IDENTIFYING IMPLEMENTATION CHALLENGES FOR A NEW COMPUTER SCIENCE CURRICULUM IN RURAL WESTERN REGIONS OF THE UNITED STATES

Astrid K Northrup, PE
University of Wyoming & Northwest College, USA
E-mail: astrid.northrup@nwc.edu

Andrea C. Burrows, Timothy F. Slater
University of Wyoming, USA
E-mail: andrea.burrows@uwyo.edu, Timothy.Slater@uwyo.edu

Abstract

Like much of the world, the United States is rapidly implementing the teaching of computer science into both primary and secondary school curricula. Uncovering what challenges U.S. schools in general–and rural U.S. schools in the unique environment of more mountainous regions of the U.S. in particular–face in implementing new curricula is not well established in the scholarly literature base. If reform advocates are able to anticipate implementation challenges, they might be more effective in facilitating needed changes. In response, the overarching research question is addressed: What are the challenges of delivering multi-age computer science in the rural, mountain regions of the western U.S.? This two-phase research project first identifies the anticipated challenges to implementing the curriculum by curriculum designers (Phase 1), and then goes on to compare those anticipated challenges to those identified by classroom teachers after completing a teacher training program (Phase 2). Thirteen teachers completed the training program in May and July of 2021, and five were selected to be interviewed. Transcripts of the interviews were analyzed in open, axial, and selective coding to identify recurrent and dominant themes. First, a qualitative methodology through the lens of the constructivist theory was used. Then, conventional narrative inquiry methods were employed to investigate the narratives using thematic analysis. The Phase 1 concerns of adhering to curriculum standards and using appropriate programming languages were contrasted to the Phase 2 themes of the future importance of coding for all students, confidence in pedagogy, the difficulty of coding, and issues of approval and safety. The results of this study serve as a bridge between the mandates created by education leaders and the actual experiences of the participating teachers tasked with delivering the curriculum.

Keywords: *computer science education, teacher education, fidelity of implementation, STEM, professional development*

Introduction

A rural, mountainous region in the western United States might seem like an unusual place to undertake the United States' first attempt at a comprehensive computer science (CS) initiative curriculum inclusion, initiative at both the primary and secondary school levels (Graf, 2021). The region has a unique energy-based economy, the smallest population of any state in the U.S., and in many locations, unstable internet access. However, in 2019, the state

ISSN 1822-7864 (Print) ISSN 2538-7111 (Online) https://doi.org/10.33225/pec/22.80.353

government authority included a goal of teaching "the use and understanding of computer science" for schools in the region.

This goal for widespread teaching students at all levels foundational concepts of CS is based upon a philosophical notion that 21st Century student requires fundamental knowledge of computers and how computers can be used to solve broad based problems (National Science Foundation, 2014; Northrup et al., 2021). Despite being a laudable goal for schools, little is known about how best to implement CS curriculum widely.

Research Problem

If reform advocates are able to anticipate implementation challenges at the school building level, they might be more effective in facilitating needed changes. As a result, this research aimed to characterize teachers' perceptions of challenges they are about to encounter in their role in delivering CS curriculum in schools. The study-participants were purposefully chosen from the first cohort of teachers who completed a targeted professional development (PD) training program representing teachers from a variety of student ages, subject areas, levels of educational attainment, years of teaching experience, and gender. In a two-phase study, the research team examined the challenges anticipated to delivering CS education and compared the anticipated challenges to those enumerated by the study-participants after completing the training program.

Research Focus

This study was conducted at a small, public, open-enrollment, two-year college in a rural, mountainous region of the western U.S. The college implemented a one-year, 15-undergraduate-credit CS training Skills Certificate in June 2020 using a cohort model. A faculty team designed this training program for licensed and certified teachers of all age bands and disciplines. The first cohort of 13 certified teachers completed the training program in May and July 2021.

Research Aim and Research Questions

For this two-phase research project, the research team has first studied the anticipated challenges to implementing the curriculum by curriculum designers (Phase 1), and then has compared those anticipated challenges to those identified by classroom teachers after completing a teacher training program (Phase 2). The specific, examined gap is the teachers' role in delivering integrated CS education, studied systematically to examine connections made by teachers between this PD and the learning outcomes. Given that curricular reform is an arduous process, a better understanding helps reformers anticipate what teachers need to succeed.

The focus of this research is the overarching question: What are the challenges of delivering multi-age computer science in the rural, mountains regions of the western U.S.?

Research Methodology

General Background

Qualitative methodology was used through the lens of the constructivist theory to pursue the core research question driving this study. Crotty (2020) describes constructivism as the making of meaning (p. 42). In this study of implementation challenges for new CS education, the teachers make meaning from their experiences in the CS training certificate program by taking what they learned and implementing it in the classroom.

Merriam and Tisdell (2015) described the process used in this research, which includes a semi-structured interview using open-ended questions and allowing participants to guide the discussion. In this way, new ideas on the topic emerge as participants' worldview can inform the discussion. Then, narrative inquiry was used to explore the interview transcripts using thematic analysis (TA).

The researchers adopted a two-phase study to examine the anticipated challenges in delivering integrated CS education when developing this CS training program (identified as Phase 1). The study then compares anticipated challenges to those enumerated by the participants after completing the training program (identified as Phase 2).

Sample

Interviewees were purposefully selected from a population of only 13 teachers completing the program. Due to the small population and sample size, it is challenging to generalize the results to an international audience, or perhaps even a nation-sized audience given the geographic size and demographic diversity present. However, this is a limitation of qualitative research in general rather than this study specifically (Wolcott, 1990). While the ability to generalize is a major consideration in quantitative research (Fallon, 2019), it figures less prominently in qualitative work. Of the thirteen teachers who completed the training program in May and July of 2021, five were interviewed for this study.

All cohort members lived and worked in rural areas. They were chosen for this study to represent various years of teaching experience, genders, education levels, grade bands, and specialization areas.

Purposeful sampling was used to "discover, understand, and gain insight" (Merriam, 1988, p. 77). Four of the study participants teach in non-STEM areas and one teaches science. As a group, they represent the cross-discipline design of the statewide CS initiative. Table 1 summarizes the characteristics of the participants.

