5-2024

Evaluating an Integrated Science, Technology, Engineering, and Math/Computational Thinking Professional Development Program for Elementary Level Paraprofessional Educators

Aubrey A. Rogowski
Utah State University, aubrey.rogowski@usu.edu

EVALUATING AN INTEGRATED SCIENCE, TECHNOLOGY, ENGINEERING, AND MATH/COMPUTATIONAL THINKING PROFESSIONAL DEVELOPMENT PROGRAM FOR ELEMENTARY LEVEL PARAPROFESSIONAL EDUCATORS

by

Aubrey A. Rogowski

A dissertation submitted in partial fulfillment
of the requirements for the degree
of

DOCTOR OF PHILOSOPHY

in

Instructional Technology and Learning Sciences

Approved:

______________________	______________________
Mimi Recker, Ph.D. Major Professor	Lisa Lundgren, Ph.D. Committee Member
______________________	______________________
Jody Clarke-Midura, Ed.D. Committee Member	Jessica Shumway, Ph.D. Committee Member
______________________	______________________
Kristin A. Searle, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2024

ABSTRACT

Evaluating an Integrated Science, Technology, Engineering, and Math/Computational Thinking Professional Development Program for Elementary Level Paraprofessional Educators

by

Aubrey A. Rogowski, Doctor of Philosophy

Utah State University, 2024

Major Professor: Mimi Recker, Ph.D.

Department: Instructional Technology and Learning Sciences

This dissertation describes an evaluation of a district-wide Science, Technology, Engineering, Math, and Computational Thinking (STEM/CT) Professional Development Program intended for paraprofessional educators to help learn about computational thinking and how it could be integrated into a Science, Technology, Engineering, and Math (STEM) specialty class for K-6 students. I evaluated eight participants' experiences participating in the STEM/CT Professional Development Program, how they understood computational thinking, and whether the professional development program prepared them to integrate computational thinking into the STEM specialty class. I conducted a qualitative study using data from three sources: pre-survey, reflective interview, and post-survey. The findings from the evaluation showed that the STEM/CT Professional Development Program was a positive experience for participants and provided opportunities for participants to develop their understanding of computational thinking. Additionally, the findings illustrated the challenges participants faced as they began integrating computational thinking into the STEM specialty class, such as uncertainty about computational thinking, a lack of time and resources, and the inability to collaborate with other paraprofessionals. The findings offer insights into how paraprofessionals can be better supported, including increased administrator support, additional preparation time, and more teaching materials and resources. Lastly, findings suggest various ways to improve the STEM/CT Professional Development Program.

(227 pages)

PUBLIC ABSTRACT

Evaluating an Integrated Science, Technology, Engineering, and Math/Computational Thinking Professional Development Program for Elementary Level Paraprofessional Educators

Aubrey A. Rogowski

For my dissertation, I looked at a training program one Utah school district used to teach paraprofessional educators science, technology, engineering, math, and computational thinking. Specifically, the program taught them about what computational thinking is and how they could use it when teaching science, technology, engineering, and math to students from kindergarten to sixth grade. While reviewing this program, I evaluated 1) The experiences the paraprofessionals had with the program, 2) Whether the paraprofessionals understood computational thinking, and 3) Whether the program prepared them to teach computational thinking to K-6 students.

I worked with eight paraprofessionals who participated in this program. Each participant was given a survey before and after the training program, and I interviewed each of them to gather their thoughts, feelings, and experiences at the end of the program.

This evaluation showed that the program provided a positive experience for participants and opportunities for them to understand computational thinking and how they can teach elementary school children those concepts. My evaluation also highlighted several ways the school district can support paraprofessionals to make them more effective when teaching computational thinking.

DEDICATION

This dissertation is dedicated to my husband and two beautiful daughters; I love you.

ACKNOWLEDGMENTS

I would like to express my deepest appreciation to my doctoral chair, Mimi Recker, and my reading committee members, Jody Clarke-Midura and Lisa Lundgren, who generously provided their guidance and feedback throughout the dissertation process. I would also like to thank committee members Kristin Searle and Jessica Shumway for their knowledge, expertise, and encouragement. This endeavor would not have been possible without the support from our department head, Andy Walker, and the Instructional Technology and Learning Sciences Department for funding my work through the Research & Development Scholarship. Additionally, I would like to thank the STEM Action Center of Utah for allowing me to take on this evaluation project. I am extremely grateful to the school district, the STEM/Computer Science Consultant, and the paraprofessionals who participated in my dissertation research- this work would not have been possible without them.

I would like to sincerely thank my peers and mentors who have been through this process with me and cheered me on to the finish line, including Dr. Ryan Cain, Dr. Katarina Pantic-Anderson, and Dr. Julie Lamara. Thank you for the many feedback sessions and moral support. I would like to express my deepest gratitude to my peer and dear friend, Dr. Megan Hamilton. Thank you for the countless hours spent brainstorming, problem-solving, and reading draft after draft throughout my journey. Thank you for encouraging me to keep going. We made it!

Lastly, I would like to recognize the love and support from my biggest cheerleaders, my family; thank you! To my parents, thank you for helping with Eleanor while I worked and traveled to conferences. I would like to acknowledge the patience and love of my sweet Eleanor as she attended online classes, sat through research meetings, and traveled with Mommy to conferences and for baby Magnolia for being the best little sleeper so I could finish my manuscript. Words cannot express my gratitude to my husband, Cameron, for his unwavering love and support throughout the frustrations, long nights and weekends, and years I have taken pursuing this dream. Thank you.

Aubrey A. Rogowski

CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1

INTRODUCTION

Equity in computer science (CS) education involves four components: access, capacity, participation, and experience (Warner et al., 2019). Applying this framework to the Utah context shows that elementary students in many communities lack equitable access to opportunities to learn computing concepts. In 2019, only 22% of Utah's elementary schools offered computer science instruction (P.J. Rich et al., 2019). Thus, most students must wait until high school to have access to computer science classes. Access to computing education helps to develop computational thinking skills required in many industries. This lack of access may ultimately leave students unprepared for future academic development and skills needed to enter the workforce.

Many areas of demand for computing in the workforce include “agriculture, transportation, healthcare, education, and financial services” (Bonilla & Paul, 2019, p. 9). In addition to workforce preparation, developing computational thinking skills has implications for students’ academic development. Computational thinking (CT) is increasingly used by teachers in all disciplines, “enabling their students to use its core concepts and dispositions to solve discipline-specific and interdisciplinary problems” (Yadav et al., 2017, p.56). In addition, integrating computational thinking across subject areas allows students opportunities to problem-solve and make connections across disciplines (Hunsaker, 2020). Engaging in computational thinking can foster students' critical thinking, collaboration, and creativity (Mishra & Yadav, 2013) and help them participate in our digital society, thus preparing them to be future leaders and innovators (K-12 Computer Science Framework, 2016).

Problem Statement

In Utah, there is a lack of access to equitable CS education (P.J. Rich et al., 2019) because there is currently no capacity within the state to provide it. There are not enough teachers with the skills needed to teach computing education. This issue begins with a lack of preparation in preservice teacher education programs and continues in school districts throughout the state. There is simply not enough support for elementary teachers regarding professional development (PD) around CS, and in some school districts, the responsibility is given to paraprofessionals. As P.J. Rich et al. (2019) noted, “the most common way for elementary school teachers to participate in [CS] professional development opportunities is by searching them out on their own and negotiating with their administrators” (p. 11). If elementary students are going to receive equitable computing education, there needs to be more support and training for elementary school teachers and paraprofessionals tasked with teaching computational thinking and computer science.

To help address this issue, the STEM Action Center of Utah created the Computing Partnership Grant to help broaden participation and experience in CS. This grant provides funding to K-12 schools throughout Utah to provide opportunities for Utah students to be exposed to STEM and computing activities and programs in various formal and informal learning environments. Part of the grant includes funding to develop professional development to enhance teachers’ and paraprofessionals' knowledge of computational thinking to teach and facilitate these types of programs. The Utah State Board of Education defines computational thinking as:

> Computational thinking (CT) is a problem-solving process that includes several characteristics, such as logically ordering and analyzing data and creating solutions using a series of ordered steps (or algorithms), and dispositions, such as the ability to confidently deal with complexity and open-ended problems. (USBE, 2019)

Many school districts throughout Utah (including Marshall School District (MSD, pseudonym for the site of this study) have received a Computing Partnership Grant to implement STEM and computing programs within their schools.

