistent challenges, as summarized in Table 4 and elaborated below.

Many teachers reported a significant increase in their confidence regarding teaching coding. Despite initial apprehension, they became more comfortable as they progressed through the curriculum. One teacher shared, "I was a little nervous this year coming into it, but so far I've been really, really impressed. They've given us the handheld directions of how to do everything. If you're really not comfortable or confident they can walk you through it even more or go to all the office hours." Teachers reported that the training not only increased their skills but also fueled their enthusiasm for teaching coding. Many described feeling newly excited and motivated to bring coding into their classrooms. One teacher shared,

> I had no experience with it [coding through ScratchJr], and came to this experience feeling less confident about my ability to learn it myself, let alone instruct my kindergartens, and I not only learned it, but I was able to teach my class to do so. And I'm also leaving this project with just a much more positive image of our experience of technology and specifically coding.

Hands-on, interactive training emerged as a key strength of the PD program. Teachers consistently cited the opportunity to actively engage with the curriculum as critical for their learning. As one teacher described, "It was a really fun training and I love the hands-on piece of it. I got really into it and the training allowed me to figure out where I was strong and where I might need to focus a little bit more on." This practical, engaging approach helped teachers feel more prepared for classroom implementation. Another teacher emphasized, "I love the hands-on aspect of the training. It really engaged me and helped me identify my strengths and areas needing improvement." Another teacher echoed this sentiment,

> I think they provided us with a lot of opportunities for hands-on activities and sharing our work, plus some homework assignments. It felt like comprehensive training with chances to learn from each other through sharing. The hands-on experience was incredibly beneficial, and learning from others was invaluable.

Other teachers highlighted the value of learning from peers who had already navigated potential challenges, emphasizing, "I like the fact that there were teachers that did it already, so they already knew some of the pitfalls and would provide examples or explicit directions."

Teachers also observed positive changes in student motivation, engagement and learning outcomes. Students demonstrated enhanced CT skills, including the ability to debug code and follow logical sequences of instructions. One teacher noted, "Even some of those computational thinking skills of debugging without getting super frustrated, for example, how do you work through a problem or a line of code, how do you follow step-by-step instructions or know the importance of order. All of those things became evident." Others highlighted the inclusive nature of coding

Table 4 Themes Identified from teacher focus group Interviews, along with example codes and quotes

Theme	Description	Example Quote	Example Code
Teacher Confidence	Teachers reported increased self-efficacy in teaching coding, especially with hands-on support and guidance.	"*I was a little nervous this year coming into it, but so far I've been really, really impressed. They've given us the handheld directions of how to do everything. If you're really not comfortable or confident they can walk you through it even more or go to all the office hours*."	Confidence Boost, Increased Self-Efficacy, Supportive Training
Teacher Concerns	Teachers expressed worries about integrating coding into already packed curricula and potential time constraints.	"*I'm nervous about the scheduling of the curriculum. My school just has a jam-packed schedule."*	Scheduling Issues, Curriculum Integration Concerns, Time Constraints
Hands-on Training	Teachers valued interactive, practical training that prepared them for implementation.	*"It was a really fun training and I love the hands-on piece of it. I got really into it and the training allowed me to figure out where I was strong and where I might need to focus a little bit more on."*	Interactive Training, Practical Learning, Engagement in Training
Challenges in Implementation	Technical/financial barriers hindered implementation.	*"[We had] tech issues related to the devices, not necessarily to the program itself."*	Technical Issues, Device Problems, Financial Barriers
Student Success	Teachers observed improved engagement and learning, particularly in computational thinking and problem-solving skills.	*"Even some of those computational thinking skills of debugging without getting super frustrated, for example, how do you work through a problem or a line of code, how do you follow step by step instructions or know the importance of order. All of those things became evident."*	Student Engagement, Growth, CT, Problem-Solving Skills, Learning Outcomes

education, noting its impact on students who were typically less engaged. As one teacher shared, "I'm amazed at how well my students are able to just create a line of code to get their character to do a set series of tasks. There is tremendous evidence of this impact." Another teacher reflected, "Yeah, I would say first off just talk up like the student engagement piece of it. It is so high and really getting some of those students that aren't traditionally engaged." Additionally, teachers noted the positive impact of the iterative process of coding on student learning. One teacher explained,

> All my kids, learn so differently. […] They were all able to learn. No one ever really felt frustrated or like a failure. Because we learned through this program like you do fail. You need to debug, you need to rethink, and you should go back. So, I really thought that all of my learners were able to complete this with success.

Despite these successes, teachers noted challenges in curriculum implementation. Chief among these were technical barriers and concerns about curriculum integration. Teachers encountered device-related issues that hindered smooth delivery of coding lessons. One participant explained, "[We had] tech issues related to the devices, not necessarily to the program itself." Financial limitations and access to reliable technology were also noted as constraints.