Table 1
Characteristics of Study Participants

Name	Gender	Teaching Experience (years)	Education	Grade Band	Specialization
Jay	M	21-25	Bachelor's	ES	None/all
Tia	F	16-20	Master's	HS	English/SS
Sam	M	21-25	Bachelor's	HS	Business/CS
Nan	F	11-15	Master's	MS	Science
Sol	M	0-5	Bachelor's	ES/MS	SS/CS

Note: M: male, F: female, CS: computer science, SS: social science,
ES: elementary school, MS: middle school, HS: high school.

The project obtained human subject approval from the researchers' institution, and all participants signed an informed consent. All participants are identified with pseudonyms.

Instrument and Procedures

Narrative Inquiry: All participants are educators with extensive classroom experience. Narrative inquiry captures the experience of these participants as both teachers and students by allowing them to tell their story, then purifying the story into a cohesive narrative. It is a "method of knowing" based on lived experience (Clandinin, 2006). This method allows the interviewees' stories to emerge from open-ended interview questions (Connelly & Clandinin, 1990). Narrative inquiry allows the greatest latitude in understanding this experience and identifying challenges from each teacher's perspective. From this basis of experience, patterns were identified to clarify their lived experiences into themes.

Thematic Analysis (TA): TA was chosen to analyze the teachers' narratives due to its flexibility, accessibility, and systematic approach to coding and analyzing qualitative data. Braun and Clarke (2006, 2012, and 2020) provided guidelines and context for its use in qualitative analysis. Specifically, they posited that TA is most appropriately used to identify participants' lived experience patterns.

Kiger and Varpio (2020) expanded on the work by Braun and Clarke, offering careful definitions for *TA* and *theme*. TA is valid within post-positivist, constructivist, and critical realist research approaches without being tied to a particular paradigmatic orientation.

Braun and Clarke (2012) and Kiger and Varpio (2020) recommended the six-phase approach to the process of TA used in this research study, presented in Table 2.

Table 2
Six-phase Approach to Thematic Analysis

Phase	Description
1	Familiarizing oneself with data
2	Generating initial codes
3	Searching for themes
4	Reviewing potential themes
5	Defining and naming themes
6	Producing the report

Credibility and Trustworthiness (Reliability and Validity): It is important to articulate the correlation between trustworthiness and both reliability and validity in establishing the credibility of this study. Krefting (1991) described a conceptual model for evaluating the reliability and validity of qualitative research. As much of what is known about these concepts derives from *quantitative* research, Krefting argued that one cannot apply these concepts directly to qualitative data. For example, external validity is the ability to generalize from a sample to a population. However, the ability to generalize is a major consideration in quantitative research but figures less prominently in qualitative work. For this reason, external validity is not applicable for many qualitative research studies.

However, trustworthiness is relevant to both qualitative and quantitative research, although it takes different forms. Guba (1981) identified four aspects of trustworthiness: truth value, applicability, consistency, and neutrality. The truth value is synonymous with credibility. When study participants possess special insight into the studied subject, they establish credibility. Applicability is similar to generalizability. Similarly, as it refers to transferability, consistency is a key objective of quantitative research. Due to the nongeneralizable nature of a qualitative inquiry, consistency is less important. Neutrality, or freedom from bias, is based upon conditions of the research and the informants (Guba, 1981). For this study, truth value (credibility) and neutrality are the basis of trustworthiness and rest on the expertise and lived experience of the research participants.

As credentialed, experienced teachers, all participants have special insight into issues of classroom instruction. Pidgeon et al. (2004) argued that this special insight is a criterion for trustworthiness. In qualitative methodologies, validity and reliability are often established through external auditing and triangulation. However, given the nature of the participants, trustworthiness, rather than validity and reliability, is established by the truth value and neutrality.

Interview Questions: Interviews were conducted with each participant between June 1 and August 2, 2021, using the Zoom online videoconferencing platform. In each interview, the participants were asked six open-ended questions. The interview questions are provided in the Appendix A.

Procedures: The anticipated challenges were delineated by the faculty team and then compared to the challenges identified by the interviewees. The following process was performed to develop codes from the interview data and organize the codes into themes.

1. The first author interviewed each participant. The interviews were conducted between June 1 and August 2, 2021, after each participant had completed the CS training program prior to their subsequent teaching of CS.
2. The interviews were conducted using the Zoom videoconferencing platform. All interviews were conducted with only the interviewee and interviewer present.
3. The interviews were recorded using the Kaltura video platform and were stored privately on the college's media site.
4. The recorded interviews were transcribed using the Kaltura video platform.
5. The interview transcripts were analyzed for themes using the six-phase approach to TA described by Braun and Clarke (2012) and Kiger and Varpio (2020).
6. The Delve software tool for analyzing qualitative data (Twenty to Nine LLC, 2021) was employed to organize the codes into themes and identify patterns among interviewees.

Data Analysis

This two-phase research project first identifies the anticipated challenges to implementing the curriculum from personal conversations with curriculum designers (Phase 1). Then researchers analyzed unedited transcripts of interviews conducted with teachers who completed the PD (Phase 2). By searching the interview transcripts for words and phrases in the interviewees' own words, a combination of descriptive coding and in vivo coding was used (Ho, 2020; Fallon, 2019). Delve software (Twenty to Nine LLC, 2021) facilitated the six-phase approach by Braun and Clarke (2012) and Kiger and Varpio (2020). From the codebook developed, four themes emerged. These themes were then compared to the challenges anticipated by the faculty team as they worked from government provided *WDE Computer Science Content and Performance Standards* reform documents (2021). Then, themes were identified to provide important insights concerning the research question (Kiger & Varpio, 2020). These codes were formed using the iterative process of coding, analytical writing, and categorizing. The distribution of codes provides an amalgamated overview of the topics identified by the participants and forms the basis of the TA.

Research Results

From the data analysis, of the interview transcripts, four themes emerged. These themes are tabulated below in Table 3.