Purpose and Objectives

With the adoption of the Utah K-5 Computer Science Standards in 2019 (USBE, 2019), school districts and universities across the state are creating plans to prepare teachers to implement the standards. For example, the Alpine School District provides 80 educators with CS education professional development to guarantee at least one qualified educator in each school. Five school districts in the state have partnerships with a commercial provider, BootUp Professional Development, to implement CS initiatives. Brigham Young University's McKay School of Education is creating a new course for all preservice teachers to learn the basics of elementary coding. The Davis School District is piloting computer programs to teach students Python in grades 5-8 (DSD, n.d.) While these efforts are commendable, thousands of current teachers are unprepared to implement computational thinking in their classrooms.

In many elementary schools throughout Utah, the responsibility to teach and engage students in computational thinking has been delegated to paraprofessionals in various school districts (e.g., Alpine, Cache, and Juab). Often, the paraprofessionals' background or understanding of computational thinking is unknown. In the Marshall

School District specifically, a preliminary survey of these paraprofessionals called "instructional assistants" indicated they are mostly middle-aged women with little technology or computer science experience. In addition, many (not all) of these paraprofessionals lack the knowledge of teaching and cultivating learning environments (K. Taylor, personal communication, 2020). With little to no background in computing education, paraprofessionals are still required to provide students with experience in CS education.

In the Marshall School District, paraprofessionals teach a STEM specialty. They will now be tasked with integrating the Utah K-5 Computer Science standards (specifically the computational thinking strand) into the STEM specialty. In this dissertation project, I evaluate (Weiss, 2004) the Marshall School District STEM/CT Professional Development Program for paraprofessionals regarding how it prepares paraprofessionals to integrate and teach computational thinking into this STEM specialty. I developed the following problem statement and goal after discussing the goals of the Computing Partnership Grant with the Utah STEM Action Center and the STEM/CT Program with the Marshall School District.

MSD Problem Statement: In Utah, elementary students in many communities lack equitable access to opportunities to be exposed to computing concepts. In the Marshall School District, there is not enough time for teachers to receive training on CS nor time in the school day to teach it. In an effort to expose all students to STEM and CS experiences, the Marshall School District has delegated the responsibility to teach the new K-5 CS standards to paraprofessionals as part of a STEM specialty.

MSD Goal: Marshall School District paraprofessionals will be able to effectively define computational thinking, recognize computational thinking in practice, and design computational thinking and STEM-integrated lessons to engage K-6 students in building STEM and computational thinking skills.

Evaluation Questions

The following questions guide this evaluation of the MSD STEM/CT Professional Development Program:

1) What were the participants' experiences in the STEM/CT Professional Development Program?
2) How does participating in the STEM/CT Professional Development Program affect participants' understanding of computational thinking?
3) What additional activities, resources, strategies, supports, and training do participants still need to design and teach STEM/CT integrated lessons?

Significance of Evaluation

This evaluation is significant because it can direct the current and future efforts as school districts strive to implement the new K-5 CS standards with elementary schools across Utah. Additionally, it provides specific strengths, areas for improvement, and additional strategies needed to support paraprofessionals in their understanding of computational thinking and their ability to integrate computational into STEM learning experiences.

Definition of Terms

*Computational Thinking: "*Computational thinking (CT) is a problem-solving process that includes several characteristics, such as logically ordering and analyzing data and creating solutions using a series of ordered steps (or algorithms), and dispositions, such as the ability to confidently deal with complexity and open-ended problems" (USBE, 2019).

Paraprofessional: a non-licensed educator who provides instruction and support

Professional Development: continuing education or learning to support educators in maintaining and developing new skills and knowledge related to a content area, teaching, and/or learning.

Summary

Considering that CS is one of the fastest-growing fields in education, the lack of equitable access to CS education for elementary students in Utah is problematic. However, equitable access is a twofold problem. Not only is there a lack of opportunity for students to be exposed to computing, but teachers also do not have the knowledge or training to teach CS within the elementary classroom. This dissertation evaluates the current efforts of the Marshall School District to provide more equitable access to computing education through paraprofessionals teaching computational thinking to all K-6 students in a STEM specialty class. Additionally, this work helps acquire a better understanding of the experiences of paraprofessionals engaging in this work and the additional types of activities, resources, strategies, supports, and training they still need.

Organization of Dissertation

This evaluation follows a traditional five-chapter dissertation format. This format includes an introduction, a review of significant and timely literature, methods, evaluation findings, a discussion of the findings, and a conclusion. Chapter One presents the background for my evaluation and the problem statement the evaluation helps to address. I also present the evaluation questions and define key terms for the reader. A literature review on paraprofessionals and their roles in Utah, computational thinking, research on computational thinking professional development for K-6 teachers, and K-6 educators' attitudes, beliefs, knowledge, and values about computational thinking are presented in Chapter Two. Chapter Two also reviews the evaluation framework, Guskey's Five Levels of Professional Development Evaluation, used in this dissertation. In Chapter Three, I describe the evaluation design used for this evaluation. This description includes recruitment, data collection, and data analysis procedures. Chapter Four reports demographic information for the participants, descriptive statistics results, and results from the evaluation by Guskey's evaluation level through qualitative analysis. Additionally, Chapter Four presents the findings from the qualitative analysis of interview data. Finally, I discuss my findings from the evaluation in Chapter Four in Chapter Five. Chapter Five also includes recommendations for practice as a result of the evaluation and suggestions for future evaluation and research.

CHAPTER II

LITERATURE REVIEW

Introduction

This chapter provides an overview of the literature about the STEM/CT Professional Development Program evaluation. The chapter is divided into several sections. First, I describe the literature review process. Then, I define paraprofessionals and their roles in Utah Public Schools. Next, I define computational thinking (CT). Following that, I review the research on computational thinking professional development for K-6 teachers. Then, I review the research on K-6 educators' attitudes, beliefs, knowledge, and values about CT. Lastly, I present the evaluation framework used to evaluate the STEM/CT Professional Development Program.

Literature Review

I conducted a thorough literature review to find relevant literature related to the topics about the elements of the STEM/CT Professional Development Program, including defining computational thinking, research on K-6 professional development on CT, K-6 educators' attitudes, beliefs, knowledge, and values about CT, and evaluation frameworks.

First, guiding questions aligned with the evaluation questions were developed for the literature review, including:

1) What is computational thinking?

2) What is known about computational thinking professional development for K-6 educators?
3) What is known about K-6 educators' attitudes, beliefs, knowledge, and values about computational thinking?

Next, inclusion and exclusion criteria were developed for each guiding question. Then, a Boolean search was used to conduct an in-depth search of all major academic databases within the field of education. Databases included in this search include Academic Search Ultimate, APA PsycInfo, Computer Source, Education Source, ERIC, Google Scholar, JSTOR, Professional Development Collection, and ProQuest Dissertations & Theses Global (see Appendix A for a list of journals included in the databases searched). The thesaurus was used in each database to determine search terms for each specific database search (e.g., use "professional education" for "professional development"). After a search using specific search terms, the title and abstract of each article were read to determine if the article was relevant to the guiding question and met the inclusion and exclusion criteria developed for each question. Next, articles were read for a full review to determine if they would be included or excluded in the study based on the inclusion and exclusion criteria. Finally, a table of findings was created for each guiding question, including the database, search terms, number of articles found, and number of relevant articles with a citation (see Appendix B).

Defining Paraprofessional Educators

Since the 1940s, paraprofessionals have been hired into educational settings for a variety of reasons, including addressing teacher shortages, addressing issues of poverty, helping bridge the gap between race, language, and culture, providing jobs for economically needy adults, and providing students with individualized services (Ashbaker & Morgan, 2001; Lewis, 2004). Today, paraprofessionals are often used due to budgetary constraints within schools.