Additionally, teachers expressed concerns about scheduling and curriculum integration. Fitting coding lessons into already packed school days posed a significant challenge. One teacher stated, "I'm nervous about the scheduling of the curriculum. My school just has a jam-packed schedule." Some teachers also voiced nervousness about supporting multiple students simultaneously, particularly when using platforms like ScratchJr. As one teacher candidly shared, "[I felt nervous about the ability] to support multiple students using this [ScratchJr] at the same time."

While teachers expressed enthusiasm for the curriculum, several also shared feelings of nervousness about implementation. Their excitement was sometimes tempered by concerns about their own comfort level with the new material. As one teacher candidly expressed:

> I haven't even gone through one lesson as myself, and now I'm imagining how I'm going to teach this to students. I feel fine about not doing a deep preparation with material with which I'm already comfortable, if I've taught this lesson 10 times. But something that's brand new, I really like to have it.

Teachers further emphasized the need for additional scaffolding, particularly for younger learners. Some noted that certain parts of the curriculum felt more advanced than what students in early grades could manage independently. One teacher observed:

> My students really liked the curriculum, but sometimes there would be an assumption that all of my kindergarten students are able to read and write; that's not the case in kindergarten. There was a lot of scaffolding that I had to do before presenting them with the lesson. I had to walk them through some of those things just because they're not there yet.

Discussion

The present study explored the impact of a virtual PD training designed to introduce CS concepts to K-2 teachers, focusing on whether its effect extended beyond technical proficiency to enhance TSE. This work aimed to address the persistent gap in CS pedagogical and content knowledge among elementary in-service educators (Codding et al., 2021; Kong & Wong, 2017; Mouza et al., 2022; Wang et al., 2020). The study also examined whether a peer-led virtual training model could produce outcomes comparable to those of an expert-led model.

The findings indicated that, although the PD program did not significantly improve teachers' self-efficacy, it led to significant gains in their coding skills. Both groups began with comparable levels of coding proficiency, yet the treatment group showed significant improvement following the intervention, underscoring the program's effectiveness in building technical skills. This result is particularly significant considering that many elementary teachers lack formal CS training (Mcinerney, 2021; Menekse, 2015). It is consistent with Zhou et al.'s (2020) study, in which a nine-week hybrid PD program similarly strengthened teachers' CK and PCK, further supporting the promise of targeted PD efforts to enhance educators' preparedness to teach CS.

The effectiveness of the PD program appeared to vary by state and grade level. Teachers in State A and those working with Kindergarten and First Grade students exhibited greater improvements in coding skills compared to teachers in State B and those teaching Grade 2. These differences suggest that factors such as local support, classroom contexts, and levels of student engagement may shape the outcomes of PD initiatives.

In terms of demographic differences, teachers identifying as Asian or Hispanic/Latino demonstrated gains in coding skills comparable to their White colleagues. However, teachers identifying as Black/African American, Mixed/Other, or Men showed smaller improvements. These disparities highlight persistent barriers that the current PD model may not fully address. Extending the training period could provide additional opportunities for skill development and reflection, potentially helping to mitigate these gaps.

The study did not find a statistically significant impact of the intervention on TSE, diverging from Zhou et al.'s (2023) meta-analysis, which reported notable TSE gains among teachers receiving similar training. Still, teachers in the treatment group linked their increased confidence in teaching CS concepts to the program's hands-on approach, consistent with literature emphasizing the value of practical skill development in PD programs (Fenton et al., 2019; Hill et al., 2020; Hußner et al., 2024).

These findings also align with Schwarzhaupt et al.'s (2021) work, which found that teaching higher grade levels is associated with stronger TSE. While coding skills can be strengthened through tangible, hands-on practice, CT development demands engagement with more abstract thinking, necessitating intentional scaffolding and extended practice. Wang et al. (2020) similarly stressed the importance of scaffolding to support young children in building CT practices. This need for additional support was reflected in teachers' feedback in the current study, with several commenting that aspects of the curriculum seemed too advanced for their younger students to navigate independently.

The full 24-session (18-hour) curriculum was designed to gradually build problem-solving skills through activities targeting deconstruction, abstraction, pattern recognition, and algorithms. However, implementation in practice varied significantly, with teachers completing an average of only 12 lessons ($SD = 7.34$). Limited device availability, teacher absences due to medical or maternity leave, and the ongoing challenges of the COVID-19 pandemic likely contributed to this shortened implementation. Teachers cited difficulties fitting the curriculum into packed schedules and managing tech issues unrelated to the curriculum or the coding program itself. A site-level administrator in State B observed the same challenge, adding, "I think the challenge continues to be time to actually do the implementation." These barriers may partially explain differences observed between State A and State B.

The analysis further revealed that the peer-led model (Model 2) was more effective than the expert-led model (Model 1) in enhancing coding skills for both teachers and students. This finding aligns with Bandura's (1997) social learning theory, which highlights the power of observing relatable peers in boosting self-efficacy. While both PD models offered implementation support, the peer-led model's longer duration and collaborative features likely facilitated more effective learning exchanges among teachers. This collaborative environment resonates with Sentence and

Csizmadia's (2017) call for comprehensive support systems in teacher training.

