Computers and Education Open 10 (2026) 100353

Contents lists available at ScienceDirect

Computers and Education Open

journal homepage: www.sciencedirect.com/journal/computers-and-education-open

Supporting K-5 computer science integration through high-quality teacher professional development

Shanan Chappell Moots [a,*], Joanna K. Garner [a], Joseph A. Brobst [a], Melani Loney [a], Lisa Steffian [a], Jennifer Maeng [b]

[a] The Center for Educational Partnerships, Darden College of Education and Professional Studies, Old Dominion University, Norfolk, VA USA

[b] Curriculum, Instruction, and Special Education, School of Education and Human Development, University of Virginia, Charlottesville, VA USA

ARTICLE INFO

Keywords:
Computer science integration
K-5
Professional development
Teacher efficacy

ABSTRACT

Workforce development and education leaders have increasingly emphasized the need for high-quality computer science (CS) instruction for K-12 students. Though states have created and mandated the implementation of CS curriculum standards, few in-service teachers have been provided sufficient opportunities to develop CS pedagogical content knowledge and self-efficacy. This study evaluated the effect of a CS integration professional development (PD) program on K-5 teachers' perceptions of their capacity to teach CS and their implementation of CS-integrated lessons using a randomized controlled trial design. Treatment included an intensive online summer institute with school year follow-up. Results indicate statistically significant effects of the PD on teachers' confidence to teach CS, experience integrating CS into core content, and ability to engage students during CS lessons; treatment teachers also taught significantly more CS lessons than control teachers. Findings indicate that the program provides a scalable model for delivering PD designed to expand access to CS at the elementary level.

Introduction

The field of computer science (CS) education has experienced a surge in interest toward ensuring broader access to CS learning opportunities across the United States [1,2]. This is encouraging, because there are millions of unfilled jobs in Science, Technology, Engineering, Mathematics, and Computer Science (STEM-CS) in the United States (U.S.), and more than 90 % of today's jobs require at least some level of digital skills [3]. With so many careers requiring knowledge of CS concepts [4], there is a critical need for CS education for all U.S. K-12 students. However, despite agreement among education and industry professionals about the need for increased STEM-CS literacy and career preparedness among all students [5,6], schools serving large populations of economically disadvantaged and underrepresented students are unlikely to provide CS instruction [7–9]. This is because educators in rural and low-income school settings often lack access to high quality, evidence-based professional development (PD), limiting their opportunities to gain STEM-CS subject matter expertise and pedagogical knowledge [10].

STEM-CS as a component of K-12 learning

Today's students would benefit from knowledge of both STEM and CS to complete daily tasks associated with learning in the twenty-first century classroom. Technologies such as artificial intelligence, augmented and/or virtual reality, and the internet have grown to have a profound impact on a child's life and learning, but students' understanding of the mechanisms and consequences of these technologies has not kept pace with their utilization in the classroom [11]. Moreover, student understanding of safety issues associated with the CS tools may be limited, putting students at risk when using the tools. Therefore, it is important for students to learn about the safe and ethical use of CS in the classroom [12], preferably at an early age.

The inclusion of STEM-CS in K-12 lessons can enhance students' academic understanding as well as social and cognitive development. CS strategies and learning tools that provide a platform for problem solving and computational thinking can enhance K-12 learning and social reasoning [13], or the ability to analyze and draw inferences about others' actions and intentions to help regulate their own behavior. In a study conducted by Strawhacker and Bers [14] a correlation was made

* Corresponding author.
E-mail address: schappel@odu.edu (S.C. Moots).

https://doi.org/10.1016/j.caeo.2026.100353
Received 26 June 2025; Received in revised form 12 March 2026; Accepted 4 April 2026
Available online 5 April 2026

between early elementary students' social reasoning development, and programming logic and practices when learning to code using Scratch Jr. In another study, young students utilizing tablet computers as a component of science were found to have high quality teacher-student interactions, including discussions to promote student understanding and a longer duration of student engagement [15], and the student benefits extended beyond the use of these tools.

Instruction through the integration of subject areas provides a means of meeting students at their current level of concept understanding and moving them forward toward new levels of comprehension while piquing their interest and enhancing their learning experience [16]. The integration of content-area concepts with CS concepts has also been shown to enhance the personalization of instruction by providing a platform for tailoring the lesson to student readiness through the selection of unique instructional materials [17]. Integrating CS with the core content areas of English, science, mathematics, and social studies has been found to provide students with a platform for gaining an understanding of key computational thinking skills such as abstraction, data, modeling and simulation and algorithms [18], which we believe may enhance students' utilization of CS concepts at home and at school.

STEM-CS in the workforce

As the twenty-first century progresses, components of both technology and CS are taking a greater role in business, education, and leisure activities. Cyber-physical systems, digital twins, blockchain, three-dimensional printing and artificial intelligence are becoming prevalent in many aspects of daily life [19]. The modern workplace requires employees trained in the STEM fields, who can utilize the concepts associated with CS and computational thinking [20].

In 2021, 24 % of the workforce in the US was employed in STEM fields (NSB, 2024). Computer and information technology is projected to grow much faster than other STEM occupations from 2023 to 2033 [21]. It is estimated that by 2026, IT skills shortages will be present in over 90 % of organizations globally [22]. Whereas the gap between labor force requirements and the number of individuals trained in CS areas will increase over the next 10 years, only 60 % of U.S. public high schools offer at least one foundational CS course and only 6.4 % of high school students are enrolled in CS courses [7].

Statement of the problem

In response to the increasing importance of STEM-CS knowledge and skills in K-12 and the workforce, since 2017 states across the U.S. have begun mandating broad implementation of explicit CS curriculum standards. While there is support for positioning CS as a distinct subject [23], this is impractical given the shortage of elementary CS teachers in most school districts and the time constraints placed on classroom teachers providing other core courses [24]. An alternative option, and the one adopted by the state of Virginia where the present study was conducted, is to train classroom teachers to integrate CS into other content areas [25]. Virginia was among the first states to develop and adopt CS curriculum standards, and policymakers in the state recognized that integration of these standards into the core curricular content areas was a pragmatic approach to teaching and learning CS.