The term "paraprofessional" is not always used as the job title in school districts across the United States, even though the job duties are the same. Instead, titles such as classroom assistant, paraeducator, classroom aide, teaching assistant, and instructional assistant are often used to describe the role and duties of a paraprofessional. The Bureau of Labor Statistics classifies a paraprofessional as a teacher assistant. This dissertation uses the term "paraprofessional."

As of 2021, more than 1.2 million paraprofessionals (teaching assistants) are working with P-12 students nationwide (U.S. Bureau of Labor Statistics, 2022). These paraprofessionals are a staple in many schools across the nation, providing students with additional attention and support in the classroom. Within Utah schools, there are both paraprofessionals and paraeducators. Paraprofessionals serve a supporting role within schools and provide non-instructional support. A paraeducator is a paraprofessional "who provides instructional support under the direct supervision of a licensed educator" (USBE Paraeducator Manual, 2023, p.7).

However, there is often ambiguity around the role of a paraprofessional. They often provide additional support to students with disabilities, working with small groups who need more support within the classroom, assisting with reading instruction in the younger grades, and even teaching their specialty classes. When No Child Left Behind (NCLB) was passed in 2000, a requirement for teaching assistants to work under the direction of a licensed teacher was mandated (Hauerwas & Goessling, 2008). This mandate means that paraprofessionals should not legally be designing instruction and that a certified teacher should prepare lesson plans (Etscheidt, 2005).

From this review, the role of "paraprofessional (instructional assistant)" for participants in my study is a paraeducator in Utah because they take on an instructional role. However, the terms "paraprofessional," "teacher assistant," "paraeducator," and "instructional assistant" appear to be used interchangeably by several school districts, and the distinction of positions is often unclear. NCLB considers teachers' aides as paraprofessionals. This lack of clarity is essential to note because they should receive additional support and training to be effective in their professional role. Additionally, No Child Left Behind (NCLB) has several implications regarding who should be designing instruction around computational thinking in the district. According to NCLB policy, paraprofessionals should not be planning their own lessons, giving direct instruction, or introducing new academic skills or concepts to students because this falls outside the scope of instructional support.

Preparation for Paraprofessionals in Utah

The Utah Code (53F-2-411) defines a paraeducator as "a school employee who: i) delivers instruction under the direct supervision of a teacher." To become a highly qualified paraprofessional or paraeducator in Utah, a paraprofessional needs to either have an associate degree (or higher) or 48 credit hours from an accredited higher education institution or pass the ParaPro Exam with a score higher than 460 to demonstrate proficiency and ability to help instruct in reading, writing, and math. In addition, several universities and professional organizations throughout the state of Utah offer courses or a paraprofessional certificate program, including Utah State University, Weber State University, Utah Library Association, and Utah Valley University Continuing Education.

A paraeducator manual is available to paraprofessionals that explain their roles and responsibilities, the Utah Standards for Instructional Paraeducators, and each standard's core and supporting competencies (Dickson & Voorhies, 2023). The manual also includes basic instructional strategies and behavior management techniques. The Utah State Board of Education also offers a Utah Paraeducator Training course for special education paraprofessionals.

Defining Computational Thinking

Computer scientists, researchers, and educators have not yet reached a consensus on the definition of computational thinking (CT) (e.g., Barr & Stephenson, 2011; Grover & Pea, 2013). Table 1 summarizes the various definitions of CT in the literature. For many, computational thinking is an approach to problem-solving. Wing stated, "Computational thinking is a fundamental skill for everyone, not just for computer scientists. To reading, writing, and arithmetic, we should add computational thinking to every child's analytical ability" (Wing, 2006, p. 33). She described computational thinking as a skill used to solve problems, design systems, and understand human behavior (Wing, 2006).

Several organizations and scholars (Barefoot Computing, n.d.; Brennan & Resnick, 2012; Chen et al., 2017; Google for Education, n.d.; Grover & Pea, 2013; ISTE, n.d.; Yadav et al., 2019) have identified various computational thinking skills/practices, and computational thinking approaches. While no consensus exists, much overlap exists on the most essential components of computational thinking.

Grover and Pea (2013) reviewed the state of K-12 computational thinking and the various elements of CT described by researchers in the literature. They defined nine elements, including debugging, efficiency and performance constraints, conditional logic, iterative thinking, systematic processing of information, symbol systems and representation, algorithmic notions of flow control, and decomposition with abstraction as the "keystones" of CT that the research community has widely acknowledged as fundamental elements of CT. While Grover and Pea (2013) acknowledged the lack of consensus amongst researchers, they stated that CT can be described generally as an approach to problem-solving that should be taught alongside other essential, basic literacies such as mathematics and science.

K12CS.org is home to the K-12 Computer Science Framework, a conglomerate of organizations, teachers, researchers, administrators, and industry leaders developed. Within the framework, computational thinking is defined as "the thought processes involved in expressing solutions as computational steps or algorithms that can be carried out by a computer" (K-12 Computer Science Framework, 2016, p. 68). The framework defines specific skills, including "designing algorithms, decomposing problems, and modeling phenomena" (K-12 Computer Science Framework, 2016, p.70).

In a joint effort, the International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) produced an operational definition of computational thinking for K-12 education. The definition is as follows:

> Computational thinking (CT) is a problem-solving process that includes (but is not limited to) the following characteristics:
>
> - Formulating problems in a way that enables us to use a computer and other tools to help solve them.
> - Logically organizing and analyzing data
> - Representing data through abstractions such as models and simulations

- Automating solutions through algorithmic thinking (a series of ordered steps)
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources
- Generalizing and transferring this problem-solving process to a wide variety of problems." (ISTE & CSTA 2011, Section 2)

The Utah State Board of Education (USBE) Computer Science Standards (2019) defined computational thinking as follows:

> Computational thinking (CT) is a problem-solving process that includes several characteristics, such as logically ordering and analyzing data and creating solutions using a series of ordered steps (or algorithms), and dispositions, such as the ability to confidently deal with complexity and open-ended problems. CT is essential to the development of computer applications, but it can also be used to support problem-solving across all disciplines, including math, science, and the humanities. Students who learn CT across the curriculum can begin to see a relationship between subjects as well as between school and life outside of the classroom. (p. 5)

Like many above, this definition emphasizes CT as an interdisciplinary problem-solving approach.

Table 1

Various Definitions/Framing of Computational Thinking in the Literature

Citation	Definition/Framing of Computational Thinking
Yadav et al., 2022	Cognitive- *metacognition*
Tsarava et al., 2022	Cognitive- *metacognition*
Cansu & Cansu, 2019	Cognitive- *activities based on CT are meant to improve cognitive skills*
Li et al., 2020	Thought Process- *situated thinking*
Guzdial et al., 2019	Thought Process- *"good thinking"*
Grover & Pea, 2018	Thought Process
Wing, 2006	Thought Process Problem-Solving Approach
Cuny, Snyder, & Wing, 2010	Thought Process
Aho, 2012	Thought Process
CSTA & ISTE, 2011	Problem-Solving Approach
Barr & Stephenson, 2011	Problem-Solving Approach
Kale et al., 2018	Problem-Solving Approach

Table 1 (continued)

Citation	Definition/Framing of Computational Thinking
Lye & Koh, 2014	Problem-Solving Approach
Kafai, 2016	Problem-Solving Approach
Wing, 2008	Problem-Solving Approach
Manilla et al., 2014	Problem-Solving Approach
Selby, 2014	Problem-Solving Approach
Hunsaker, 2020	Problem-Solving Approach
Grover & Pea, 2013	Problem-Solving Approach
Taslibeyaz et al., 2020	Problem-Solving Approach
Barr & Stephenson., 2011	Problem-Solving Approach
Perkovic et al., 2010	CT Components- *Automation, communication, computation, coordination, design, evaluation, recollection*
Csizmadia et al., 2015	CT Components- *Algorithm design, decomposition, generalization, pattern recognition, abstraction, evaluation*
Shute et al., 2017	CT Components- *Algorithms, debugging, decomposition, abstraction, iteration, generalization*
Lee et al., 2011	CT Components- *Abstraction, automation, analysis*
Bell & Lodi, 2019	CT Components- *Abstraction, decomposition, algorithmic design, evaluation, generalization, logical thinking*
Brennan & Resnick, 2012	CT Components- *sequences, loops, parallelism, events, conditionals, operators, data, iterative design, testing & debugging, reusing & remixing, abstracting & modularising*
Kotsopoulos et al., 2017	CT Framework Pedagogical Framework- *unplugged, tinkering, making, remixing*
Brennan & Resnick, 2012	CT Framework- *Computational concepts, computational practices, computational perspectives*
Weintrop et al., 2016	CT Framework- *data practices, modeling & simulation practices, computational problem-solving practices, systems thinking practices*
Angeli et al., 2016	CT Framework- *Curriculum framework based on CT skills for K-6 students*

The CT4EDU project identifies four CT components (they call them "practices"), including abstraction, decomposition, debugging, and patterns (Yadav et al., 2019). Google's Exploring Computational Thinking course identifies eleven CT components: abstraction, algorithm design, automation, data collection, data analysis, data representation, decomposition, parallelization, pattern generalization, pattern recognition, and simulation. Additionally, several other researchers have identified similar components in their CT research. Although these conceptual ideas and working definitions of computational thinking differ, they can help educators guide their teaching practices and integrate computational thinking into the curriculum.