Model 2 also reflects broader calls for sustainable and cost-effective PD approaches (Hassler et al., 2018; Campbell, 2014). By combining peer leadership with strong facilitation, it offers a scalable approach to expanding CS education without compromising quality. Focus group discussions reinforced this, as teachers valued the clear, "explicit directions" and the opportunity to learn from peers who had firsthand experience with common challenges.

At the same time, teachers expressed hesitation about fitting the curriculum into existing school schedules. Several raised concerns that without protected time, they would struggle to devote the "attention that [the curriculum] deserves and requires." Addressing these concerns will require not only continued teacher development but also broader policy changes that elevate the role of CS education in the elementary curriculum. Ensuring that curriculum demands are aligned with school structures will be essential to maximizing the impact of PD efforts.

The study's second aim examined how improvements in teacher coding skills affect student outcomes. Path analysis results showed that increases in teacher coding skills significantly predicted gains in student coding proficiency. This finding builds on work by Copur-Gencturk et al. (2024), who linked asynchronous PD to higher student math achievement, and aligns with studies by Mok and Moore (2019) and Swarnalatha (2019) pointing to a positive correlation between teacher coding skills to student performance. These findings, along with growing evidence on the lasting benefits of early CS exposure (Aivalogou & Hermans, 2019; Rosson et al., 2011; Tai et al., 2006), point to the critical importance of sustained teacher development in advancing student learning.

Focus group findings reinforced the quantitative results. Teachers reported high levels of student engagement, particularly among students who often struggled in other academic areas. Many were surprised by how quickly students picked up coding concepts and noted that the iterative process of coding fostered perseverance and problem-solving skills. Teachers also highlighted the inclusive nature of coding education, emphasizing its ability to engage and motivate a diverse range of learners.

Conclusions

This study provides clear evidence that targeted professional development can meaningfully strengthen elementary teachers' CS proficiency, which in turn supports improvements in student performance. By demonstrating a positive relationship between teachers' skills and student outcomes, our findings affirm the critical role of teacher preparation in expanding equitable access to foundational CS learning. Importantly, the success of the 'train the trainer' model highlights a promising pathway for scaling PD efforts sustainably. However, the challenges identified, particularly related to teacher self-efficacy and scheduling constraints, point to necessary areas for improvement to ensure broad and lasting impact.

These results contribute important insights for both practice and policy. Supporting elementary teachers in CS instruction demands not only the development of technical skills but also sustained opportunities for practice and reflection. Short-term PD offerings, while helpful, may not be sufficient; extending the duration of PD programs could deepen learning and build teacher confidence. Additionally, addressing systemic barriers, such as limited instructional time for CS, will require coordinated policy interventions that elevate the status of CS education at the elementary level.

Several limitations should be noted. The relatively brief PD intervention period limits our ability to draw conclusions about long-term teacher and student outcomes. Variability in teachers' prior exposure to CS concepts may also have influenced engagement and growth. Future studies should track outcomes over longer periods, employ more rigorous assessments of contextual factors, and explore strategies to differentiate PD for teachers with varying levels of experience.

Future research should also examine mediators and moderators of PD effectiveness beyond those explored here, such as teacher motivation, school leadership support, and students' prior experience with technology. Moreover, the long-term scalability and sustainability of 'train the trainer' models must be evaluated systematically, considering both teacher outcomes and broader school- and district-level impacts. Research that centers teachers' voices and diverse backgrounds will be especially important for designing PD that meets the needs of all learners.

Overall, this study advances understanding of how professional development can serve as a lever for improving CS education at the elementary level. While the results are encouraging, sustained investment in research, practice, and policy will be essential to fully realize the potential of CS for all students.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10643-025-01961-4.

Acknowledgements We wish to thank the participating schools, teachers, and students for their invaluable contributions. We also thank the Department of Education for providing the grant that made this study possible. Finally, our appreciation goes to the external evaluation group, Shaffer, for their insightful input during the analysis of the focus group discussions and to the reviewers and editors of this journal for their time and recommendations.

Author Contributions Ghaida S. Alrawashdeh: Conceptualization, Investigation, Formal analysis, Visualization, Writing—original draft preparation, Writing—review and editing. Abigail Bergman: Writing— preparation of the original draft of the literature review, Validation. Marina U. Bers: Data curation, Methodology, Writing—review and editing.

Funding This paper is based on work supported by the Department of Education under Grant Number U411C220202. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Education.

Data Availability De-identified data that support the findings of this study are available from the authors upon reasonable request and with the permission of the participating districts.

Declarations

Competing Interests The authors have no relevant financial or non-financial interests to disclose.

Ethical Approval Approval for the study was obtained from the Institutional Review Board at Boston College [protocol number #23.063.01].