Table 3
Frequency of Qualitative Codes in Cohort Member Interviews in Descending Order

Code	Noted by #	Count
CS is the future	5	29
Confidence in pedagogy	5	19
Coding is difficult	5	15
Approval and safety	4	9

Three of the four themes were mentioned by all five interviewees; the fourth was mentioned by four interviewees. These themes were then compared to the Anticipated Challenges found in Phase 1.

Phase 1: Anticipated Challenges

Phase 1 uncovered two challenges anticipated by the faculty team in integrating CS into existing curricula. These challenges are described below (Dechert and Floyd, personal communication, December 2, 2019).

Anticipated Challenge 1: Teaching to State Department of Education Standards. The faculty team developed the CS training program curriculum based upon the seven practices delineated in the State Computer Science Content and Performance *Standards* (2021). These practices are presented in Table 4.

Table 4
State Department of Education Computer Science Content and Performance Standards

Practice	Description
1	Fostering an inclusive computing culture
2	Collaborating around computing
3	Recognizing and defining computational problems
4	Developing and using abstractions
5	Creating computational artifacts
6	Testing and refining computational artifacts
7	Communicating about computing

Because the content and performance standards are broader than the two-year college's traditional CS offerings, two new professional development training courses (Application Development and Social Media for Teachers) were created; and one course (Robotics) was modified from an existing catalog course. Combining these courses with two existing catalog courses (Introduction to CS and CS I) allowed the faculty team to round out the curriculum and address each standard in a meaningful way. Table 5 delineates the connection between each course and the Content and Performance standards addressed.

Table 5
Standards Addressed by Each Course of the CS Training Program

Course	Standards
Intro to CS	3 4 5 and 6
CS I	3 4 5 and 6
Robotics	2 and 3
Application Dev.	5 and 6
Soc. Media for Teachers	1 2 and 7

Note: (WDE, 2021; Northrup et al., 2021)

Anticipated Challenge 2: Choosing Appropriate Programming Development Environments and Languages. As with designing any CS course, a core question is which languages and programming development environments are best used to teach structured programming. The two existing catalog courses – Introduction to CS and CS I – were used to teach structured programming in Matrix Laboratory (MATLAB) and microprocessor without interlocked pipeline stages (MIPS) Assembler and Runtime Simulator (MARS/MIPS) development environments. Before offering the CS endorsement, the audience for these courses were primarily CS majors, so the languages used to teach structured programming were oriented toward solving mathematical problems. The faculty team debated introducing numerous languages and ultimately decided to continue using MATLAB and MARS/MIPS in Introduction to CS and CS I. It was further decided to use Python and Scratch in other courses and evaluate the use of these languages after completing the first cohort.

Phase 2: Identified Challenges

Phase 2 examines the challenges identified by participants to the delivery of integrated CS education based upon their experiences in the computer science training program. The themes found in Phase 2 are presented below; a summary of the interviews is provided in Appendix B.

Theme 1: Computer Science is the Future; If We Do Not Do This, We Will Fail Students. The first major theme that emerged from this research study was the teachers' conviction that CS is the future. All participants noted this theme, which was mentioned 29 times in the interviews. This theme was noted between three and nine times by each participant and did not correlate with a particular grade band. The interviewees considered CS a necessary element of the curriculum for all students. All participants cited examples of technical and nontechnical careers requiring CS knowledge. Further, they cited life skills requiring computer knowledge, such as using computer-based scheduling systems, navigating with a GPS, and even ordering food with computer applications. As middle school science teacher Nan said, "The technology piece? There's no way around it. We need to bring back innovation and creativity, and if we are not learning it now, we are behind already."

Theme 2: Confidence in Required Pedagogy. The participants all noted that a required pedagogical shift is required to teach CS. In the traditional classroom, the teacher is the subject expert, and the students learn from the teacher. With CS, the students may know more than the teacher, and the teacher must accept students' expertise. Jay noted, "[We] are teaching students to live in a world that they probably understand better than we do," and learning along with the students is a necessary part of teaching CS.

Conversely, the teachers noted that teaching and learning in an egalitarian classroom engender confidence in the students. When students are content experts in a particular area, they teach that content to their fellow learners, including the teacher. This act of teaching and learning generates confidence in the student expert.

Theme 3: Coding Is Difficult for this Population. At the college level, coding is generally taught by and to computer scientists and engineers. When STEM students are taught to code, they are generally familiar with the concepts of computational thinking (CT) and encounter few difficulties with basic conceptualizations.

The participants of this study had a different experience. Although the participants are successful academics and expert learners, the basic ideas of CT and coding were foreign to them. Each participant discussed the difficulty of learning foundational concepts, describing the act of writing computer programs as "daunting", "mind-boggling", and "difficult". They found themselves out of their element with CT and were surprised by the degree of explicit instruction required when coding. Tia equated coding to giving clear and straightforward instructions to a person. She said, "You can't tell a robot or a teenage boy to mow the lawn; you have to be very, very specific [and say] if you hit a tree back up and go a little bit around the tree and then come back."

Theme 4: Approval and Safety. Four of the participants cited concerns about administrative approval and safety. These concepts are interrelated, as administration and trustees are cautious about allowing internet use in schools due to security issues. These stakeholders are uneasy about the presence of danger to students who use the internet. While the participants acknowledged the risk, they also felt the benefit of guided internet use would outweigh these concerns. From an instructional viewpoint, the availability of excellent resources should be balanced against the potential safety issues to provide students with a rich educational experience, including internet use. Sam suggested that this fear is a product of anecdotal experience. He said, "I interviewed my superintendent and my principal in general about CS in the classroom; [they had] preconceived notions that 'there ain't no way I'm doing that' because one time in class one person called another person a name, and it goes bad." Sam allayed their fears by describing ways to monitor and prevent bullying, concluding that internet use can be safe when properly monitored and guided.

Comparison of Phase 1 and Phase 2:

This research project endeavors to characterize teachers' perceptions of their role in delivering integrated CS education. A specific focus was to compare the challenges anticipated by the faculty team while they developed the curriculum to the challenges identified by the teachers who completed the CS endorsement.