Table 2 describes the four CT practices the CT4EDU team identified (Yadav et al., 2019) with the addition of an algorithm and additional definitions from the research. The CT4EDU team's definitions come from their CT classroom poster set. The school district participating in this evaluation used these definitions to teach students and paraprofessionals about computational thinking.

Table 2

Components of Computational Thinking

Component of Computational Thinking	Citation	Definition	CT4EDU (Yadav et al., 2019) poster definitions
Abstraction	Yadav et al., 2016	"Reviewing how the solution transfers to similar problems" (p. 565)	Focusing on the information I need while ignoring unnecessary details
	Barr & Stephenson, 2011	"*Abstraction* involves generalizing and transferring the problem solving process to similar problems" (Yadav et al., 2016, p. 566, drawn from Barr & Stephenson, 2011).	
	K.M. Rich et al., 2020	"Reducing complexity by focusing on important elements of a problem or situation" (p. 3163.	
	Grover & Pea, 2018	"Simply put, abstraction is 'information hiding.' The act of 'black-box'-ing details allows one to focus only on the input and output. In this sense, then, abstraction provides a way of simplifying and managing complexity" (pp. 25-26).	

Table 2 (continued)

Component of Computational Thinking	**Citation**	**Definition**	**CT4EDU (Yadav et al., 2019) poster definitions**
Decomposition	Yadav et al., 2016	"Breaking down complex problems into more familiar/manageable sub-problems" (p. 565)	Breaking something down into smaller, more manageable parts
	K.M. Rich et al., 2020	"Breaking apart a complex problem or situation to make it more manageable" (p. 3163).	
Debugging	K.M. Rich et al., 2020	"Systematically finding and correcting problems and errors" (p. 3163)	Finding and fixing errors or mistakes
Patterns	K.M. Rich et al., 2020	"Looking for similarities between new problems and problems that have already been solved" (p. 3163)	Looking for similarities or patterns between things
	Grover & Pea, 2018	"Recognizing a repeating pattern also informs how to incorporate iteration or recursion in an algorithmic solution or a functional breakdown of a problem (that also serves the cause of creating manageable and modular solutions)" (p. 25)	
Algorithm	Yadav et al., 2016	"Using a sequence of steps (algorithms) to solve problems" (p. 565) "An *algorithm* (much like a recipe) is a step-by-step series of instructions" (p.566)	Creating step-by-step instructions to complete a task
	Grover & Pea, 2018	"Algorithms are precise step-by-step plans or procedures to meet an end goal or to solve a problem; algorithmic thinking is the skill involved in developing an algorithm" (p. 24).	

Research on Computational Thinking Professional Development for K-6 Teachers

There is a lack of research on professional development for paraprofessionals outside special education. Thus, I focused on research about professional development (PD) for K-6 teachers that focused on computational thinking (CT) (see Table 3). Many of the existing empirical studies have smaller sample sizes, meaning the findings may not be generalizable or representative of elementary teachers across America; that being said, there are still many key concepts and principles that can be drawn from their findings.

Table 3

Empirical Research on Computational Thinking Professional Development for K-6 Teachers

Author(s), Article/Chapter Year	**Participants**	**Professional Development Model**	**Computational Thinking + Discipline**
Hestness, Ketelhut, McGinnis, & Plane. (2018). Professional knowledge building within an elementary teacher professional development experience on computational thinking in science education.	13 Elementary mentor teachers (3rd-5th grades)	Communities of Practice w/pre-service teachers /Workshop series	CT integration with science
Israel, Pearson, Tapia, Wherfel, & Reese. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis	7 elementary teachers and 2 administrators	Embedded Coaching	School-wide CT implementation/integration (Library/media, technology, art, enrichment, classroom)

Table 3 (continued)

Author(s), Article/Chapter Year	**Participants**	**Professional Development Model**	**Computational Thinking + Discipline**
Li, Richman, Haines, & McNary. (2019). Computational thinking in classrooms: A study of a PD for STEM teachers in high-needs schools	25 teachers- 10 elementary, 6 middle schools, 8 high schools, 1 K-12 instructional coach	Blended PD (5-day summer institute, 3-hour F2F meeting, online modules, online interaction w/CT experts, math and science educators, and technical support)	CT integration
Ouyang, Hayden, & Remold. (2018). Preparing upper elementary school teachers for integrating computational thinking into regular classroom activities.	21 elementary teachers (4^{th}-6^{th})	Job-embedded PD, coaching & small groups	CT + STEM integration
P.J. Rich, Mason, & O'Leary. (2021). Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking	291 elementary teachers, 8 district coaches, and 8 school districts	Continuous PD using BootUp Professional Development 501(c)(3) non-profit	CT + coding w/ Scratch
Sands, Yadav, & Good. (2018). Computational thinking in K-12: In-service teacher perceptions of computational thinking.	74 elementary and secondary teachers (45 elementary)	Communities of Practice	CT integration

Many studies taught computational thinking by integrating another subject (usually science). Second, three studies used communities of practice as the professional development model. Third, developing computational thinking skills and practices is difficult for many teachers. Lastly, extended professional development appears to be more effective for providing teachers ample time to develop their CT knowledge and allow them to demonstrate growth in their ability to learn how to teach and integrate computational thinking. These key points are discussed below.

Teaching Computational Thinking Through Integration

Teaching computational thinking through integration is a promising yet daunting teaching model. Most research on CT has been explicitly done with CS education and programming (Li et al., 2020). A literature review from 2000 to 2018 only found six articles with explicit STEM education and CT integration as the research focus (Li et al., 2020). Several studies reviewed in this search (Ketelhut et al., 2020; Israel et al., 2015; P.J. Rich et al., 2021; Sands et al., 2018) approached implementing CT in the K-6 classroom through content integration.

One struggle with this approach was having enough time to plan CT-integrated lessons. Ketelhut et al. (2020) found that although teachers were excited to integrate CT, more time was needed for teachers to plan CT-integrated lessons so they could be taught in the classroom. Additionally, teachers in a few studies (Ketelhut et al., 2020; Li et al., 2019; Sands et al., 2018) struggled to know where or how they could integrate CT within their curriculum, especially for younger grades (K-2). Li et al. (2019) found that it was much easier for secondary teachers to find ways to integrate CT into their content areas. The difficulty of integration may be part of why younger grade teachers may have chosen not to participate in Ketelhut et al.'s (2020) study. However, Israel et al. (2015) found success in integrating CT into other content areas when implementing embedded coaching. Co-planning and co-teaching CT-integrated lessons were crucial to teacher success in the study. Last, Sands et al. (2018) found that most teachers in their study could not successfully integrate computational thinking skills in the classroom due to false ideas about what computational thinking is and a lack of knowledge and awareness of how to implement CT in their classrooms.