One immediate consequence that emerged from Virginia's adoption of this policy, which required CS to be integrated into existing subjects from Kindergarten onwards, was that grade-level classroom teachers bore the primary responsibility for integrating CS into instruction, even though many had no prior training in CS content or pedagogy [9,25]. This was particularly true at the K-5 level in Virginia, where fewer than one in five elementary schools surveyed were teaching CS [7], despite evidence suggesting that CS activities can support the development of critical and computational thinking even for younger learners [26] and that building CS literacy simultaneously with other early-childhood literacies appears realistic and feasible [27]. Still, while promising findings are emerging from research on PD for CS-integration at the elementary level, studies to date have been focused primarily on quasi-experimental techniques [28], qualitative approaches [29], or single-case designs [30]. Though one randomized controlled trial examining impacts at the elementary level was conducted by Yang et al. [8], the study focused on a prescribed CS curriculum rather than multi-disciplinary CS-integration. To date, little causal-comparative evidence exists to support the impact of high-quality teacher CS-integration PD designed to promote access to CS concepts and build skills among elementary students, which are necessary steps in paving the pathway to advanced CS coursework at the secondary level.

K-5 teachers must be effectively trained to deliver high-quality instruction that provides equitable opportunities and promotes student success related to CS learning outcomes to maximize students' readiness to pursue middle and high school CS coursework, post-secondary CS studies, and/or meaningful CS-related employment. However, educators tasked with cross-content integration often face challenges related to self-efficacy, content knowledge, and pedagogical skills necessary for effective instruction [23,31,32]. PD is widely recognized as essential in equipping teachers with the concepts and skills to deliver effective CS-integrated instruction [23,32], but CS-integration training should also address educators' motivation and self-efficacy [31], as these factors significantly influence the likelihood of implementing CS content in their teaching post-training [33].

Purpose of the study and research questions

The current study is a subset of a larger project, [REDACTED], which was designed to broaden elementary students' experience with CS by increasing teacher CS pedagogical and content knowledge and self-efficacy and improving their instructional skills to teach CS through an interdisciplinary lens. We limited our recruitment to teachers serving in schools with large populations (50 % or greater) of economically disadvantaged and/or underrepresented minority[1] students. Initially, the project also focused on elementary teachers in rural school settings in Virginia, though participation was later made available to a broader selection of schools in the state. The project's leadership team included university faculty, leaders of a statewide non-profit agency with expertise in K-12 CS education, and representatives from the state Department of Education. The project's implementation team included university faculty and non-profit agency staff. A third-party evaluator was also engaged throughout the project and managed data collection tasks. The overarching goal of the project was to broaden participation in CS through high-quality PD leading to CS integrated instruction for K-5 students.

Program impact on participating teachers' CS content knowledge and on student outcomes was examined in a separate impact study [34]; the purpose of the current study was to examine impacts of the [REDACTED] PD on other teacher outcomes, including their confidence and experiences in integrating CS concepts into their core curriculum and the frequency with which they teach CS lessons. To do this, we examined the following research questions:

1. How does the [REDACTED] program's CS integration PD impact participating teachers' confidence in performing and teaching CS, experience teaching CS, and ability to engage students during CS lessons compared to teachers randomized to the control condition?
2. How does [REDACTED] program's CS integration PD impact the number of CS lessons taught by participating teachers compared to the number of lessons taught by teachers randomized to the control condition?

[1] For purposes of this study, we are defining underrepresented students as those who identify as Black, Hispanic, female, or who receive special education services.

Theoretical framework

The design of the [REDACTED] CS integration PD model is grounded in situated learning theory [35], specifically the concept that elementary teachers may be effectively prepared to integrate CS into core curricular areas via engagement in a community of practice (CoP). [REDACTED] teacher participants first build their knowledge, experience, confidence, and perceived competence related to CS integration by participating in an intensive PD workshop that promotes learning of both CS content and instructional strategies in a coaching-focused, supportive online community. Teachers' CS-related professional learning is then bolstered as they participate in a networked improvement community (NIC) [36,37] during the school year following the intensive PD workshop. Membership in this community spans both grade-level teams within participating schools and the larger pool of [REDACTED] teachers across the project and the state. The [REDACTED] model builds directly upon prior professional development projects that utilized communities of practice and showed significant positive impacts related to mathematics [38] and science [39,40]. At the same time, it is well-aligned to projects that have demonstrated the value of utilizing learning communities and peer support within professional development activities geared toward CS teachers particularly [41–43].

Methods

Researchers and educators from the [REDACTED] project's implementation and evaluation teams partnered on this study. We employed a true experimental design through a two-cohort randomized controlled trial to examine program impacts on outcomes for elementary teachers randomly assigned to the treatment condition, specifically their confidence towards and experience in teaching and integrating CS concepts and standards into their instruction and their self-perceived ability to engage their students during CS lessons, as compared to those randomized to the business-as-usual control group [44]. Although some minor programmatic changes were made to the PD model in cohort two (see Procedures section below), the treatment and measures remained consistent across the cohorts, allowing us to combine the cohorts to investigate the causal impact of the model over the duration of the project. Our institution's Human Subjects Review Board reviewed the project prior to the onset of study activities and determined that the protocol met the qualifications for approval as described in 45 CFR 46.

Participants

To support broader scaling of the project within schools and to promote sustainability of CS teaching and learning, we encouraged school-level participation teams. Elementary-level educators (teachers, librarians, media specialists, STEM coaches, etc.) from schools with large populations (50 % or greater) of economically disadvantaged and/or underrepresented minority students across Virginia were recruited to participate in the program, with activities beginning in summer of 2021 for cohort one and summer of 2023 for cohort two. All participants were instructional staff who were responsible for instruction for one or more classes during their cohort. In cohort one, 77 teachers from 23 schools enrolled in the program and consented to participate whether assigned to treatment or control, with n=33 teachers randomized to treatment and n=44 randomized to the control condition[2]. For cohort two, n=121 teachers from 47 schools consented to participation, with n=60 teachers randomized to the treatment condition and n=61 teachers assigned to control.