References

Aivalogou, E., & Hermans, F. (2019). Early Programming Education and Career Orientation: The Effects of Gender, Self-Efficacy, Motivation and Stereotypes. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 679–685. https://doi.org/10.1145/3287324.3287358

Alrawashdeh, G. S., Nadler, E., & Bers, M. U. (2024). Virtual professional development enhances elementary teacher' coding skills and self-efficacy: A comparison of three models. In S. Papadakis & M. Kalogiannakis (Eds.), *New Approaches in Mobile Learning for Early Childhood Education* (pp. 14–137). IGI Global. https://doi.org/10.4018/979-8-3693-2377-9.ch005

Avalos, B. (2011). Teacher professional development in teaching and teacher education over ten years. *Teaching and Teacher Education*, *27*(1), 10–20. https://doi.org/10.1016/j.tate.2010.08.007

Bandura, A. (1985). *Social Foundations of Thought and Action: A Social Cognitive Theory*. Prentice-Hall.

Bandura, A. (1997). Behavior theory and the models of man (1974). In J. M. Notterman (Ed.), *The evolution of psychology: Fifty years of the American Psychologist* (pp. 154–172). American Psychological Association.

Bandura, A. (2008). An agentic perspective on positive psychology. In S. J. Lopez (Ed.), *Positive psychology: Exploring the best in people* (Vol. 1, pp. 167–197). Praeger Publishers/Greenwood Publishing Group.

Bers, M. U. (2017). *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom (1st edition)*. Routledge.

Blake-West, J., Alrawashdeh, G., & Bers, M. (2024). Validating a Creative Coding Rubric through expressive activities for elementary grades. *Journal of Research on Technology in Education*, 1–20. https://doi.org/10.1080/15391523.2024.2398502

Bonner, H. J., Wong, K. S., Pedwell, R. K., & Rowland, S. L. (2019). A short-term peer mentor/mentee activity develops bachelor of science students' career management skills. *Mentoring & Tutoring: Partnership in Learning*, *27*(5), 509–530. https://doi.org/10.1080/13611267.2019.1675849

Bower, M., & Falkner, K. (2015). *Computational Thinking, the Notional Machine, Pre-Service Teachers, and Research Opportunities*. 17th Australasian Computing Education Conference, Sydney, AUS. https://crpit.scem.westernsydney.edu.au/confpapers/CRPITV160Bower.pdf

Bragg, L. A., Walsh, C., & Heyeres, M. (2021). Successful design and delivery of online professional development for teachers: A systematic review of the literature. *Computers & Education,166*, 104158. https://doi.org/10.1016/j.compedu.2021.104158

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa

Broley, L., Buteau, C., & Sardella, J. (2023). When preservice and inservice teachers join forces: A collaborative way to support the enactment of new coding curricula in mathematics classrooms. *Journal of Pedagogical Research, 7*(2). https://doi.org/10.33902/JPR.202318636

Burack, C., Melchior, A., & Hoover, M. (2019). Do after-school robotics programs expand the pipeline into STEM majors in college? *Journal of Pre-College Engineering Education Research (J-PEER), 9*(2). https://doi.org/10.7771/2157-9288.1244

Butt, M., Aziz, A., & Nadeem, M. (2021). An investigation of the factors preventing teachers to attend professional development. *Global Educational Studies Review VI(I,* 86–93. https://doi.org/10.31703/gesr.2021(VI-I).09

Campbell, C. (2014). *Teachers teaching teachers: A sustainable and inexpensive professional development program to improve instruction*. Doctoral, Portland State University. https://doi.org/10.15760/etd.2071

Chen, Y., & Cao, L. (2022). Promoting maker-centred instruction through virtual professional development activities for K-12 teachers in low-income rural areas. *British Journal of Educational Technology,53*(4), 1025–1048. https://doi.org/10.1111/bjet.13183

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.

Copur-Gencturk, Y., & Tolar, T. (2022). Mathematics teaching expertise: A study of the dimensionality of content knowledge, pedagogical content knowledge, and content-specific noticing skills. *Teaching and Teacher Education*, *114*, 103696. https://doi.org/10.1016/j.tate.2022.103696

Copur-Gencturk, Y., Li, J., Cohen, A. S., & Orrill, C. H. (2024). The impact of an interactive, personalized computer-based teacher professional development program on student performance: A randomized controlled trial. *Computers & Education,210*, 104963. https://doi.org/10.1016/j.compedu.2023.104963

Corradini, I., Lodi, M., & Nardelli, E. (2017). Conceptions and misconceptions about computational thinking among Italian primary school teachers. *Proceedings of the 2017 ACM Conference on International Computing Educational Research,* 136–144. https://doi.org/10.1145/3105726.3106194

CSTA (2017). *Computational Thinking Competencies*. https://iste.org/standards/computational-thinking-competencies

Darling-Hammond, L., Hyler, M., & Gardner, M. (2017). *Effective Teacher Professional Development*. Learning Policy Institute.

de Ruiter, L. E., & Bers, M. U. (2022). The coding stages assessment: Development and validation of an instrument for assessing young children's proficiency in the ScratchJr programming language.