As researchers implement professional development for teachers focused on CT integration, they find that teachers are not yet prepared to teach computational thinking in an integrated curriculum due to the lack of CT knowledge and pedagogy. However, the only way CT will be taught equitably for all students is to provide teachers with PD that connects CT to teachers' current curriculum (Yadav et al., 2016). Teachers need continued support as they develop their CT knowledge and pedagogy and find ways to integrate CT into their existing curricula. Before we can push for the broad adoption and implementation of CT in elementary classrooms, more work and research must be conducted to understand better how teachers develop correct knowledge and understanding of CT and how it can be implemented effectively in the elementary classroom. Considering that some teachers struggle to integrate CT (Sands et al., 2018), imagine the uphill battle paraprofessionals will face.

Communities of Practice as a Professional Development Model

Many studies on CS PD use a variation of a community of practice as their chosen professional development model. From my analysis of the literature, a possible reason that communities of practice are used frequently in the teaching and learning of computational thinking is due to the complex nature of the topic. Computational thinking is a difficult skill to develop and integrate. Teachers need support and opportunities to share ideas and get feedback. Yadav et al. (2016) recommend that CT professional development be taught online as a continuous learning opportunity rather than a day or workshop and paired with communities of practice to support and reinforce teacher learning. In their study, Israel et al. (2015) found that being eager and invested in teaching computational thinking was insufficient. The teachers needed support and coaching to plan and teach CT-integrated lessons.

To help provide the support and coaching necessary to integrate CT, Killen et al. (2020) designed a longitudinal PD model focused on communities of practice called the Science Teacher Computational Thinking Inquiry Group ($STIG^{CT}$). This community of practice included pre-service, in-service, and facilitator researchers. The group held four monthly sessions for three hours each. This group structure provided several affordances. First, the teachers, pre-service teachers, and facilitators could share their expertise and ask for help. Second, it allowed them to identify different expertise and knowledge within the group, identify where they had knowledge gaps, and fill in those gaps from others' expertise within the group. Third, the group provided mentorship opportunities. Lastly, teachers had an easier time implementing CT into their inquiry-based lessons. Killen et al. (2020) found this PD model very successful. More than 80% of the participants in their study successfully demonstrated their ability to integrate CT into their final lessons.

Ouyang et al. (2018) took a different approach to implement a community of practice as their PD model by providing more than100 hours of PD that included a weeklong academy, workshops, lesson study cycles with a coach, and continued mentoring, coaching, and support throughout the school year. Teachers were given opportunities to team plan, team-teach, and receive coaching during their teaching. They could create a lesson, test it in the classroom, and then return to their group and refine it as needed. This environment, rich in support and mentorship, is very successful. Ouyang et al. (2018) found that this model significantly increased the teachers' confidence to engage and teach

CT through STEM-based classroom activities. On average, teachers in their study taught three CT/STEM-based lessons every month.

These studies demonstrate the power communities of practice can have on teachers' professional development. In addition, the research on effective professional development has determined that instructional coaching, similar to what was implemented in a few of these studies as part of their community of practice, is a practical, highly influential professional learning activity (Desimone & Pak, 2017). Li et al. (2019) found several benefits of community and collaboration in their study. First, because CT is challenging for many teachers, having peer collaborators within the same school to encourage and learn from each other is beneficial. Additionally, having a peer teacher or team to collaborate with encouraged teachers to stick to it and not give up on challenging tasks.

Moreover, if basic programming is going to be taught, significant technical support must be provided to beginners new to programming. As mentioned above, P.J. Rich et al. (2021) found that although teachers may have years of experience, most are still at a novice level for coding and computational thinking. These "novice" teachers sometimes fear coding and programming, so technical support can help ease their discomfort and fear. Lastly, another benefit Li et al. (2019) found is that when teachers worked collaboratively and shared their different perspectives on CT, the lessons they designed were apt to be more comprehensive due to the collaborative nature and different perspectives put into the design. After examining the results of their study, Sands et al. (2018) also recommend using an online community of practice as a place for teachers to collaborate, plan, and share ideas on integrating CT into their specific content areas to meet their learning goals. These studies provide various examples and models of how communities of practice can be implemented into PD sessions to help teachers learn how to teach and integrate computational thinking.

The Difficulty of Computational Thinking

As part of their study, Ouyang et al. (2108) had the teachers participating in their study self-report their confidence levels by teaching via nine different aspects of CT (e.g., data collection, data analysis, data representation, problem decomposition, abstraction, algorithms and procedures automation, simulation, and parallelization). They found that even after almost a full year of professional development, teachers still did not feel confident or prepared to teach the abstraction or automation aspects of computational thinking. P.J. Rich et al. (2021) found similar results concerning teachers' self-efficacy in teaching CT. Teachers' CT self-efficacy remained stagnant at several research sites and did not improve. The researchers found that "teachers showed the slowest growth in their ability to foster computational practices and perspectives" (P.J. Rich et al., 2021, p. 13). Even though teachers had received a year of continuous PD, there was little growth. Before we can ask teachers to integrate CT into the elementary classroom, much more work must be done (Sands et al., 2018). Part of this work includes determining why teachers' confidence, growth, and self-efficacy in computational thinking are challenging to develop.

One reason growth may be so difficult to achieve in knowledge and ability of CT is that, in addition to the differing definitions of CT, many teachers hold incorrect assumptions and misconceptions of what CT is. In their research, Sands et al. (2018) developed a survey to administer to participants that defined what CT does and does not involve (e.g., *CT involves logical thinking. CT does not involve doing mathematics*). After administering the survey, researchers found that the "majority of the teachers strongly agreed with all the components of computational thinking outlined in the survey items and in many cases that teachers incorrectly agreed with concepts that we did not view as computational thinking" (p.161). These results show that CT is a complex topic for teachers new to the concepts and that more explicit instruction about what computational thinking entails is needed. Furthermore, the survey response Sands et al. (2018) received in their study demonstrates that:

> Many educators have very little knowledge about what these skills are and lack awareness of how these skills can be implemented in their classrooms. The results suggest that there is much work to be done before in-service teachers are able to implement computational thinking in their classrooms. (p.161)

More research must be conducted on what types of instruction, professional development, and activities current elementary teachers need to understand CT correctly and how it can be integrated into the classroom. Researchers working with mentor teachers like Ketelhut et al. (2020) came to similar conclusions. Teachers expressed difficulty integrating it into the science curriculum because the Next Generation Science Standards (NGSS) did not provide guidance on where CT could be integrated within the curriculum or how teachers should integrate CT into the NGSS standards. Again, on similar lines, Israel et al. (2015) concluded, based on their findings, that future research needs to examine how to sequence teacher professional development for novice teachers in CT/CS education. The CT skills and concepts may be more accessible for teachers to understand, adopt, and implement with a correct teaching sequence.

Additionally, Li et al. (2019) recommended that to account for the difficulty of learning CT. The PD needs to be structured to include ample hands-on experiences and activities with CT concepts for teachers to engage with before they can be proficient in the concepts. (Li et al., 2021). P.J. Rich et al. (2021) found that although teachers may have years of experience, most are still at a novice level for coding and computational thinking. At times, these "novice" teachers can fear coding/programming, so having ample support through a community of practice can help ease their fears and provide the technical support many beginners may need if using coding or programming to learn CT. Results from Sands et al. (2018) suggest that technology should not even be introduced to teachers when first learning CT and instead should use an "unplugged" curriculum to introduce CT. Additionally, their results indicate that PD must be explicit about the differences between tools used to help us engage in CT and the CT concepts and practices characteristic of and integral to CT to prevent further misconceptions and incorrect assumptions of computational thinking.

Continuous/Extended Professional Development for Computational Thinking

Continuous or extended professional development is professional development that is of sustained duration. The field has not yet determined a "clear threshold for the duration of effective PD models" (Darling-Hammond, Hyler, & Gardner., 2017, p.15), but research shows that learning that leads to changes in teacher practice seldom occurs through short, one-time professional development sessions (Darling-Hammond et al., 2017). Desimone (2009) described that:

> "pedagogical change requires professional development activities to be of sufficient duration…research has not indicated an exact "tipping point" for duration but shows support for activities that are spread over a semester (or

> intense summer institutes with follow-up during the semester) and include 20 hours or more of contact time." (p.184)

Guskey and Yoon (2009) examined Kennedy's (1998) analysis of time spent in PD and determined that PD that "showed positive effects included 30 or more contact hours" (p. 497). However, Guskey and Yoon (2009) also warn that increased time will be of no benefit if the time is not used effectively. Although there are various opinions on the amount of time required for continuous/extended PD, we know that sustained duration, and not just a one-off workshop, is required.