The anticipated and identified challenges were compared and contrasted, then aggregated and interpreted. Finally, the results were analyzed.

On initial inspection, there were no significant similarities between anticipated and identified challenges. The faculty team anticipated challenges of teaching to all WDE standards and choosing appropriate programming languages. The participants identified preparing students for the future, confidence in pedagogy, difficulty learning to code, and administrative approval as challenges. Upon close study, a similarity was uncovered between anticipated Challenge 2: choosing appropriate programming languages and identified Challenge 3: coding is difficult. Identified challenge 3 was mentioned by every participant and was the most frequently mentioned theme. No participant explicitly mentioned the programming languages or environments, but inspection of the transcripts revealed a major connection related to universal difficulty in Introduction to CS and CS I, with writing code as the most common theme. This was not mentioned with the other classes, where different programming languages were used. It follows that the programming languages themselves were challenging for participants.

Since the faculty team had chosen to use these two existing classes to teach structured programming, the participants had been taught the same programming languages (MATLAB and MARS/MIPS) as the faculty team had used in the past to teach CS majors. These languages are very effective for teaching CS students but are not particularly accessible to novice programmers. Moreover, MARS/MIPS is a machine-level language, and is notoriously difficult to learn. Participants' concerns with the difficulty of programming may be partially attributable to using these languages.

In Robotics and Application Development, the teachers used a variety of programming languages, including Python, Scratch, and a Lego Mindstorm language based on the C programming language. These languages are much more accessible, high-level languages, and participants did not mention difficulty in these classes. This further supports that MATLAB and MARS/MIPS programming languages were the actual challenges identified in Theme 3.

As a result of this finding, the faculty team will modify the choice of languages employed to teach structured programming and will employ more accessible programming environments for future CS training program offerings. Future participants will then be interviewed to determine whether this challenge has been resolved.

Discussion

Literature Review

A review of the literature has identified a need to study integrating CS education across curriculum in a systematic manner. This study elucidates this need.

In this literature review, the reader's attention is directed to this gap, beginning with the current state of CS education integration, and leading to the current PD trends for CS teachers. Finally, the gap has been identified as a disconnect in the systematic study between PD and learning outcomes; specifically, the relationship between challenges anticipated by the faculty development team compared to the challenges identified by the cohort members.

Integrating Computer Science Education/Computational Thinking into Discrete Subject Areas: Numerous authors have written about teaching CS in an integrated manner. Foremost among them, Wing (2006, 2011) is a foundational researcher in the field of CT as it applies in K-12 education. Wing defined CT as "a process that involves solving problems, designing systems, and understanding human behavior, based on the fundamental concepts of computer science." Although CS encompasses CT and coding, the three are not synonymous. While CT permeates human culture, in the past, it was the purview of scientists and engineers. However, Wing argued that it is a foundational skill on par with reading, writing, and arithmetic, required by everyone in the 21st century.

Building on Wing's research, Webb et al. (2017) identified CT and CS curriculum design as a key area for future research that is important from the perspective of curriculum theory and is a necessary element for student empowerment. Building on Wing's foundational work, Barr (2011) made a connection between problem-solving and CS. More recently, Borowczak and Burrows (2019) refined this thinking by defining CS as "the science of problem solving within a computational context" and framing CT as a connection between CS and the visible practice of computer programming.

Fessakis and Prantsoudi (2019) posited that the teachers' perceptions and misperceptions about teaching CT require future study. Teaching CT across the curriculum rather than as a discrete subject area requires a pedagogical shift to using CS theory to teach the more general ideas of CT, rather than as an end goal of a discrete subject (Sentance & Csizmadia, 2017). Pedagogy for teaching CT includes scaffolding programming tasks and collaborative learning in addition to the traditional methods of teaching coding. In this way, CT is the goal of education, and coding is the primary tool for teaching CT. Although Wing has carefully defined CT and its relationship to CS and coding, the terms are often not carefully defined or differentiated in common use or within the academic literature (Grover & Pea, 2013).

Recently, initiatives have been undertaken to provide students with CT opportunities, and *Computer Science For All* (CS4All) is one such initiative. By examining the core purposes envisioned by different stakeholders, Vogel et al. (2017) identified seven core required principles for effective universal CS education, including pragmatic CS issues (workforce, citizenship, and innovation) and larger issues more closely associated with CT and social justice (equity, literacy, school reform, and personal fulfillment and fun). Webb (2017), Hubwieser et al. (2014), Hodhod et al. (2016), and Gray et al. (2015) have proposed new ways to teach CS, CT, and coding based on different principles of curriculum design. These designs share core principles of creating sustainable models based on established teaching practices and pedagogy, differing mainly in the definitions of the key concepts of CT, CS, and coding and the relative importance of these core principles.

A review of the literature reveals that the terms "computational thinking," "computer science," and "coding" are often conflated or used interchangeably. While these terms are intricately related, they are not identical (Borowczak & Burrows, 2019). Regardless of the term used, there is a great deal of support in teaching them across disciplines rather than as discrete subjects (Burrows & Slater, 2015). CS education in our state will be taught by STEM and non-STEM teachers alike; thus, our state's model is based on teaching CS in an integrated environment, consistent with the WDE Computer Science Content and Performance Standards (2021).

K-12 Teachers' Perceptions and Experiences in Teaching Computer Science: Preparing teachers to teach CS/CT is another topic in the literature. The researchers delved into this topic, as it is important for an integrated model of CS education. Yadav et al. (2016) discussed the current trend of educating noncomputer scientists to teach CS and focused on these teachers' perspectives as they deliver CS curriculum, their lack of adequate CS training, and the isolation they feel from often working alone. Yadav and colleagues are not the only authors concerned about this issue.

Astrachan and Briggs (2012) addressed not only coding but also the creative and intellectual aspects of CS. In this way, they referred more to CT than CS. ChanLin (2007) focused on teachers' perceptions of the importance of integrating CS into their classrooms, whereas Goode et al. (2014) studied the role of teachers in teaching CS. This research study explored the applicability of PD models and asserted that curriculum alone is not enough: Teachers require well-constructed PD and support to teach CS effectively. In this way, Goode et al. separated and related the knowledge of CS from the practice of teaching CT. In follow-up research, Goode et al. (2014) set up the connection between educational research related to PD for CS and CT teachers, describing the Exploring Computer Science PD model.