Computer Science Education,32(4), 388–417. https://doi.org/10.1080/08993408.2021.1956216

Delacruz, S. (2020). Starting from scratch (Jr.): Integrating code literacy in the primary grades. *The Reading Teacher*, *73*(6), 805–812. https://doi.org/10.1002/trtr.1909

Desimone, L. M. (2009). Improving impact studies of teachers' professional development: Toward better conceptualizations and measures. *Educational Researcher*, *38*(3), 181–199. https://doi.org/10.3102/0013189X08331140

Eells, R. J. (2011). *Meta-analysis of the relationship between collective teacher efficacy and student achievement* [Dissertation, Loyola University Chicago]. https://ecommons.luc.edu/luc_diss/133?utm_source=ecommons.luc.edu%2Fluc_diss%2F133&utm_medium=PDF&utm_campaign=PDFCoverPages

Egbert, J., Shahrokni, S. A., Abobaker, R., & Borysenko, N. (2021). It's a chance to make mistakes: Processes and outcomes of coding in 2nd grade classrooms. *Computers & Education*, *168*, 104173. https://doi.org/10.1016/j.compedu.2021.104173

Ertmer, P. A., Ottenbreit-Leftwich, A. T., Sadik, O., Sendurur, E., & Sendurur, P. (2012). Teacher beliefs and technology integration practices: A critical relationship. *Computers & Education*, *59*(2), 423–435. https://doi.org/10.1016/j.compedu.2012.02.001

Fenton, M. P., Hawley, L., Frerichs, S. W., & Lodl, K. (2019). STEM professional development for youth workers: Results of a triangulated study. *Journal of Youth Development*, *14*(4), 178–196. https://doi.org/10.5195/jyd.2019.738

Fraillon, J., Ainley, J., Schulz, W., Friedman, T., & Duckworth, D. (2020). *Preparing for Life in a Digital World: IEA International Computer and Information Literacy Study 2018 International Report*. Springer International Publishing. https://doi.org/10.1007/978-3-030-38781-5

Frumin, K., Dede, C., Fischer, C., Foster, B., Lawrenz, F., Eisenkraft, A., Fishman, B., Levy, J., A., & McCoy, A. (2018). Adapting to large-scale changes in advanced placement biology, chemistry, and physics: The impact of online teacher communities. *International Journal of Science Education*, *40*(4), 397–420. https://doi.org/10.1080/09500693.2018.1424962

Gess-Newsome, J., Taylor, J. A., Carlson, J., Gardner, A. L., Wilson, C. D., & Stuhlsatz, M. A. M. (2019). Teacher pedagogical content knowledge, practice, and student achievement†. *International Journal of Science Education*, *41*(7), 944–963. https://doi.org/10.1080/09500693.2016.1265158

Gunzler, D., Chen, T., Wu, P., & Zhang, H. (2013). Introduction to mediation analysis with structural equation modeling. *Shanghai Archives of Psychiatry*, *25*(6), 390–394. https://doi.org/10.3969/j.issn.1002-0829.2013.06.009

Guskey, T. R., & Yoon, K. S. (2009). What works in professional development?? *Phi Delta Kappan*, *90*(7), 495–500. https://doi.org/10.1177/003172170909000709

Hassler, B., Hennessy, S., & Hofmann, R. (2018). *Sustaining and scaling pedagogic innovation in Sub-saharan Africa: Grounded insights for teacher professional development*. https://www.repository.cam.ac.uk/handle/1810/275192

Hill, H. C., & Loewenberg Ball, D. (2004). Learning mathematics for teaching: Results from california's mathematics professional development institutes. *Journal for Research in Mathematics Education*, *35*(5), 330–351. https://doi.org/10.2307/30034819

Hill, H. C., Lynch, K., Gonzalez, K. E., & Pollard, C. (2020). Professional development that improves STEM outcomes. *Phi Delta Kappan*, *101*(5), 50–56. https://doi.org/10.1177/0031721720903829

Hinckle, M., Rachmatullah, A., Mott, B., Boyer, K. E., Lester, J., & Wiebe, E. (2020). The relationship of gender, experiential, and psychological factors to achievement in computer science. *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, 225–231. ITiCSE '20: Innovation and Technology in Computer Science Education. https://doi.org/10.1145/3341525.3387403

Hooper, D., Coughlan, J., & Mullen, M. (2008). Structural equation modelling: Guidelines for determining model fit. *Electronic Journal of Business Research Methods*, *6*(1), 53–60. https://doi.org/10.21427/D7CF7R

Huang, X., Bernacki, M. L., Kim, D., & Hong, W. (2022). Examining the role of self-efficacy and online metacognitive monitoring behaviors in undergraduate life science education. *Learning and Instruction*, *80*, 101577. https://doi.org/10.1016/j.learninstruc.2021.101577

Hudin, S. (2023). A systematic review of the challenges in teaching programming for primary schools' students. *Online Journal for TVET Practitioners*, *8*(1), 75–88.