Regarding the professional development for computational thinking, researchers are finding that long-term, continuous professional development is needed. For example, Ketelhut et al. (2020) found that short-term professional development initially motivated teachers to integrate computational thinking into their curriculum but that continuous professional development is needed to help teachers change and transform their teaching practices around computational thinking. Interestingly, the same teachers from P.J. Rich et al.'s (2021) study who had little to no improvement in CT self-efficacy gains saw significant growth in their computer science/coding self-efficacy throughout the continuous PD. So, based on P.J. Rich et al.'s (2021) findings, when it comes to computational thinking and coding, even teachers with years of experience need more than one year of professional development and experience to develop the self-efficacy needed to teach computational thinking.

Would more continuous PD (perhaps two or three years) significantly increase teachers' self-efficacy regarding teaching computational thinking? Li et al. (2019) concluded from their study that extended PD for teachers is necessary when learning computational thinking. The teachers need sufficient time to grasp and explore the CT concepts they are learning rather than participate in an intensive PD session. Sands et al. (2018) also came to the same conclusion that for teachers to be taught to integrate CT into their classrooms successfully, “we need to develop ongoing and continuous professional development programs that help teachers develop a thorough understanding of what it means to think computationally and then engage their students in computing ideas (Sands et al., 2018; p. 162). Additionally, researchers working with mentor teachers (Ketelhut et al., 2020) found that continuous PD allows teachers to enact change and transform their teaching practices, while more short-term professional development motivates teachers to integrate CT into their teaching practices.

Ouyang et al. (2018) found great success with their continuous PD model. Teachers in their study participated in more than 100 hours of PD and experienced increased confidence and ability to engage in CT instruction in their classrooms. Like Ouyang et al. (2018), Killen et al. (2020) also found similar success in their study by implementing their $STIG^{CT}$ model. These findings align with current research on effective teacher professional development regarding continuous or extended PD cited above. Desimone (2009) identified core features of successful teacher PD. One core feature is duration. Desimone (2009) defined duration as a PD providing more than 20 hours of contact time and spreading PD activities throughout an entire semester (or more). The studies above meet the duration criteria in their PD models and implementations. These studies show that pairing a continuous PD model with communities of practice can potentially impact teachers’ learning of computational thinking significantly.

While this area requires much research, these eight studies provide valuable insight into the successes and challenges of professional development for in-service elementary teachers as they learn about CT and how to integrate CT into their content areas and classrooms. These efforts to train teachers on how to teach and integrate CT into their curricula are needed; however, many professional development models presented in these studies seem unattainable given the current financial situation of most school districts nationwide. For example, a few studies required nearly 100 hours of professional development.

While these extended PD models are effective, are they sustainable? Is district-wide implementation feasible for all teachers in the district to participate in that many hours of professional development? Currently, in Utah, teachers need 100 hours of PD over five years to renew their teaching license (Utah State Board of Education [USBE], n.d.).

K-6 Educators' Attitudes, Beliefs, Knowledge, and Values about Computational Thinking

Table 4

Empirical Research on K-6 Teachers' Attitudes, Beliefs, Knowledge, and Values about Computational Thinking

Author(s), Year, Article	Participants	Attitudes, Beliefs, Knowledge, and Values
Harris. (2018). Computational thinking unplugged: Comparing the impact on confidence and competence from analog and digital resources in computer science professional development for elementary teachers.	4 female elementary teachers (K -1st)	Teachers had more evident changes in their confidence when learning CT through analog-focused PD over digital-focused PD. However, there was no evident change in teacher competence of CT between the analog and digital-focused PD.

Table 4 (continued)

Author(s), Year, Article	**Participants**	**Attitudes, Beliefs, Knowledge, and Values**
Hunsaker & West. (2020). Designing computational thinking and coding badges for early childhood educators.	40 preservice and in-service teachers	Earned badges were effective in helping preservice and in-service define CT. Preservice and in-service teacher self-efficacy and attitudes were slightly higher post-survey than pre-survey.
Ketelhut, Mills, Hestness, Cabrera, Plane, & McGinnis. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science	13 Elementary mentor teachers (3^{rd}-5^{th} grades)	After participating in PD, mentor teachers were excited to integrate CT into their curriculum and teaching. Mentor teachers reported that CT was engaging for their students and encouraged them to problem-solve creatively. After participation in PD, mentor teachers believed that integrating CT offered opportunities to engage in best teaching practices for science education. Mentor teachers recognized that some students struggle with the higher-level thinking CT requires and that providing them with the necessary support is difficult.
K.M. Rich, Yadav, & Schwarz. (2019). Computational thinking, mathematics, and science: Elementary teachers' perspectives on integration	12 elementary teachers (primarily 4^{th}/5^{th} grade)	Teachers think of CT as problem-solving. CT and math connections made to their classroom teaching practices were stronger than the teachers' CT and science connections made to their classroom teaching practices. Teachers are concerned about having adequate class time to teach CT. Teachers expressed the difficulty of using developmentally appropriate practices to teach CT.

Table 4 (continued)

Author(s), Year, Article	Participants	Attitudes, Beliefs, Knowledge, and Values
K.M. Rich, Yadav, & Larimore. (2020). Teacher implementation profiles for integrating computational thinking into elementary mathematics and science instruction	8 elementary teachers	Teachers generally engage students in CT through unplugged math and science activities in four ways: 1. Using CT as a general problem-solving strategy 2. Using CT to structure or frame a lesson 3. Highlighting CT through prompting and discussion 4. Using CT to guide teacher planning Teachers who used CT to structure or frame a lesson and highlighted CT through prompting had the most success in providing students with the opportunity to engage in CT.
P.J. Rich, Mason, & O'Leary (2021a). Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking	127 teachers (majority computer lab or library media specialists) 46 certified 76 noncertified	At the end of the year survey, 54 teachers taught students coding/CT once a month or less. However, for 65 teachers, it was expected by their school to be taught, and it was structured into the school day, so they taught coding /CT once per week or more. Teachers began the PD with a fairly high-value belief toward the importance of teaching coding (slightly higher than 5 on a 6-point scale). After a year of PD, teachers' confidence in their ability to teach coding increased from 3.84 to 4.84, statistically significant with a large effect size. After a year of PD, teachers' self-efficacy for CT increased from 4.06 to 4.27, statistically significant with a moderate effect size. After a year of PD, teachers did not report a "confident" or "very confident" level in any area of CT. Teachers reported a lack of confidence in their ability to foster computational perspectives.

Table 4 (continued)

Author(s), Year, Article	Participants	Attitudes, Beliefs, Knowledge, and Values
P.J. Rich, Larsen, & Mason. (2021b). Measuring teacher beliefs about coding and computational thinking	122 teachers 46 certified 76 noncertified	Factors that affect teachers' beliefs about teaching CT in the elementary classroom include: - Value beliefs about computing - Coding efficacy - Computational thinking efficacy - Teaching efficacy Teachers' computational thinking self-efficacy and coding efficacy are only weakly correlated.
Yadav, Krist, Good, & Caeli. (2018). Computational thinking in elementary classrooms: measuring teacher understanding of computational ideas for teaching science	9 teachers	Teachers conceptualized CT as the following: - CT is problem-solving - CT is programming - CT entails using logic - CT entails data collection - CT entails algorithms - CT entails pattern recognition - CT assists in prediction and efficiency

A literature review (see Table 4 above) identified three key findings. First, teachers respond better to analog or "unplugged" approaches for learning and developing their confidence with CT than digital approaches. Second, teachers' efficacy in CT lags behind teachers' coding efficacy. Third, teachers' understanding of what CT is varies. These key points are discussed below.