Gray et al. (2015) described statewide PD models for teacher preparation for CS teaching in Alabama. The project, entitled CS4Alabama, is a year-long PD using a teacher leader model of mentoring new CS teachers. Although this report predates our state's model, the core principles are similar in most cases.

Along this line of research, Lee et al. (2020) made connections between established STEM subject areas and CS integration, describing a model for in-service teacher PD that is valuable for integrating CS by making connections between STEM fields and other areas of study. Similarly, Borowczak and Burrows (2019) studied NetLogo as a platform for teacher PD for integrating CS into the curriculum. The results revealed that classroom applications were limited to topics addressed in the PD.

In summary, the relationship between CS, CT, and coding is a current focus in the academic literature. Integrating CS and CT into new and existing curricula has been examined using different approaches and different coding platforms. From this work, a view of connecting CS, CT, and coding through effective PD is emerging and new areas for potential study are being revealed.

Literature Gap: While it is apparent that CS, CT, and coding have been studied extensively from the

viewpoint of teachers' PD and the classroom, the following gap in the literature was identified. In applying the systematic study of CS education as it is integrated across discrete subjects, there is a disconnect between traditional teacher preparation, which focuses on one or two distinct disciplines, and integrated STEM (iSTEM) education. Hewson (2007) and Burrows and Slater (2015) pointed to this gap, stating that integration is crucial to teaching iSTEM using a contemporary approach.

Although PD among STEM teachers has been studied and connected to national standards, this gap exists in systematically studying PD, especially regarding the connection between PD and learning outcomes (Jackson et al, 2020). In another reference to this gap, Hubwieser et al. (2014) stated that there is still much to be done in when CS is integrated across discrete subject areas. By connecting the work by Hubwieser et al. with that by Hewson (2007) and Burrows and Slater (2015), the gap in the literature is highlighted, indicating a need to study integrating CS across a curriculum systematically. This gap is targeted in this research project.

Contextualization of Research

In this state, the Department of Education provides content and performance standards for guidance and direction to school districts; in turn, the districts are responsible for implementing education for all students. Certified teachers serve as the link between the standards and implementation.

The state studied has resolved to rely on certified teachers to deliver universal CS education instead of relying on content specialists, or even STEM teachers exclusively (WDE, 2021). The faculty development team must recognize that the PD participants do not have a background in structured programming, CS or CT, and the coursework must be designed to take this into account. For these reasons, programming language and environments that are appropriate for CS and engineering majors may not be appropriate for participants in a CS training program.

Further, the state studied has also resolved to teach CT across the curriculum rather than as a discrete subject area. This will require classroom instruction based on the more general ideas of CT, integrated through subjects, rather than focusing on CT as an end goal of a discrete subject (Sentance & Csizmadia, 2017). This is expressed in the Content and Performance Standards developed by the state's Department of Education (2021).

Beginning in the fall semester of 2022, all schools within the state will be required to provide CS education (WDE, 2021). As CS delivery becomes more widespread, training programs must remain responsive and teach appropriate languages and environments to bridge the gap for teachers who may have no formal training in CS.

However, by relating anticipated and identified challenges to the academic literature on CS education, this study may inform school districts, states, or other administrative units regarding the design of universal compulsory CS education. Further, the study will inform the faculty team when choosing programming environments for the future.

Conclusions and Implications

As schools in this rural state approach the implementation date for universal CS education integrated across the curriculum, providing PD to certified teachers to teach CS has begun. The regional government's decision to create training programs for all certified teachers to teach CS within their grade bands, rather than rely on content experts or STEM teachers, requires that PD be carefully constructed to meet their needs.

At the two-year college studied, a faculty development team was empaneled to create a CS training program to qualify certified teachers to teach CS. The team anticipated challenges (Phase 1) that the teachers would experience. These anticipated challenges were compared to the identified challenges (Phase 2) that the teachers encountered while completing the training program.

The main challenge, both anticipated and identified, was in regard to the choice of programming languages and environments used to teach structured programming. The languages and environments chosen by the faculty development team had been used successfully to teach structured programming to CS and engineering majors. Through the training program, the team learned that these languages and environments were not appropriate for teachers who have no background in structured programming.

In order to deliver curriculum that remains appropriate for the times, teachers must understand foundational structured programming. The PD must include not only applications for immediate classroom use, but structured programming to teach foundational CS skills. Choosing more accessible programming languages and environments is key to success. For this reason, PYTHON will replace MATLAB and MARS/MIPS to teach structured programming, beginning in the 2022 school year.

CS is evolving quickly, so teaching CS must remain responsive to change. CS training programs must provide foundations as well as applications in order to prepare teachers to understand the foundations of CS and CT and remain responsive in a rapidly changing classroom. Continuing research and innovation will be needed for the state's CS education to remain relevant. This is an evolving field of study and continuing research will be required to continually provide students with meaningful CS education.

This research study is the second of a three-study research project and serves as a bridge between the first and third studies. The first study is archival research regarding K-12 CS education curriculum in the targeted region. The third study will investigate the differences between expectations and reality for the teachers' experiences as they complete the CS training program and implement the material from the program into their classrooms. These three research studies form a body of work that comprehensively describes the experience of delivering CS education in the targeted region from the viewpoint of the teachers who have completed the CS training in May and July 2021 by addressing a current literature gap of studying educational programs in a systematic way. Recognizing the differences between CS students and teachers seeking CS endorsements will lead to different choices of programming design environments and languages for these two distinct groups. Finally, future research studies can focus on integrating CS education across discrete subject areas, as suggested by the literature gap studied in this research project. By recognizing a disconnect between traditional teacher preparation, which focuses on one or two distinct disciplines, and STEM education, which requires integration across disciplines, improvements may be made to PD programs in CS.