Hußner, I., Lazarides, R., & Symes, W. (2024). The relation between self-efficacy and teaching behaviour: A video-based analysis of student teachers. *Learning and Instruction*, *91*, 101880. https://doi.org/10.1016/j.learninstruc.2024.101880

Iserbyt, P., Ward, P., & Li, W. (2017). Effects of improved content knowledge on pedagogical content knowledge and student performance in physical education. *Physical Education and Sport Pedagogy*, *22*(1), 71–88. https://doi.org/10.1080/17408989.2015.1095868

Jerrim, J., Sims, S., & Oliver, M. (2023). Teacher self-efficacy and pupil achievement: Much ado about nothing? International evidence from TIMSS. *Teachers and Teaching*, *29*(2), 220–240. https://doi.org/10.1080/13540602.2022.2159365

Kaya, E., Yesilyurt, E., Newley, A., & Deniz, H. (2019). Examining the impact of a computational thinking intervention on pre-service elementary science teachers' computational thinking teaching efficacy beliefs, interest and confidence. *Journal of Computers in Mathematics and Science Teaching, 38*(4), 385–392

Kelley, T. R., Knowles, J. G., Holland, J. D., & Han, J. (2020). Increasing high school teachers' self-efficacy for integrated STEM instruction through a collaborative community of practice. *International Journal of STEM Education, 7*(14). https://doi.org/10.1186/s40594-020-00211-w

Kline, R. B. (2016). *Principles and practcice of structural equation modeling* (4th ed.). Guilford Press.

Kong, R., & Wong, G. K. W. (2017). Teachers' perception of professional development in coding education. *2017 IEEE 6th International Conference on Teaching Assessment and Learning for Engineering (TALE)*, *377*, 380. https://doi.org/10.1109/TALE.2017.8252365

König, J., Blömeke, S., Jentsch, A., Schlesinger, L., Nehls, Musekamp, C. F., F., & Kaiser, G. (2021). The links between pedagogical competence, instructional quality, and mathematics achievement in the lower secondary classroom. *Educational Studies in Mathematics*, *107*(1), 189–212. https://doi.org/10.1007/s10649-020-10021-0

Krille, C. (2020). Individual and Context Characteristics Related to Teachers' Professional Development Behavior. In C. Krille, *Teachers' Participation in Professional Development* (pp. 41–90). Springer International Publishing. https://doi.org/10.1007/978-3-030-38844-7_5

Kulgemeyer, C., & Riese, J. (2018). From professional knowledge to professional performance: The impact of CK and PCK on teaching quality in explaining situations. *Journal of Research in Science Teaching*, *55*(10), 1393–1418. https://doi.org/10.1002/tea.21457

Lauer, P. A., Christopher, D. E., Firpo-Triplett, R., & Buchting, F. (2014). The impact of short-term professional development on participant outcomes: A review of the literature. *Professional Development in Education*, *40*(2), 207–227. https://doi.org/10.1080/19415257.2013.776619

Lauermann, F., & ten Hagen, I. (2021). Do teachers' perceived teaching competence and self-efficacy affect students' academic outcomes? A closer look at student-reported classroom processes and outcomes. *Educational Psychologist*, *56*(4), 265–282. https://doi.org/10.1080/00461520.2021.1991355

Lawrence, K., & Ogundolire, H. (2022). Repurposing African teachers for sustainable development: Online global trends. *World Journal on Educational Technology: Current Issues, 14*(3). https://doi.org/10.18844/wjet.v14i3.7192

Li, S., Liu, X., Yang, Y., & Tripp, J. (2022). Effects of teacher professional development and science classroom learning environment on teachers' science achievement. *Reseach in Science Education*, *52*, 1031–1053. https://doi.org/10.1007/s11165-020-09979-x

Loewenberg Ball, D., Thames, M. H., & Phelps, G. (2008). Content knowledge for teaching: What makes it special? *Journal of Teacher Education*, *59*(5), 389–407. https://doi.org/10.1177/0022487108324554

Louka, K., & Papadakis, S. (2024). Enhancing computational thinking in early childhood education through ScratchJr integration. *Heliyon*, *10*(10), e30482. https://doi.org/10.1016/j.heliyon.2024.e30482

Mason, S. L., & Rich, P. J. (2019). Preparing elementary school teachers to teach computing, coding, and computational thinking. *Contemporary Issues in Technology and Teacher Education*, *19*(4), 790–824.

Master, A., Cheryan, S., Moscatelli, A., & Meltzoff, A. N. (2017). Programming experience promotes higher STEM motivation among first-grade girls. *Journal of Experimental Child Psychology*, *160*, 92–106. https://doi.org/10.1016/j.jecp.2017.03.013

McCormick, K. I., & Hall, J. A. (2022). Computational thinking learning experiences, outcomes, and research in preschool settings: A scoping review of literature. *Education and Information Technologies*, *27*(3), 3777–3812. https://doi.org/10.1007/s10639-021-10765-z

Mcinerney, C. (2021). *Second level computer science teacher self-efficacy and how it influences the use of teaching and assessment strategies* [Thesis, University of Limerick]. https://researchrepository.ul.ie/articles/thesis/Second_level_computer_science_teacher_self-efficacy_and_how_it_influences_the_use_of_teaching_and_assessment_strategies/19811470/1

Mcinerney, C., Exton, C., & Hinchey, M. (2020). A study of high school computer science teacher confidence levels. *15th Workshop on Primary and Secondary Computing Education*, *20*, 1–18. https://doi.org/10.1145/3421590.3421614