For the current study, we only included those participants who completed the pre-survey and end-of-year survey, which resulted in an analysis sample of n=132, with n=67 treatment teachers and n=65 control teachers. The groups were statistically equivalent for number of teachers who hold a STEM-CS degree (T=7, C=4, p =.58), mean years of teaching experience (T=15.2, C=14.9, p = .85), gender (T=67 female, C=57 female, p = .06), and ethnicity (T=59 white, C=47 white, p = .07). The groups were also statistically similar on whether participants had attended any non-[REDACTED] CS PD during the project year (T=22 did, C=25 did, p = .58). Though school-level demographics were included as strata in the randomization process, there were minor differences in student populations by treatment group for our analysis sample, where teachers randomized to the treatment condition were serving in schools with slightly larger percentages of economically disadvantaged students (51.5 % compared to 48.7 %) and underrepresented minority students (48.7 % compared to 47.6 %) than those randomized to the control condition.

Data sources

Participating teachers were surveyed using the [REDACTED] Teacher Survey prior to the PD and at the end of the school year. This questionnaire consists of measures to assess participants' content knowledge, pedagogical knowledge, self-efficacy, and culturally responsive teaching. Items also assess CS confidence and experience teaching and integrating CS. Items assessing confidence and experience with CS and content knowledge were aligned with the *Computer Science Standards of Learning Framework* (Virginia Department of Education, 2017). We adapted items from Bean et al.'s [45] *Teachers' Self-efficacy in Computational Thinking* survey, Washington et al.'s [46] *Computer Science Attitudes and Identity Survey*, the *Computing Attitudes Survey* [47], the *Culturally Responsive Teaching Survey* [48], and the *Culturally Responsive Teaching Self-Efficacy Scale* [49] to create the instrument (Appendix A).

The measure was piloted and revised prior to administration for the current study, with the same versions (pre/post) used for both cohorts. Support for face and content validity was established through a review of the assessment by a panel of subject matter experts. Reliability (Cronbach's α) for the Confidence, Experience, and Engagement subscales on the measure ranged from α = .74 - .94.

Procedures

By integrating CS curricular standards into core elementary subjects, the [REDACTED] program aims to enrich student learning and increase readiness for future STEM and CS coursework. Like STEM-integration PD, a deliberate and strategic approach to CS-integration PD is more effective than simply increasing the number of training opportunities [25]. CS-integration PD should be intentionally designed to structure learning experiences and offer multiple opportunities to practice newly acquired skills [50]. Training should also take place over multiple sessions with appropriate spacing to prevent information overload and burnout while promoting classroom implementation [51,52]. Following this guidance, the PD sequence employed in this study consisted of a 5-day summer institute, followed by teacher engagement in a NIC [37] that allowed for multiple opportunities for interaction amongst participants. A critical element of this initiative is its partnership with a statewide not-for-profit agency that provides K-12 CS educational support to schools throughout Virginia. This collaboration focuses on hands-on, practical learning experiences that inspire teachers to become advocates for CS education within their local communities. To ensure that educators are properly equipped for this digital transition, the program also provides each participant with a Chromebook.

Core professional development components. The program's instructional model is delivered through a blend of intensive summer training and ongoing school-year support. First, participants completed a Summer PD Academy, a five-day, comprehensive course designed for elementary school educators. The academy introduced participants to

[2] Randomization occurred at the school level, with some schools having more participants than others, which may have resulted in unequal numbers of teachers per group.

the six strands of the Virginia CS *Standards of Learning* [53]: (1) Algorithms and Programming, (2) Computing Systems, (3) Cybersecurity, (4) Data and Analysis, (5) Impacts of Computing, and (6) Networking and the Internet. Activities focused on identifying overlapping concepts between core curricula and CS, enabling teachers to design integrated lessons. Participants utilized a specialized CS Integration Guide [54] to explore pre-made lessons with expert facilitators. To accommodate diverse schedules and extend reach through the state, two summer sessions were offered for each cohort, with activities occurring synchronously and asynchronously online.

Learning objectives for the summer institute included that participants would develop: a) Knowledge of the Virginia CS curriculum standards; b) Coding skills using Scratch programming language; c) Experience creating integrated CS lessons d) Experience teaching/implementing CS professional learning activities (to support their peers) e) Awareness of resources and tools to support teacher and student learning (for both in in-person and online settings) f) Confidence in coaching others in their respective schools on CS-integration. During the 5-day academy, teachers learned instructional strategies for integrating the Virginia CS curriculum standards' strands into elementary instruction in reading, writing, science, mathematics, and social studies. They developed pedagogical knowledge and assessment literacy designed to enable them to teach and assess students' understanding and acquisition of CS concepts and skills. Participants gained experience developing integrated CS lesson plans, for which they then received feedback and guidance from the PD facilitators (see Supplementary Materials for a sample lesson created through the program). [REDACTED] also integrated a culturally responsive teaching component to building participating teachers' capacity to incorporate students' interests and experiences into lesson content, particularly when new content (CS, in this case) is introduced. Making new content culturally and contextually relevant can promote students' sense of social belonging, self-efficacy, and academic achievement [55,56].

Networked Improvement Community. Following the summer academy, the program continued through a NIC [37]. NICs are professional learning groups that possess four key characteristics: they focus on a well-specified aim; they are guided by a deep understanding of a problem and develop a theory of change to solve it; they deliberately attend to improvement metrics to demonstrate movement toward an intended solution; they are coordinated such that educational interventions can be implemented in varying contexts [36]. Teacher engagement in the NIC was promoted through an online resource platform and a series of virtual "Learning Bytes" modules offered by our non-profit CS partner who provided the summer academy. Learning Bytes were succinct, two-hour virtual sessions that explored advanced CS concepts and pedagogical strategies, including topics such as Artificial Intelligence basics, Cybersecurity, and programming in Scratch or Twine. These sessions were designed to be accessible to all participants regardless of their teaching context. Multiple Learning Bytes sessions were held throughout the school year to maximize attendance and participation. Teachers were expected to complete four of these modules during the school year and were able to select the topics that most interested them or best fit with their needs.

Targeted Support and Mentorship. Recognizing that classroom implementation can be challenging, the program also introduced two additional layers of support for cohorts two and three: Keeping in Touch Sessions (KITS) and mentoring. KITS were one-hour virtual workshops designed to help teachers use the programmable, robotic educational tools provided by the program. KITS provided setup training and hands-on activities aligned with state standards. KITS were led by industry professionals skilled at introductory and advanced level implementation of the tools to support all participants regardless of their experience or comfort level with the tools. Participants were also assigned a program mentor to provide regular communication and troubleshooting support. These mentors were experienced classroom teachers skilled in CS pedagogical content knowledge who helped participants overcome the specific challenges of integrating CS into core elementary curricula.