As universal CS education is implemented, changing programming languages for the CS training program is a current task. This change will be undertaken in the spring 2022 semester for implementation beginning in the summer 2022 semester for the third cohort.

Declaration of Interest

Authors declare no competing interest.

References

Astrachan, O., & Briggs, A. (2012). The CS principles project. *ACM Inroads*, *3*(2), 38–42. https://doi.org/10.1145/2189835.2189849

Borowczak, M., & Burrows, A. C. (2019). Ants go marching–Integrating computer science into teacher professional development with NetLogo. *Education Sciences*, *9*(1), 66. https://doi.org/10.3390/educsci9010066

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

Braun, V., & Clarke, V. (2012). Thematic analysis. In H. E. Cooper, P. M. Camic, D. L. Long, A. T. Panter, D. E. Rindskopf, & K. J. Sher (Eds.), *APA handbook of research methods in psychology, vol. 2. Research designs: Quantitative, qualitative, neuropsychological, and biological* (pp. 57–71). American Psychological Association. https://doi.org/10.1037/13620-004

Braun, V., & Clarke, V. (2021). One size fits all? What counts as quality practice in (reflexive) thematic analysis? *Qualitative Research in Psychology*, *18*(3), 328-352. https://doi.org/10.1080/14780887.2020.1769238

Burrows, A., & Slater, T. (2015). A proposed integrated STEM framework for contemporary teacher preparation. *Teacher Education and Practice*, *28*(2/3), 318–330. https://www.academia.edu/23424414/A_Proposed_Integrated_STEM_Framework_for_Contemporary_Teacher_Preparation

ChanLin, L. J. (2007). Perceived importance and manageability of teachers toward the factors of integrating computer technology into classrooms. *Innovations in Education and Teaching International*, *44*(1), 45–55. https://doi.org/10.1080/14703290601090390

Connelly, F. M., & Clandinin, D. J. (1990). Stories of experience and narrative inquiry. *Educational Researcher*, *19*(5), 2–14. https://doi.org/10.3102/0013189X019005002

Clandinin, D. J. (2006). Narrative inquiry: A methodology for studying lived experience. *Research Studies in Music Education*, *27*(1), 44–54. https://doi.org/10.1177/1321103X060270010301

Crotty, M. (2020). Constructionism: the making of meaning. In M. Crotty (Ed.), *The foundations of social research: Meaning and perspective in the research process* (1st ed., pp. 42-66). Routledge. https://doi.org/10.4324/9781003115700

Fallon, M. (2019). *Writing up quantitative research in the social and behavioral sciences*. Brill.

Fessakis, G., & Prantsoudi, S. (2019). Computer science teachers' perceptions, beliefs and attitudes on computational thinking in Greece. *Informatics in Education*, *18*(2), 227–258.

Goode, J., Margolis, J., & Chapman, G. (2014, March). Curriculum is not enough: The educational theory and research foundation of the exploring computer science professional development model. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (pp. 493–498). https://doi.org/10.1145/2538862.2538948

Graf, D. (2021, April 21). WySLICE program introduced to educators across the state. *Wyoming News Now*. https://www.wyomingnewsnow.tv/2021/04/21/wyslice-program-introduced-to-educators-across-the-state/

Gray, J., Haynie, K., Packman, S., Boehm, M., Crawford, C., & Muralidhar, D. (2015, February). A mid-project report on a statewide professional development model for CS principles. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (pp. 380–385). https://doi.org/10.1145/2676723.2677306

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational researcher*, *42*(1), 38–43. https://doi.org/10.3102/0013189X12463051

Guba, E. G. (1981). Criteria for assessing the trustworthiness of naturalistic inquiries. *ECTJ*, *29*(2), 75–91. https://doi.org/10.1007/BF02766777

Hewson, P. W. (2007). Teacher professional development in science. In S.K. Abell, K. Appleton, & D. Hanuscin (Eds.), *Handbook of Research on Science Education* (1st ed., pp.1179–1203). Routledge. https://doi.org/10.4324/9780203824696

Ho, L.-Y. (2020). *How to analyze qualitative data using Delve.* https://ocean.sagepub.com/blog/tools-and-tech/how-to-analyze-qualitative-data-using-delve

Hodhod, R., Khan, S., Kurt-Peker, Y., & Ray, L. (2016, February). Training teachers to integrate computational thinking into K-12 teaching. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education* (pp. 156–157). https://doi.org/10.1145/2839509.2844675

Hubwieser, P., Armoni, M., Giannakos, M. N., & Mittermeir, R. T. (2014). Perspectives and visions of computer science education in primary and secondary (K-12) schools. *ACM Transactions on Computing Education (TOCE)*, *14*(2), 1–9. https://doi.org/10.1145/2602482

Jackson, C., Tank, K. M., Appelgate, M. H., Jurgenson, K., Delaney, A., & Erden, C. (2020). History of integrated STEM curriculum. In C.C. Johnson, M.J. Mohr-Schroeder, T.J. Moore, & L.D. English (Eds.), *Handbook of research on STEM education* (1st ed., pp. 169-183). Routledge. https://doi.org/10.4324/9780429021381

Kiger, M. E., & Varpio, L. (2020). Thematic analysis of qualitative data: AMEE guide no. 131. *Medical Teacher*, *42*(8), 846–854. https://doi.org/10.1080/0142159X.2020.1755030

Krefting, L. (1991). Rigor in qualitative research: The assessment of trustworthiness. *American Journal of Occupational Therapy*, *45*(3), 214–222. https://doi.org/10.5014/ajot.45.3.214

Lee, I., Grover, S., Martin, F., Pillai, S., & Malyn-Smith, J. (2020). Computational thinking from a disciplinary perspective: Integrating computational thinking in K-12 science, technology, engineering, and mathematics education. *Journal of Science Education and Technology*, *29*(1), 1–8. https://doi.org/10.1007/s10956-019-09803-w

Merriam, S. B., & Tisdell, E. J. (2015). *Qualitative research: A guide to design and implementation*. John Wiley & Sons.

Merriam, S.B. (1988). *Case study research in education.* Jossey-Bass.