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. https://doi.org/10.1080/08993408.2015.1111645

Mihm, C. (2021). Why teach coding to early elementary learners. In M. U. Bers (Ed.), *Teaching computational thinking and coding to young children*. IGI Global. https://www.igi-global.com/chapter/why-teach-coding-to-early-elementary-learners/286041

Miller, B., Kirn, A., Anderson, M., Major, J., Feil-Seifer, D., & Jurkiewicz, M. (2018). *Unplugged robotics to increase K-12 students' engineering interest and attitudes,* 1–5. https://doi.org/10.1109/FIE.2018.8658959

Mills, K. A., Cope, J., Scholes, L., & Rowe, L. (2024). Coding and computational thinking across the curriculum: A review of educational outcomes. *Review of Educational Research,* 00346543241241327. https://doi.org/10.3102/00346543241241327

Moè, A. (2021). Doubling mental rotation scores in high school students: Effects of motivational and strategic trainings. *Learning and Instruction*, *74*, 101461. https://doi.org/10.1016/j.learninstruc.2021.101461

Mok, M. M. C., & Moore, P. J. (2019). Teachers & self-efficacy. *Educational Psychology,39*(1), 1–3. https://doi.org/10.1080/01443410.2019.1567070

Montuori, C., Gambarota, F., Altoé, G., & Arfé, B. (2024). The cognitive effects of computational thinking: A systematic review and meta-analytic study. *Computers & Education,210*, 104961. https://doi.org/10.1016/j.compedu.2023.104961

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*, *186*, 104530. https://doi.org/10.1016/j.compedu.2022.104530

NGSS Lead States. (2013). *Next Generation Science Standards: For States, By States*. National Academies. https://doi.org/10.17226/18290

Nissen, J. M., & Shemwell, J. T. (2016). Gender, experience, and self-efficacy in introductory physics. *Physical Review Physics Education Research*, *12*(2), 020105. https://doi.org/10.1103/PhysRevPhysEducRes.12.020105

Odden, A., & Picus, L. (2014). *School finance: A policy perspective* (5th Edition).

Papadakis, S. (2020). Apps to promote Computational Thinking concepts and coding skills in children of preschool and pre-primary school age. In *Mobile learning applications in early childhood education* (pp. 101–121). Information Science Reference/IGI Global. https://doi.org/10.4018/978-1-7998-1486-3.ch006

Preacher, K. J., & Hayes, A. F. (2004). SPSS and SAS procedures for estimating indirect effects in simple mediation models. *Behavior Research Methods Instruments & Computers*, *36*(4), 717–731. https://doi.org/10.3758/BF03206553

Rich, P. J., Jones, B., Belikov, O., Yoshikawa, E., & Perkins, M. (2017). Computing and engineering in elementary school: The effect of year-long training on elementary teacher self-efficacy and beliefs about teaching computing and engineering. *International Journal of Computer Science Education in Schools, 1*(1). https://doi.org/10.21585/ijcses.v1i1.6

Rich, P. J., Larsen, R. A., & Mason, S. L. (2021a). Measuring teacher beliefs about coding and computational thinking. *Journal of Research on Technology in Education*, *53*(3), 296–316. https://doi.org/10.1080/15391523.2020.1771232

Rich, P. J., Mason, S. L., & O'Leary, J. (2021b). Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking. *Computers & Education*, *168*, 104196. https://doi.org/10.1016/j.compedu.2021.104196

Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test. *Computers in Human Behavior*, *72*, 678–691. https://doi.org/10.1016/j.chb.2016.08.047

Rosson, M. B., Carroll, J., & Sinha, H. (2011). Orientation of undergraduates toward careers in the computer and information sciences: Gender, self-efficacy and social support. *ACM Transactions on Computing Education*, *11*(3), 1–23. https://doi.org/10.1145/2037276.2037278

Schreiber, J. B., Nora, A., Stage, F. K., Barlow, E. A., & King, J. (2006). Reporting structural equation modeling and confirmatory factor analysis results: A review. *The Journal of Educational Research*, *99*(6), 323–338. https://doi.org/10.3200/JOER.99.6.323-338

Schwarzhaupt, R., Wilson, J., Lee, F., & Raspberry, M. (2021). Teachers' engagement and self-efficacy in a PK–12 computer science teacher virtual community of practice. *Journal of Computer Science Integration*. https://doi.org/10.26716/jcsi.2021.10.8.34