Commitment to Equity and Accessibility. A fundamental aspect of the program is its dedication to culturally competent instruction, aiming to counteract STEM-CS attrition among minority, rural, and female students. By teaching educators to incorporate diverse cultures and lived experiences into their materials through provision of resources and examples of culturally responsive exemplars that are easily incorporated into new or existing lessons, the program fosters an inclusive educational environment supportive of all students in today's technology-driven world.

Analytic approach

We examined between-group differences using multiple analysis of covariance (MANCOVA) techniques, with treatment status as the independent variable. We calculated subscale composite scores from the pre- and end-of-year surveys for the 13 items related to confidence in programming and teaching CS (Confidence), 6 items focusing on experience integrating CS (Experience), and 4 items regarding ability to engage students during CS lessons (Engagement) and used the scores at pre-survey point as covariates to control for differences prior to treatment. Although differences prior to PD were not statistically significant, the control group had higher initial mean scores on Confidence and Engagement. We also included a covariate that identified whether participants had engaged in any non-project CS PD during the program year. We used the three composite scores (Confidence, Experience, Engagement) from the end-of-year survey as dependent, or outcome, variables.

Participants were also asked if they had taught any CS or CS integration lessons during the program year, and if yes, they were asked to provide the number of lessons for each of the six CS curriculum standards strands (e.g., 1-2 lessons, 3-4 lessons, 5 or more lessons). More treatment respondents than control respondents answered yes ($n = 51$ treatment, $n = 30$ control) and subsequently estimated the number of CS lessons taught. Given the different response rate than the other three dependent variables, we opted not to include this outcome in the MANOVA but rather to conduct a separate set of analyses to assess group differences for this variable.

To determine whether the groups differed on the content of the lessons taught for each of the six CS standards strands, we conducted 2×4 chi-square analyses, with treatment status x number of lessons categories (0 lessons, 1-2 lessons, 3-4 lessons, 5 or more lessons). We also calculated a conservative estimate of the number of lessons for each participant and each strand, using a value of 0 for the "0 lessons" response, a value of 1 for the "1-2 lessons" response, a value of 3 for the "3-4 lessons" option, and a value of 5 for the "5 or more lessons" response. We then summed these values to arrive at an estimate of the total number of lessons taught for each participant. Finally, we used the total number of lessons taught to conduct an independent samples *t*-test to determine differences by treatment status.

Results

The data met all the assumptions for the MANCOVA analysis (independence of observations, multivariate normality, homogeneity of variance-covariance matrices, linearity [57]), so the MANCOVA was performed. The results indicated a significant effect of treatment, Wilks' lambda = 0.60, $F(3, 108) = 24.04$ $p < .001$, partial eta squared = 0.40, revealing statistically significant differences for Confidence, Experience, and Engagement, where the treatment group scored significantly higher than control on all outcomes. Tests of between-subjects effects revealed a substantial impact of treatment at the end-of-year point on Confidence (partial eta squared = 0.34), Experience (partial eta squared = 0.32), and Engagement (partial eta squared = 0.16). Means and standard deviations by treatment status are reported in Table 1 for the three outcome variables.

Table 1
Teacher outcomes by treatment status.

	Treatment		Control	
	M	*SD*	*m*	*SD*
Confidence	55.66	10.23	41.75	15.39
Experience	26.36	6.22	18.77	8.17
Engagement	20.29	3.06	18.05	3.76

Minimum-maximum score ranges: Confidence 13.00-73.00; Experience: 6.00-36.00; Engagement: 6.00-24.00

The examination of the number of lessons taught by strand revealed statistically significant differences for each of the six strands between groups, where participants randomized to treatment had higher values for the three lesson-count indicator options (1-2, 3-4, 5 or more) than control respondents (see Table 2 and Fig. 1). An independent *t*-test also revealed a statistically significant difference between groups, with the treatment group indicating a larger number of total CS lessons taught ($m = 8.78$) than the control group ($m = 5.97$) ($t = -2.01$, $df = 79$, $p = .05$). When summed, the total CS lessons taught variable revealed that treatment teachers taught an estimated 448 lessons, compared to 179 lessons taught by teachers in the control group, as illustrated in Fig. 2.

Discussion

The [REDACTED] project and this embedded study responded to a statewide need to improve teachers' CS related knowledge and pedagogical skills, and a national need for research on the impact of high-quality PD to enhance elementary teachers' ability to integrate CS into the curriculum. Its findings illustrate that participating in high-quality CS-integration PD was more effective than business-as-usual for increasing teachers' confidence to teach CS integrated lessons, perceived experiences in teaching CS integrated lessons, and perceived self-efficacy to engage students in CS integrated lessons. Teachers' perceptions were in alignment with their actions, as treatment group teachers also reported teaching more CS lessons than control group teachers for each of the six CS content area strands within the statewide CS curriculum standards, with a significance value of $p = .05$. We acknowledge that a significance value of $p < .05$ is common practice in quantitative research, though some modern researchers suggest that acceptance of $p \leq .05$, as originally proposed by Fisher [58], as appropriate for indicating strength of difference between groups [59,60] or that a significance value of .05 should be considered in the context of other equation statistics [61]. As treatment teachers reported teaching 2.5 more lessons than control teachers, we assert that this finding is *meaningfully* significant, even if its *statistical* significance is in question. We believe that the partnership between the [REDACTED] project team, the VDOE, and the non-profit CS agency supported these findings and will also promote sustainability of CS teaching and learning as it provides evidence of commitment to CS.

The findings here are notable, given that the [REDACTED] PD program was offered entirely online, and that nearly 40 % of the control group teachers indicated that they had participated in non-[REDACTED] CS PD during the study period (compared to 33 % of treatment teachers who indicated participated in non-program CS PD during the same time). The virtual format of the PD and the NIC will also likely play a significant role toward access to high-quality teacher PD as the [REDACTED] team works to scale the project beyond the treatment group and across the state, particularly in rural areas. Other features of [REDACTED], such as an integrated, multidisciplinary approach to CS instruction, have been shown to be effective in improving teacher CS confidence and competence [30,62] and in increasing the number and frequency of integrated CS lessons taught [28], findings that mirror the outcomes here and which may further promote sustainability and scaling of the [REDACTED] program.