Analog Over Digital Interfaces and Methods to Increase Teacher Confidence in CT

In his dissertation research, Harris (2018) conducted a quantitative study with female kindergarten and first-grade teachers to compare the use of the Robot Turtles board game to Scratch Jr. in computational thinking professional development. He analyzed the two modalities' effect on teacher gains regarding confidence and competence with computational thinking. He found that teachers had a higher confidence level when engaging with analog approaches in professional development than the digital approach. However, when assessing teachers' competence regarding CT, there was no significant difference between the analog and digital modalities. These findings give a reason for us to reconsider how we approach teaching professional development regarding computational thinking.

Ketelhut et al. (2020) also provided computational thinking professional development to mentor teachers and their mentees (pre-service teachers). The after-school teacher inquiry group first took an unplugged approach and added digital simulations later in the year. The researchers found that the mentor teachers were excited to use and implement computational thinking into their teaching practices. Additionally, the mentor teachers desired more examples of unplugged CT activities that could be practically implemented in their classrooms.

After interviewing teachers to help guide the design and development of professional development, K.M. Rich et al. (2019) decided to use an unplugged or analog approach in their professional development sessions. This decision was made to leverage the connections the teachers in their study made between their current teaching practices and CT.

Although conducted with adults, these studies support prior research that has suggested that children, especially girls, prefer analog activities over digital activities when learning computer programming. (Horn et al., 2009, Horn et al., 2012). These studies suggest that when planning professional development to prepare educators to develop the competence of and the ability to teach CT, an analog first approach should be considered when designing and developing the professional development.

Computational Thinking Self-Efficacy

Hunsaker and West (2020) designed, piloted, and evaluated computational thinking and coding badges for early childhood educators and preservice teachers intending to develop competency in CT through a badging program. The design focused on using developmentally appropriate practices for early childhood education. The evaluation found that the badging program was helpful for educators to earn computational thinking credentials and helped to increase their CT knowledge. However, concerning teaching CT, educators' self-efficacy was “only slightly more positive in the post-survey than the pre-survey” (Hunsaker & West, 2020, p.15). Nevertheless, educators who participated in the CT badge improved significantly when defining CT. Interestingly, although most educators (88%) were successful in earning their CT badge, there was only a minor change in their CT self-efficacy, while there was a significant change in their understanding of CT.

Additionally, P.J. Rich and various colleagues have researched educators' computational thinking self-efficacy. Using a mixed-methods design, P.J. Rich et al. (2021a) measured the effect of continuous professional development. Specifically, they strived to understand the effect of PD on teachers’ self-efficacy in teaching CT and coding concepts in elementary school. They surveyed 291 teachers in eight different school districts using the BootUp Professional Development program. They found that “despite having very little to no experience teaching coding, teachers’ value beliefs began relatively high (slightly above 5 on a 6-point scale). This finding indicates that they began the training believing it is important for young children to learn how to code" (P.J. Rich et al., 2021a, p.7). Although teachers believed coding was important, many were at a novice level regarding CT, coding skills, and experience.

Teachers' computer science and coding self-efficacy demonstrated strong growth after a full year of professional development. However, interestingly, CT self-efficacy did not see the same type of growth. Indeed, teachers' CT self-efficacy remained unchanged at several research sites participating in the study. After a year of professional development, "teachers showed the slowest growth in their ability to foster computational practices and perspectives" (P.J. Rich et al., 2021a, p.13). P.J. Rich et al. have yet to identify why there was little growth in CT self-efficacy compared to CS/coding self-efficacy. They concluded that more than one year of professional development was needed. However, I wonder if, similar to the studies mentioned above, a more continuous PD model is needed for teachers to grow. This model provides ongoing professional development to help teachers maintain and enhance their learning.

To help ascertain the various factors that affect teachers' beliefs about computing in elementary classrooms, P.J. Rich, Larsen, & Mason (2021b) developed the "Teacher Beliefs about Coding and Computational Thinking" (TBaCCT) scale. This scale was developed while conducting research in three different states where the BootUp PD was implemented. This research included 122 educators comprising 46 certified teachers and 76 paraprofessionals from eight school districts. P.J. Rich et al. (2021b) found that educators' coding and CT self-efficacy correlated weakly. Additionally, their analyses showed statistically significant increases in computing, CT, and coding self-efficacies.

However, the effect size for change in CT self-efficacy was only moderate (0.52), while the effect size for changes in coding self-efficacy was much higher (0.89). The TBaCCT tool can quickly assess teachers' professional development and teacher change around computing and CT. Notwithstanding, more research is needed to understand what makes CT and coding self-efficacy so different and why educators struggle with CT self-efficacy more.

Teachers’ Understanding of CT is Varied

Lastly, a review of the studies found that teachers’ understandings of computational thinking vary. Yadav et al. (2018) used causal reasoning to design assessments to measure teachers’ situated understandings of CT. Vignettes and open-ended questions were created to measure elementary teachers' shifts in thinking about CT. Nine teachers participated in professional development to learn how to integrate CT into their science curriculum. Professional development included a two-week summer workshop, bi-weekly afterschool sessions throughout the school year, and another two-week summer workshop.

Researchers used previous literature and grounded analysis to categorize teachers’ thinking and conceptions around CT. These include CT is (defining), CT involves (essential aspects), and CT aids in (practices supported by CT). Yadav et al. (2018) found that teachers conceptualize CT in various ways (some of which may not accurately represent CT). First, teachers mainly define CT as programming or problem-solving. Second, teachers believe that essential aspects of CT include using logic, data collection, algorithms, and pattern recognition. Lastly, teachers felt that CT was mainly used for prediction and efficiency. Surprisingly, researchers found that the teachers had more CT knowledge than expected, given previous research (e.g., Sands et al., 2018). Yadav et al. (2018) claimed this may be because all the teachers in their sample were science teachers. In addition, the recent adoption of the Next Generation Science Standards (NGSS) (which include CT) and the training teachers have received may explain why their CT knowledge was more significant than expected.

Additionally, K.M. Rich et al. (2020) conducted a study with eight elementary teachers to explore how they used unplugged math and science activities to engage their students in computational thinking. They used Carroll's (1963, 1989) Opportunities to Learn theory as their lens for their data. K.M. Rich et al. (2020) used observation notes, videos of CT lesson enactments, and teacher lesson plans. First, they searched for the four CT practices (abstraction, decomposition, patterns, and debugging). Next, they sorted the teachers' CT opportunities into three categories- framing, prompting, and inviting reflection, demonstrating their knowledge and views of CT. Based upon this analysis, the researchers identified four implementation profiles of teachers, including 1) using CT as a general problem-solving strategy, 2) using CT to structure their lesson, 3) highlighting CT through prompting, and 4) using CT to guide their planning (K.M Rich et al., 2020).

K.M. Rich et al. (2020) concluded that just inviting reflection does not allow students to use CT. They found that teachers who constructed their lessons around CT and prompted students throughout were most successful in creating opportunities for their students to engage in computational thinking. As K.M. Rich et al. (2020) concluded, these findings are significant because it is not enough for teachers to invite their students to reflect on CT because, at the point of reflection, CT has already been used. However, teachers can create intentional moments for their students to use CT practices by framing and prompting them throughout a lesson or activity to truly support students in engaging with and using CT practices.

Lastly, to better understand the myriad of ways teachers view computational thinking and how it connects to their current classroom teaching practices, K.M. Rich et al. (2019) conducted a qualitative study to determine teachers' perspectives on integrating CT with mathematics and science. They identified several themes within the data. First, teachers most often think of CT as problem-solving. Second, teachers' CT and math connections were stronger than their CT and science connections. K.M. Rich et al. (2019) found that the teachers connected CT to math more than science because of a shared vocabulary between CT and math (e.g., algorithm, automation with fact fluency, etc.). Third, teachers are concerned about having time in class for CT and being able to use developmentally appropriate practices to teach it to their elementary students. Their findings suggest that math may be a more viable avenue for teachers to integrate CT since they can already identify CT and math connections. Clarke-Midura and colleagues (2022) are currently developing computer science instruction integrated with an elementary math program to support elementary teachers in integrating mathematics and CS/CT.

The studies above have shown that educators' attitudes, beliefs, knowledge, and values surrounding computational thinking vary. The three themes identified above serve as a springboard for further research on supporting teachers' development of self-efficacy, understanding, and abilities surrounding computational thinking.