Sentence, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teachers' perspective. *Education*

and Information Technologies, *24*, 469–495. https://doi.org/10.1007/s10639-016-9482-0
Shulman, L. S. (1986). Those who understand: Knowledge growth in teaching. *Educational Researcher*, *15*(2), 4–14. https://doi.org/10.3102/0013189X015002004
Simmonds, J., Gutierrez, F. J., Meza, F., Torrent, C., & Villalobos, J. (2021). Changing teacher perceptions about computational thinking in grades 1–6 through a national teacher training program. *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21), 260–266*,. https://doi.org/10.1145/3408877.3432542
Singh, R. K. (2018). Teacher's efficacy: Review and update. *International Journal of Science and Research, 9*(1). https://doi.org/10.21275/ART20204299
Skaalvik, E. M., Federici, R. A., & Klassen, R. M. (2015). Mathematics achievement and self-efficacy: Relations with motivation for mathematics. *International Journal of Educational Research*, *72*, 129–136. https://doi.org/10.1016/j.ijer.2015.06.008
Sun, Z., Xu, R., Deng, L., Jin, F., Song, Z., & Lin, C. H. (2023). Beyond coding and counting: Exploring teachers' practical knowledge online through epistemic network analysis. *Computers & Education,192*, 104647. https://doi.org/10.1016/j.compedu.2022.104647
Swarnalatha, S. (2019). Influence of teacher self efficacy on academic achievement of secondary school students. *The International Journal of Indian Psychology, 7*(3). https://doi.org/10.25215/0703.073
Tai, R. H., Liu, C. Q., Maltese, A. V., & Fan, X. (2006). Planning for early careers in science. *Science*, *312*(5777), 1143–1144. https://doi.org/10.1126/science.1128690
Tang, K. Y., Chou, T. L., & Tsai, C. C. (2020). A content analysis of computational thinking research: An international publication trends and research typology. *The Asia-Pacific Education Researcher*, *29*(1), 9–19. https://doi.org/10.1007/s40299-019-00442-8
Thompson, J., & Childers, G. (2021). The impact of learning to code on elementary students' writing skills. *Computers & Education,175*, 104336. https://doi.org/10.1016/j.compedu.2021.104336
Tschannen-Moran, M., & Hoy, A. W. (2007). The differential antecedents of self-efficacy beliefs of novice and experienced teachers. *Teaching and Teacher Education*, *23*(6), 944–956. https://doi.org/10.1016/j.tate.2006.05.003
Usher, E. L., Li, C. R., Butz, A. R., & Rojas, J. P. (2019). Perseverant grit and self-efficacy: Are both essential for children's academic success? *Journal of Educational Psychology*, *111*(5), 877–902. https://doi.org/10.1037/edu0000324
Wakil, K., KHDIR, S., SABIR, L., & Nawzad, L. (2019). Student ability for learning computer programming languages in primary schools. *International E-Journal of Educational Studies*, *3*, 109–115. https://doi.org/10.31458/iejes.591938
Wang, X. C., Choi, Y., Benson, K., Eggleston, C., & Weber, D. (2020). Teacher's role in fostering preschoolers' computational thinking: An exploratory case study. *Early Education and Development*, *32*(1), 26–48. https://doi.org/10.1080/10409289.2020.1759012
Warner, J. R., Fletcher, C. L., Torbey, R., & Garbrecht, L. S. (2019). Increasing capacity for computer science education in rural areas through a large-scale collective impact model. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 1157–1163. https://doi.org/10.1145/3287324.3287418
Williams, H. E., Williams, S., & Kendall, K. (2020). CS in schools: Developing a sustainable coding programme in Australian schools. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education* (pp. 321–327). https://doi.org/10.1145/3341525.3387422
Wing, J. M. (2011). Research notebook: Computational thinking—What and Why? *he Link: Carnegie Mellon University School of Computer Science*. https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why
Witherspoon, E. B., Higashi, R. M., Shunn, C. D., Baehr, E. C., & Shoop, R. (2017). Developing computational thinking through a virtual robotics programming curriculum. *ACM Transactions on Computing Education*, *18*(1), 1–20. https://doi.org/10.1145/3104982
Yadav, A., Gretter, S., Hambrusch, S., & Sands, P. (2016). Expanding computer science education in schools: Understanding teacher experiences and challenges. *Computer Science Education*, *26*(4), 235–254.
Yeşilyurt, E., Ulaş, A. H., & Akan, D. (2016). Teacher self-efficacy, academic self-efficacy, and computer self-efficacy as predictors of attitude toward applying computer-supported education. *Computers in Human Behavior*, *64*, 591–601. https://doi.org/10.1016/j.chb.2016.07.038
Zhao, X., LynchJr., J. G., & Chen, Q. (2010). Reconsidering Baron and kenny: Myths and truths about mediation analysis. *Journal of Consumer Research*, *37*(2), 197–206. https://doi.org/10.1086/651257
Zheng, X., Yin, H., & Li, Z. (2019). Exploring the relationships among instructional leadership, professional learning communities and teacher self-efficacy in China. *Educational Management Administration & Leadership,47*(6), 843–859. https://doi.org/10.1177/1741143218764176
Zhou, N., Nguyen, H., Fischer, C., Richardson, D., & Warschauer, M. (2020). High school teachers' self-efficacy in teaching computer science. *ACM Transactions on Computing Education,20*(3), 1–18. https://doi.org/10.1145/3410631
Zhou, X., Shu, L., Xu, Z., & Padrón, Y. (2023). The effect of professional development on in-service STEM teachers' self-efficacy: A meta-analysis of experimental studies. *International Journal of STEM Education, 10*(37). https://doi.org/10.1186/s40594-023-00422-x