National Science Foundation. (2014, December 8). *The future of computer science education*. https://www.nsf.gov/news/news_summ.jsp?cntn_id=133577

Northrup, A. K., & Floyd, R. E., & Dechert, S. R., & Burrows, A. C. (2021, July), *Coding is the new coal: A history of integrating computer science across Wyoming's K-12 curriculum.* Paper presented at 2021 ASEE Virtual Annual Conference Content Access, Virtual Conference. https://doi.org/10.18260/1-2--36801

Pidgeon, N., Henwood, K., Hardy, M., & Bryman, A. (2004). Grounded theory. In M. Hardy & A. Bryman (Eds.), *Handbook of data analysis* (625–648). Sage. https://dx.doi.org/10.4135/9781848608184

Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies*, *22*(2), 469–495. https://doi.org/10.1007/s10639-016-9482-0

Twenty to Nine LLC. (2021). *Delve*. Online qualitative analysis software. http://delvetool.com

Vogel, S., Santo, R., & Ching, D. (2017, March). Visions of computer science education: Unpacking arguments for and projected impacts of CS4All initiatives. In *SIGCSE '17: Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (pp. 609–614). (Annual Conference on Innovation and Technology in Computer Science Education, ItiCSE). ACM. https://doi.org/10.1145/3017680.3017755

Webb, M., Davis, N., Bell, T., Katz, Y. J., Reynolds, N., Chambers, D. P., & Sysło, M. M. (2017). Computer science in school curricula of the 2lst century: Why, what and when? *Education and Information Technologies*, *22*(2), 445–468.

Wing, J. M. (2006, March). Computational thinking. In *Communications of the ACM. 49*(3)*,* 33-35. ACM. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2011, Sept.). Computational thinking. In 2011 *IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* (p. 3) IEEE. https://doi.org/10.1109/VLHCC.2011.6070404.

Wolcott, H.F. (1990). *Writing up qualitative research.* Sage.

Wyoming Department of Education. (2021, April 7). *2020 Wyoming Computer Science Content & Performance Standards (WYCPS).* https://edu.wyoming.gov/wp-content/uploads/2021/04/2020-CS-WYCPS-with-all-PLDs-effective-04.07.21.pdf

Wyoming 2019 Senate Bill 29. Education-computer science and computational thinking. Retrieved from https://legiscan.com/WY/bill/F0029/2018

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.

Appendix A: Interview Questions

Interviews were conducted individually between June 1 and August 2, 2021, using the Zoom online platform. In each interview, the interviewees were asked the following open-ended questions.

1. From your point of view as an educator, how is universal, compulsory computer science education important to:
   A. Your students?
   B. Your school district?
   C. The future of our state and region?

   What specific benefits do you think students will gain? Due to the already crowded curriculum, what tradeoffs raise your concern?

2. What specific skills and knowledge from the computer science training program do you foresee that you will be readily able to use with your students and apply in your classroom? What learning activities will be most influential on your future teaching?

3. What is your action plan in applying specific computer science skills you have learned in the computer science training program to your classroom?

4. In terms of specific instruction and overarching issues, do you see compulsory computer science education as a vehicle for student empowerment? Do you envision this to be different for different groups of students (for example, college-bound, CTE, etc.)? If so, in what ways?

5. What other issues surrounding the delivery of universal computer science education in our state and region would you like to address?

6. What else would you like to add to this discussion?

Appendix B: Summary of Interviews

The following excerpts summarize each interview as it relates to the research question.

Jay: “In computer technology, everything changes so quickly. The trick is not doing this in one great district but doing it everywhere.”

The first author interviewed Jay on June 25, 2021. The first author has known Jay for approximately 20 years. He earned his first degree, an associate of applied science in photographic communications. He completed his bachelor’s degree in education from an experimental distance program delivered in his town of residence. He has been teaching in the same district for over 20 years.

Jay believes that our state needs to keep innovating and looking to the future, and CS education is foundational to innovation. In addition, iPads and other devices are available to most students in his district, and he felt that device availability promotes computer literacy. As telework becomes common, the students’ knowledge of computers will help them live and work in our rural state after they graduate. Instituting universal CS education will level the playing field. All students deserve the same opportunity, and CS is foundational to the equality of opportunity.

Regarding the challenges he faced in the CS training program, he found that, as his first class, coding was difficult, and he was concerned that this sequence might discourage teachers from starting the program. He would have benefited from more technical support in the beginning. Although learning the assembler was important for understanding the history of coding, he believed that teachers learn differently than traditional CS students. Overall, he felt that the program was well constructed and met a need in the school districts but suggested evaluating the sequence of classes.

At the interview's conclusion, Jay mentioned that he would have benefited from more personal help when learning to code. He speculated whether using lab assistants would facilitate differentiated instruction and whether this would be a good model for teaching coding in schools. In closing, Jay stated, "Universal computer science instruction will be necessary for the future when everyone will need computer literacy to live effectively in society."

Tia: "Computer programming is a form of communication. Education changes as the world changes, so we remove that which is obsolete and replace it with what is currently important."

The first author interviewed Tia on June 10, 2021. The first author has known Tia for approximately 10 years. Tia has an associate of arts in elementary education from the same two-year college where she earned her Computer Science endorsement, then earned bachelor's degrees in history and secondary education from a university in another state. She also holds a master's degree in curriculum and instruction. She taught in a different region for two years before moving back to her current location, where she is in her 18th year of teaching U.S. history and English at the High School level.

Tia discussed the tradeoffs needed to fit CS into the curriculum. While keyboarding and other basic skills will be pushed back to elementary, Tia stated that she is not very concerned about the time constraints. Districts will be able to evaluate outdated curriculum elements, such as typing, and although students are currently allowed to choose between electives, such as U.S. firearms education and technology/CS, this will change with time, and CS will move from an elective to a requirement. Tia stated, "Computer science will become a universal requirement at all levels of and probably K-16."

When discussing her experience in the provided CS training program, she first focused on troubleshooting, beginning with the learning management system (LMS), which has immediate classroom applicability. She mentioned building her confidence in making her LMS pages more user friendly. Troubleshooting and confidence go hand in hand. Students will not try to fix a coding problem without confidence. She said her students have different levels of confidence. She discussed her male students in advanced placement CS class or those on the robotics team as having abundant confidence.