Limitations

It is important to highlight limitations that may have affected our findings, presumably the most influential being that cohort one activities occurred during the 2021-2022 school year when COVID-19 challenges persisted in K-12 schools. Inconsistent class settings (e.g. in-person, hybrid, virtual), higher absenteeism, and continued health concerns are among these challenges, any or all of which may have impacted outcomes for either the treatment or control group. It is also possible that, despite the mandate from the state department of education to integrate CS curriculum standards beginning in 2019, conflicting priorities at the division or school level may have influenced the focus on CS teaching and learning, in a positive or negative manner. Finally, our findings are derived from participants' self-reported data, and we recognize the limitations that self-reported data may introduce. This includes limitations for the variable for number of lessons taught: the response options for this variable provided ranges that could be interpreted as arbitrary (0, 1-2, 3-4, 5 or more) and the item allowed teachers to count lessons that integrated more than one CS standard strand as multiple lessons. This may have artificially inflated the numbers of lessons taught or, inversely, limited the numbers of lessons taught for any participant who taught six or more lessons. However, in the absence of classroom observations (primarily because of persistent COVID challenges), we felt that this was an appropriate proxy for the frequency of CS lessons taught, even if not ideal.

Table 2
Teachers' implementation of CS integrated lessons.

CS SOL Strand & Group	*0 lessons*	*1-2 lessons*	*3-4 lessons*	*5 + lessons*	χ^2	*Df*	*p*
Algorithms & Programming	3	20	18	10	17.82	4	.001
Treatment	1	20	6	3			
Control							
Computing Systems	17	21	10	3	15.56	4	.004
Treatment	14	13	2	1			
Control							
Cybersecurity	16	21	10	4	15.79	4	.003
Treatment	11	16	2	1			
Control							
Data & Analysis	16	23	10	2	19.74	4	<.001
Treatment	16	10	1	3			
Control							
Impacts of Computing	18	22	8	3	17.40	4	.002
Treatment	12	16	0	2			
Control							
Networking & the Internet	19	18	10	4	16.74	4	.002
Treatment	17	9	4	0			
Control							

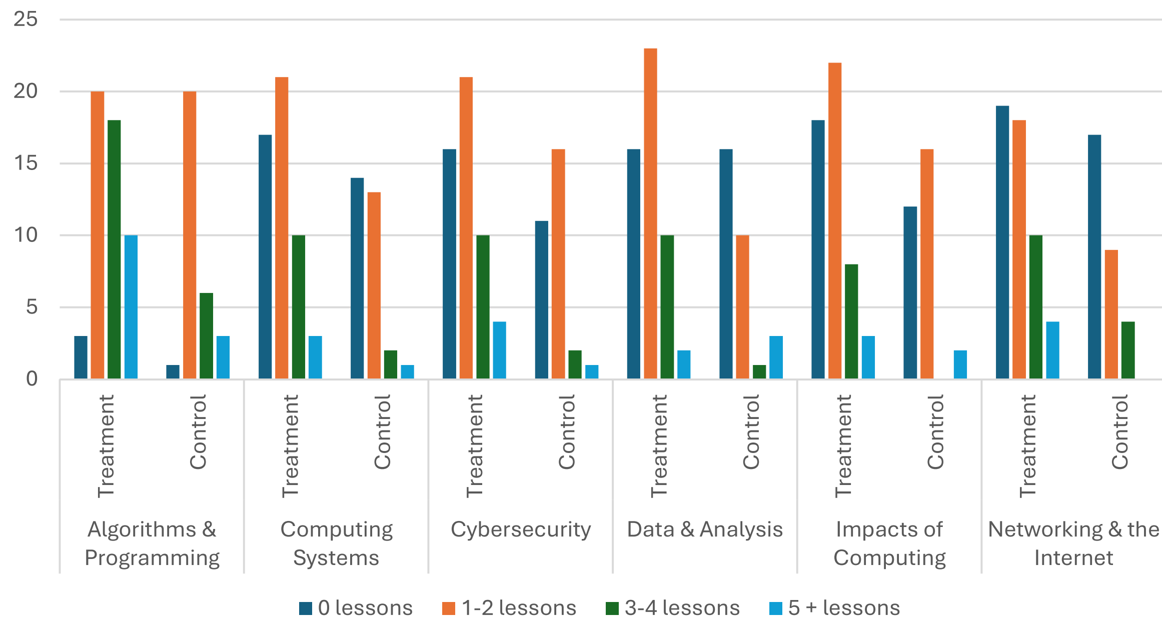

Fig. 1. Number and type of lessons taught by treatment and control teachers.

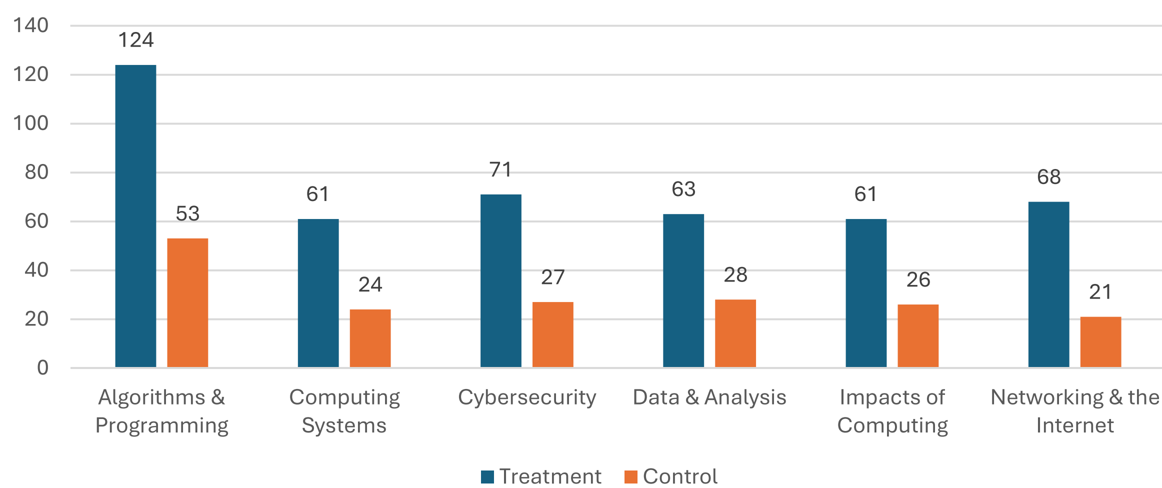

Fig. 2. Total number of lessons taught by CS strand and treatment status.

Conclusions

Because this was a true randomized controlled trial, we can infer cause, meaning that we can conclude that the [REDACTED] program positively impacted teachers' confidence in programming and teaching CS concepts, their experience integrating CS into their core curricular lessons, and their ability to engage students during CS lessons. [REDACTED] PD also statistically significantly impacted the number of CS lessons taught by participating teachers, where [REDACTED] teachers taught CS lessons at two-and-a-half times the rate of control teachers, exposing students of treatment teachers to CS content and concepts and broadening participation in CS at the elementary level. Other studies from [REDACTED] are ongoing to examine how the program impacted student affect and knowledge relative to CS, and how variations in teacher fidelity of implementation may yield differential impacts on student performance. Additional work is also underway to update [REDACTED] components to align with Virginia's revised CS curriculum standards, which were released in 2024.

CRediT authorship contribution statement

Shanan Chappell Moots: Writing – original draft, Methodology, Formal analysis, Conceptualization. **Joanna K. Garner:** Writing – review & editing, Supervision. **Joseph A. Brobst:** Writing – original draft. **Melani Loney:** Writing – original draft. **Lisa Steffian:** Writing – original draft. **Jennifer Maeng:** Data curation, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Joanna Garner reports financial support was provided by US Department of Education. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the U.S. Department of Education's

Education Innovation and Research program under Grant # U411C190032.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.caeo.2026.100353.

APPENDIX A

ARCS PD Teacher Survey

*Differences between pre-treatment and end-of-year administration points are denoted. Only items relevant to the current study are presented here.

The purpose of this survey is to understand your current thinking about computer science and elicit your feedback on your confidence, perceptions and experiences relative to computer science teaching and learning.

Confidence Subscale

Six response options: Not at all confident, Unconfident, Somewhat unconfident, Somewhat confident, Confident, Very confident

Rate your confidence with the following.

1. Programming (any language)
2. Coding in a block language (e.g. Scratch)
3. Coding in a text-based language (e.g. Python)
4. Running an "Hour of Code" event
5. Teaching programming (any language)
6. Teaching coding in a block language (e.g. Scratch)
7. Teaching coding in a text-based language (e.g. Python)

Rate your confidence integrating the following into your K-12 instruction.

1. The Virginia Computer Science standards
2. Algorithms and programming
3. Information about computer systems
4. Information about cybersecurity
5. Data and analysis
6. Information about the impacts of computing

Experience Subscale

Six response options: Very unexperienced, Unexperienced, Somewhat unexperienced, Somewhat experienced, Experienced, Very experienced

Rate your experience with the following.

1. Programming (any language)
2. Coding in a block language (e.g. Scratch)
3. Coding in a text-based language (e.g. Python)
4. Running an "Hour of Code" event
5. Teaching programming (any language)
6. Teaching coding in a block language (e.g. Scratch)
7. Teaching coding in a text-based language (e.g. Python)

Rate your experience integrating the following into your K-12 instruction.

1. The Virginia Computer Science standards
2. Algorithms and programming
3. Information about computer systems
4. Information about cybersecurity
5. Data and analysis
6. Information about the impacts of computing

Engagement Subscale

Six response options: Strongly disagree, Disagree, Somewhat disagree, Somewhat agree, Agree, Strongly agree

1. I can engage students from rural areas in computer science.
2. I can engage students from low socioeconomic backgrounds (i.e., students receiving free and reduced-price meals) in computer science.
3. I can engage students who are traditionally underrepresented in STEM (i.e., Black, Hispanic, female, receiving special education services) in computer science.
4. I can address issues of access to computer technologies for students in my school.

Computer Science Lessons Subscale (end-of-year version only)

Did you teach any lessons that explicitly targeted CS SOLs this school year?

No

Yes

(A “No” response skips the next set of items)

If yes, approximately how many lessons related to each of the following CS SOL strands did you teach? If a lesson was designed to target multiple CS SOL strands, count it for each strand.

Four response options for each strand: 0 Lessons, 1-2 lessons, 3-4 lessons, 5 or more lessons.

1. Algorithms and Programming
2. Computing Systems
3. Cybersecurity
4. Data and Analysis
5. Impacts of Computing
6. Networking and the Internet

References

[1] Code.org CSTA, Alliance ECEP. 2020 State Comput Sci Educ: Illum Disparities 2020. Retrieved from, https://advocacy.code.org/stateofcs.

[2] Gallup & Google. (2020). *Current perspectives and continuing challenges in computer science education in U.S. K-12 schools*. Retrieved from https://csedu.gallup.com/home.aspx.

[3] Bergson-Shilcock A, Taylor R. Closing Digit Ski Divide, Payoff Work Bus Econ, Natl Ski Coalit Fed Reserve Bank Atlanta 2023.

[4] Lazio R, Ford H. The US needs to prepare workers for STEM jobs: why retooling the American workforce can help technology unleash the U.S. economy. SHRM 2019. https://www.shrm.org/hr-today/news/hr-magazine/summer2019/Pages/the-u.s.-needs-to-prepareworkers-for-stem-jobs.aspx.

[5] National Science and Technology Council Committee on STEM Education. Charting a course for success: America’s strategy for STEM Education. 2018. Retrieved from, https://www.whitehouse.gov/wp-content/.../STEM-Education-Strategic-Plan-2018.pdf.

[6] Google & Gallup (2017). *Computer Science learning: closing the gap*. Retrieved serv. google.com/fh/files/misc/computer-science-learning-closing-the-gap-rural-small-town-brief.pdf.

[7] Code.org CSTA, Alliance ECEP. 2024 State Comput Sci Educ 2024. Retrieved from, https://advocacy.code.org/stateofcs.