Sam: "In the end, language and coding [are] not the biggest deal; it's all about problem-solving."

Sam was interviewed on June 8, 2021. The first author met him through the cohort. Sam began his academic career in journalism, earning a bachelor's degree in radio and television from a southern university. Later, he earned a bachelor's degree in education. He has taught business and career and technical education (CTE) for 21 years. For the last five years, he has also taught CS and coding. In addition to the CS training program, Sam has completed teacher training through code.org.

Sam identified problem-solving as a central principle of CS. Regarding specific gains for students, using computers allows students to gain confidence in computing and problem-solving. With confidence and experience, students can use computers as tools to solve problems in other academic areas and in their lives.

When asked if he was concerned about the tradeoffs required to incorporate CS into the already crowded curriculum, he stated that he does not see this as a problem. For example, there has always been room for computing in his personal finance class. He expressed his appreciation for the published CS standards because they give him goals to pursue with his students. He has found that he can address approximately 25% of the standards in applications courses. His process is to integrate CS and technology into other subjects while addressing CS standards.

As the interview finished, Sam noted that CS is a great vehicle for individualized instruction, allowing him to meet students where they are in terms of skill. In his current CS and coding classes, he employs individualized instruction because his students are at different skill levels. With 16 students, there may be 16 different problems being solved concurrently. The most effective teaching method is differentiated instruction. He also noted that students learn especially well when actively engaged in programming. In the last couple of years, he has found that his students want to get to work instead of listening to him for very long.

Nan: "The technology piece? There's no way around it. We need to bring back innovation and creativity, and if we are not learning it now, we are behind already."

The first author interviewed Nan on June 1, 2021. The first author met her through the cohort. Nan has an associate's degree from a Community College in a neighboring state, a bachelor's degree in microbiology from another neighboring state, and a master's degree in education from an online university. She has been a Middle School science teacher for 12 years.

Nan felt that technology and CS provide students with a well-rounded education. We need to teach what students will need and level the curriculum so that students are not spending more time on math and reading than everything else. We need to blend and balance, assigning equal weight to all STEM fields, including CS. While schools focus on reading, writing, and mathematics, the statewide science test scores indicate that only 46% of students in our state are proficient in science. We must prepare students for "jobs that have not been created," so we need to validate the importance of CS alongside other core subjects to help prepare students for the future.

When asked about what tradeoffs raise her concern, given the already crowded curriculum, Nan stated that this is always the scramble, especially for elementary teachers, because school district administrators tend to add teaching requirements without removing anything. She sees a need to be innovative and creative to round out the curriculum. We must prepare students for their futures, and there is no way to do that without CS. Students should learn CS in early elementary school, and it should be part of the core subjects along with mathematics and English. She acknowledged that this is a hard sell to many stakeholders, but CS is necessary for a modern curriculum. We also need to examine what drives the time allocation and curriculum and ensure the focus stays on the students. For example, can stakeholders evaluate whether they are obtaining "bang for our buck" regarding reading at the grade level? Are there strategies other than pulling the students from social studies, science, and other subjects to help them? What if we scaled back mathematics and English interventions and simultaneously enriched other curricular areas? She concluded by stating that "we need to seek balance and make sure all subjects are given the appropriate weight."

Sol: "Our future is technology; it's not an option, it's reality."

The first author interviewed Sol on August 2, 2021. The first author met him through the cohort. Sol has a bachelor of science degree in secondary education–social studies. He has recently started work toward a master's degree in administration. He began his degree work at a southern state college, where he played baseball, then transferred to the university in his home state to complete his degree. He completed his CS training in May 2021. Sol and his wife taught in an urban school system, then moved home a year ago to teach in a local school district. Last year, he divided his teaching duties between High School student support and sixth-grade social studies. This year, Sol serves as the Middle School CS instructor.

Sol began the interview by saying that CS and technology are ubiquitous and important to the students, school district, and future of our state. All schools are implementing technology as an integrated part of teaching and learning. Paper is becoming "a thing of the past" in many areas. Due to the changing nature of information literacy, students need to learn where to go and not go on the internet when using digital sources. The internet is pervasive throughout his school district and the world at large, so teaching students how to use technology is necessary for all aspects of life. He posited that educators need to accept this as reality.

He was not overly concerned when asked what concerns he had regarding tradeoffs in the already crowded curriculum. He will be teaching technology and CS as an elective, and his class rosters have full enrollment, indicating that students acknowledge CS as the path forward to the modern world. Technology is not an option; it is a reality, and teachers need to integrate it into everything taught.

Other concerns include a potential barrier to using social media in the classroom. He felt that his administration has a limited view of what is appropriate and acknowledged that students are distracted by the internet if he does not check on them continually. There could be an issue with students going to forbidden sites, but overall, he feels responsible cyber-citizenship is beneficial and that the good outweighs the bad. He will push for allowing social media and internet access to play a role in his classroom.

Received: *February 17, 2022* Revised: *March 22, 2022* Accepted: *April 06, 2022*

Cite as: Northrup, A. K., Burrows A. C., & Slater, T. F. (2022). Identifying implementation challenges for a new computer science curriculum in rural western regions of the United States. *Problems of Education in the 21st Century*, *80*(2), 353-370. https://doi.org/10.33225/pec/22.80.353

Astrid K. Northrup, PE *(Corresponding author)*	MS, PhD Candidate, University of Wyoming & Professor, Northwest College, USA. E-mail: astrid.northrup@nwc.edu ORCID: https://orcid.org/0000-0001-7572-8926
Andrea C. Burrows	Ed.D., Professor, College of Education, University of Wyoming, USA. E-mail: andrea.burrows@uwyo.edu ORCID: https://orcid.org/0000-0001-5925-3596
Timothy F. Slater	PhD, Professor, College of Education, University of Wyoming, USA. E-mail: Timothy.Slater@uwyo.edu ORCID: https://orcid.org/0000-0002-2086-213X