[8] Yang D, Yang ZY, Bers MU. The efficacy of a computer science curriculum for early childhood: evidence from a randomized controlled trial in K-2 classrooms. Comput Sci Educ 2023. https://doi.org/10.1080/08993408.2023.2279198.

[9] Virginia Department of Education [VDOE]. (2021). *A review of computer science in the Commonwealth report.* HB 1885 Status Report 2021. Retrieved from https://rga.lis.virginia.gov/Published/2022/RD83/PDF.

[10] Abell SK, Abell SK. Research on science teacher knowledge. In: Lederman NG, editor. Handbook of Research on Science Education. New Jersey: Lawrence Erlbaum Associates, Inc; 2007. p. 1105–49.

[11] Van Mechelen M, Smith RC, Schaper MM, Tamashiro M, Bilstrup KE, Lunding M, Peterson MG, Iverson OS. Emerging technologies in K-12 education: A future HCI research agenda. ACM Trans Comput-Hum Interact 2023;30(3). 1145/3569897.

[12] Saglam RB, Miller V, Franqueira VNL. A systematic review of cyber security education for children. IEEE Trans Educ 2023;66(3).

[13] Kwon K, Ottenbreit-Leftwich AT, Brush TA, Jeon M, Yan G. Integration of problem-based learning in elementary computer science education: effects on computational thinking and attitudes. Educ Technol Res Dev 2021;69:2761–87. https://doi.org/10.1007/s11423-021-10034-3.

[14] Strawhacker A, Bers MU. What they learn when they learn coding: investigating cognitive domains and computer programming knowledge in young children. Educ Technol Res Dev 2019;67. https://doi.org/10.1007/s11423-018-9622-x.
[15] Studhalter UT, Jossen P, Seeli M, Tettenborn A. Tablet computers in early science education: enriching teacher-child interactions. Early Child Educ J 2024. https://doi.org/10.1007/s10643-024-01766-x.
[16] Constantin AI, Popovici AF. Curriculum integration and learning motivation of the secondary school students. J Educ Soc Multicult 2022;3(2):61–85. https://doi.org/10.2478/jesm-2022-0018.
[17] Ballard ED, Haroldson R. Analysis of computational thinking in children's literature for K-6 students: literature as a non-programming unplugged resource. J Educ Comput Res 2022;59(8):1487–516. https://doi.org/10.1177/07356331211004048.
[18] Waterman KP, Goldsmith L, Pasquale M. Integrating computational thinking into elementary science curriculum: an examination of activities that support students' Computational thinking in the service of disciplinary learning. J Sci Educ Technol 2020;29(1):53–64. https://doi.org/10.1007/s10956-019-09801-y.
[19] Schiele H, Bos-Nehles A, Delke V, Stegmaier P, Torn RJ. Interpreting the industry 4.0 future: technology, business, society and people. J Bus Strategy 2022;43(3): 157–67. https://doi.org/10.1108/JBS-08-2020-0181.
[20] National Science Board. The STEM labor force: scientists, engineers, and skilled technical workers. Natl Sci Found 2024.
[21] Bureau of Labor Statistics, US Department of Labor [BLS]. (N.D.) *Occupational Outlook Handbook, Computer and Information Technology Occupations.* Retrieved from https://www.bls.gov/ooh/computer-and-information-technology/.
[22] Roller A. 2025 Tech Job Mark Stat Outlook 2024. Retrieved from Tech Target, https://www.techtarget.com/whatis/feature/Tech-job-market-statistics-and-outlook.
[23] Heintz, F., Mannila, L., and Färnqvist, T. (2016). A review of models for introducing computational thinking, computer science and computing in K-12 education. *IEEE Frontiers in Education Conference (FIE)*, Erie, PA, 1-9, 10.1109/FIE.2016.7757410.
[24] Ni L, Bausch G, Benjamin R. Computer science teacher professional development and professional learning communities: a review of the research literature. Comput Sci Educ 2021;33(1):29–60. https://doi.org/10.1080/08993408.2021.1993666.
[25] National Academy of Engineering and National Research Council [NRC]. STEM Integration in K-12 Education: Status, Prospects, and an Agenda for Research. Washington, DC: The National Academies Press; 2014. https://doi.org/10.17226/18612.
[26] Zimpfer B. *The need to integrate computer science.* Master's theses & capstone projects. Educ Commons 2023;563. Retrieved from, https://nwcommons.nwciowa.edu/education_masters/563.
[27] Meyer R. Megan Smith thinks every child should be able to code. Atlantic 2014. Retrieved from, http://www.theatlantic.com/technology/archive/2014/10/us-cto-every-child-should-be-able-to-code/382089/.
[28] Century J, Ferris KA, Zuo H. Finding time for computer science in the elementary school day: a quasi-experimental study of a transdisciplinary problem-based learning approach. Int J STEM Educ 2020;7(20). https://doi.org/10.1186/s40594-020-00218-3.
[29] Ozturk Z, Dooley CM, Welch M. Finding the hook: computer science education in elementary contexts. J Res Technol Educ 2018;50(2). https://doi.org/10.1080/15391523.2018.1431573.
[30] Harris C. *Computational Thinking Unplugged: Comparing the Impact on Confidence and Competence from Analog and Digital Resources in Computer Science Professional Development for Elementary Teachers. Education Doctoral.* Fisher Digital Publications; 2018. Paper 374.
[31] Yadav A, Gretter S, Hambrusch S, Sands P. Expanding computer science education in schools: understanding teacher experiences and challenges. Comput Sci Educ 2017;26(4):235–54. https://doi.org/10.1080/08993408.2016.1257418.
[32] El-Hamamsy L, Chessel-Lazzarotto F, Bruno B, Didier R, Cahlikova T, Chevalier M, Parriaux G, Pellet J, Lanarès J, Zufferey J, Mondada F. A computer science and robotics integration model for primary school: evaluation of a large-scale in-service K-4 teacher-training program. Educ Inf Technol 2021;26(3):2445–75. https://doi.org/10.1007/s10639-020-10355-5.
[33] Harper-Hill K, Beamish W, Hay S, Whelan M, Kerr J, Zelenko O, Villalba C. Teacher engagement in professional learning: what makes the difference to teacher practice? Stud Contin Educ 2020;44(1):105–18. https://doi.org/10.1080/0158037X.2020.1781611.
[34] Liu, R., Maeng, J.L., Moots, S.C., & Garner, J.K. (2025). Building elementary teachers' capacity for computer science instruction through professional development: A randomized control trial. Contemporary Issues in Technology and Teacher Education, 25(1). Available at https://citejournal.org/volume-25/issue-1-25/general/building-elementary-teachers-capacity-for-computer-science-instruction-through-professional-development-a-randomized-control-trial.
[35] Lave J, Wenger E. Situated learning: Legitimate peripheral participation. Cambridge University Press; 1991. https://doi.org/10.1017/CBO9780511815355.
[36] LeMahieu P. Why NIC? Carnegie Found Adv Teach 2015. Retrieved from, https://www.carnegiefoundation.org/blog/why-a-nic/.
[37] McKay, S. (2017). Five essential building blocks for a successful networked improvement community. Retrieved from https://www.carnegiefoundation.org/blog/five-essential-building-blocks-for-a-successful-networked-improvement-community/.
[38] Meyers C, Molefe A, Dhillon S, Zhu B. The impact of eMINTS professional development on teacher instruction and student achievement. Year 3 report. Washington: DC: American Institutes for Research; 2015.
[39] Maeng JL, Bell RL, Konold T, Whitworth BA. The effect of professional development on elementary teachers' understanding and implementation of reforms-based science instruction. Fairfax, VA: VISTA; 2015. Retrieved from, http://vista.gmu.edu/.
[40] Taylor JA, Roth K, Wilson C, Stuhlsatz M, Tipton E. The effect of an analysis-of-practice, videocase-based, teacher professional development program on elementary students' science achievement. J Res Educ Eff 2017;10(2):241–71. https://doi.org/10.1080/19345747.2016.1147628.
[41] Cooper S, Rodger SH, Schep M, Stalvey RH, Dann W. Growing a K-12 community of practice. In: Proceedings of the 46th ACM Technical Symposium on Computer Science Education; 2015. p. 290–5. https://doi.org/10.1145/2676723.2677255.
[42] Kosmas P. Online sharing of knowledge among In-service teachers for professional development purposes. In: Proceedings of the 9th International Conference on Education Technology and Computers; 2017. p. 162–6. https://doi.org/10.1145/3175536.3175557.
[43] Sentance S, Humphreys S. Understanding professional learning for computing teachers from the perspective of situated learning. Comput Sci Educ 2018;28(4): 345–70. https://doi.org/10.1080/08993408.2018.1525233.
[44] McMillan JH, Schumacher S. Res Educ: Evid-Based Inq 2006. Allyn and Bacon.
[45] Bean, N., Weese, J., Feldhausen, R., & Bell, R. S. (2015). Starting from scratch: developing a pre-service teacher training program in computational thinking. Presented at the 2015 Frontiers in Education Conference (FIE) (pp. 1-8). IEEE.
[46] Washington, A. N., Grays, S., & Dasmohapatra, S. (2016). *The computer science attitude and identity survey (CSAIS): A novel tool for measuring the impact of ethnic identity in underrepresented computer science students.* Paper presented at the 2016 ASEE Annual Conference & Exposition, New Orleans, LA.
[47] Dorn B, Tew AE. Empirical validation and application of the computing attitudes survey. Comput Sci Educ 2015;25(1):1–36. https://doi.org/10.1080/08993408.2015.1014142.
[48] Rhodes CM. Validation of the culturally responsive Teaching Survey. Univers J Educ Res 2016;5(1):45–53.
[49] Siwatu KO. Preservice teachers' culturally responsive teaching self-efficacy and outcome expectancy beliefs. Teach Teach Educ 2007;23(7):1086–101.
[50] Renninger KA, Elias RuthC, Kamiya MJ, Paige JN, Youngblood RA. PD supporting CS and math integration: implications of teacher interest and confidence for workshop design. Comput Sci Educ 2024:1–30. https://doi.org/10.1080/08993408.2024.2433334.
[51] Darling-Hammond L, Hyler ME, Gardner M. Effective Teacher Professional Development. Learning Policy Institute; 2017.
[52] Dooley K. Towards a holistic model for the diffusion of educational technologies: an integrative review of educational innovation studies. J Educ Technol Soc 1999;2(4):35–45.
[53] Virginia Department of Education [VDOE]. (2017). *Comput Sci Stand Learn Curric Framew.* Retrieved from https://www.doe.virginia.gov/teaching-learning-assessment/instruction.
[54] CodeVA. CS Integr Toolkit 2024 [Website], https://curriculum.codevirginia.org/library/browse/integration-toolkit. CC BY-SA-NC 4.0.
[55] Byrd, C. (2016). Does culturally relevant teaching work? An examination from student perspectives. *SAGE Open*, 6. https://doi.org/10.1177/2158244016660744.
[56] Cruse SK. Bringing Cult Classr: Impact Cult Relev Pedagogy Stud Engagem Self-Effic Soc Stud, (Dr diss) 2019. Retrieved from, https://scholarcommons.sc.edu/etd/5184.
[57] Field A. Discovering statistics using SPSS. SAGE publications; 2009.
[58] Fisher RA. Statistical methods for research workers. Oliver & Boyd; 1925.
[59] Andrade C. The P value and statistical significance: misunderstandings, explanations, challenges, and alternatives. Indian J Psychol Med 2019;41:210–5. https://doi.org/10.4103/ijpsym.ijpsym_193_19.
[60] Kennedy-Shaffer L. Before $p < 0.05$ to beyond $p < 0.05$: using history to contextualize p-values and significance testing. Am Stat 2019;73:82–90. https://doi.org/10.1080/00031305.2018.1537891.
[61] Silvey B. Comments from the editor: how big are your P values? Update: Appl Res Music Educ 2021;40:3–4. https://doi.org/10.1177/87551233211043843.
[62] Garrett, R. (n.d.) *Teach Tech Ky Eval Impact Comput Sci Focus Teach Micro-Credentials.* Retrieved from https://www.air.org/project/teach-tech-kentucky-evaluating-impact-computer-science-focused-teacher-micro-credentials.