Evaluation Framework

Evaluation of a Professional Development Program

Various evaluation methodologies have been developed to help judge the merit or worth of learning processes and environments. Three evaluation models, in particular, are most often used to evaluate teachers' professional development (Hanover Research, 2014). They include the Clarke-Hollingsworth model (Clarke & Hollingsworth, 2002), the Kirkpatrick Training Evaluation Model (Kirkpatrick & Kirkpatrick, 2016), and Guskey’s Five Levels of Professional Development Evaluation (Guskey, 2000, 2002). In this dissertation, Guskey’s Model, the Five Levels of Professional Development Evaluation (Guskey, 2000, 2002), is used to evaluate the MSD STEM/CT Professional Development Program.

Guskey’s Five Levels of Professional Development Evaluation

Guskey’s Five Levels of Professional Development Evaluation model (see Table 5) approaches teacher change differently than other professional development models or studies (Guskey, 1986). First, Guskey’s views on changes in teacher attitudes and knowledge are predicated upon implementing the knowledge gained in professional development and seeing results in student learning (Guskey, 1986). When teachers see that the new method or technique is working, they are more likely to continue to apply it.

Second, Guskey developed a professional development evaluation model that builds upon Kirkpatrick’s evaluation model but adds a fifth level to account for student learning outcomes. Guskey’s levels include 1) participants’ reactions, 2) participants’ learning, 3) organization support and change, 4) participants' use of new knowledge and skills, and 5) student learning outcomes (Guskey, 2000, 2002).

Level One.

Guskey's level one examines the reactions from participants regarding their experience with professional development. Measuring participants' satisfaction is an important indicator when deciding to improve the professional development experience. Although the feedback may seem trivial, even minor changes can help improve teachers' attitudes and experiences in a professional development program.

Level Two.

Level two focuses on measuring participants' knowledge and skills obtained throughout the professional development to determine if the learning goals of the professional development were achieved. Learning goals can be measured through formative assessment, summative assessment, observation, portfolios, or any combination of formal and informal assessment strategies. However, these measurements require well-established learning objectives/outcomes to develop the correct assessment forms. In addition, it is important to "consider the possibility of unintended learning outcomes, both positive and negative" (Guskey, 2005, para. 27). As an evaluator, how might one capture unintended learning outcomes? The information gathered at level two can assist the evaluator in beginning to generate recommendations for improvements to the design, structure, delivery, or methods of the professional development program that will be built upon as additional information is gathered on levels three through five.

Level Three.

Unlike Kirkpatrick's model, Guskey's level three evaluates the change and support provided by the organization. Without sufficient organizational support, even well-designed professional development can fail. Unfortunately, if no professional development is aligned with the school or district culture, policy, or goals, the level one and two evaluation results become irrelevant. Professional development goals must be aligned with the school or district policies and mission to avoid potential clashes in mission or purpose and thwart teachers' implementation and change efforts. Additionally, teachers need encouragement and support to implement practice changes and access needed resources.

Level Four.

At level four, the evaluator examines if participants have changed their professional practices based on the knowledge and skills gained during professional development. Before beginning to assess level four, the evaluator needs to ensure the teachers have sufficient time to implement the instruction learned in professional development so that changes in their professional teaching practices can be observed and measured. Guskey emphasizes that implementation often happens gradually or at various rates amongst teachers, so evaluators may need to measure teachers' progress at various time points (Guskey, 2005). Providing ample time for implementation and opportunity for teacher reflection will help evaluators truly measure how teachers have used their new knowledge and skills in the classroom.

Level Five.

Lastly, at level five, student learning outcomes are evaluated. Did the professional development affect the participants' students' learning? Student learning can be evaluated in various ways, including standardized tests, class assessments, grades, portfolios, etc. Guskey instructs that other measures and outcomes such as affect, psychomotor, and indicators may also be considered. In addition, he suggests that evaluators consider interviews with students, teachers, parents, and administrators to help determine the overall impact of the different aspects of the professional development program (Guskey, 2005).

Table 5

Guskey's Five Levels of Professional Development Evaluation (Adapted from Guskey, 2002)

Level	**Questions that can be Answered**	**How to Collect Information**	**What to Assess or Measure**	**How to Use the Information**
Level 1: Participants' Reactions	Did they enjoy the professional development? Were they comfortable in the space? Was the professional development helpful to them as a teacher?	Survey at the end of the professional development sessions	Satisfaction with the professional development	Improve the professional development sessions and content
Level 2: Participants' Learning	Did they develop the knowledge and skills needed?	Surveys. Interviews, Artifacts, and Roleplays	New skills and knowledge	Improve the professional development content, activities, etc.

Table 5 (continued)

Level	Questions that can be Answered	How to Collect Information	What to Assess or Measure	How to Use the Information
Level 3: Organizational Support & Change	Did they have the support and resources they needed? Were their concerns and issues addressed? Did the organization experience impacts from the implementation?	Surveys Interviews with district leaders, school leaders, and Focus groups	Support and recognition	Inform future professional development efforts and document support
Level 4: Participants' Use of New Knowledge and Skills	Was new knowledge applied? Did they implement new skills and concepts learned?	Surveys Interviews Observations Reflection	Quality of application	Improve professional development and document applications of PD
Level 5: Learning Outcomes of Students	What was the impact on students?	Interviews Artifacts Observations	Caliber of implementation	Improve professional development and document student impacts

Summary

Paraprofessionals play a key role in many K-6 schools across the United States. There are varying terms for paraprofessionals, including paraeducator, teacher assistant, teacher aide, instructional assistant, etc. Paraprofessionals work under a licensed teacher's direction if they assist with providing instruction to students. Paraprofessionals are increasingly responsible for CS education, including computational thinking. There are varying definitions of computational thinking, with most defining computational thinking as a cognitive ability, a thought process, or a problem-solving approach. Many researchers have identified the key components of computational thinking (often called "practices" or "concepts") that generally include (but are not limited to) abstraction, algorithmic thinking, decomposition, and pattern recognition.

However, the research on CT professional development for paraprofessionals is lacking. There is a body of research on CT professional development for K-6 teachers. Much of the empirical research has smaller sample sizes, so findings may or may not be generalizable. These studies on PD show that computational thinking is often taught to K-6 students through integration into the general education curriculum. Many studies use Communities of Practice as a professional development model or provide continuous or extended professional development. However, many studies experienced or identified difficulties with computational thinking, including developing teacher self-efficacy, incorrect assumptions or misconceptions, and the learning curve associated with CT. Additionally, there is a body of research on K-6 teachers' attitudes, beliefs, knowledge, and values about CT. The research shows teachers respond better to an "unplugged" approach to learning CT. Their self-efficacy is challenging to develop, and their understanding of CT varies.

CHAPTER III

METHODOLOGY

This chapter presents the methodology and design used in this evaluation. First, I present my positionality as a researcher and evaluator, followed by my evaluation approach. Then, I present the logic model guiding this evaluation. Next is a brief overview of the context and the MSD STEM/CT Professional Development Program. Then, I describe my participants and recruitment efforts. Following participant and recruitment, I describe my data collection procedures in-depth and then describe data sources and analysis. I then present the analytical strategy used and conclude the chapter by explaining the ethical considerations taken, my trustworthiness as a researcher and evaluator, and finally, the limitations of this evaluation.

Positionality

Because this is a qualitative evaluation, I am considered an instrument of data collection as the evaluator/researcher (Denzin & Lincoln, 2003). My role is purely evaluative. As I interviewed participants and analyzed data and artifacts, I evaluated how well the STEM/CT Professional Development Program met the short-term outcomes outlined in the logic model guided by the evaluation questions. As a former teacher, I brought a specific lens to the evaluation due to my previous experience in the classroom and elementary school environment.

Evaluation Approach

I drew from Guskey's (2002) Five Levels of Professional Development Evaluation Model (see Table 5 & Chapter 2). I evaluated the STEM/CT Professional Development Program, focusing on levels one, two, and four. Levels three (organizational support) and five (student learning) of Guskey's model fell outside the scope of this evaluation. Evaluation questions were developed based on the logic model (see Figure 1).

Logic Model

A logic model illustrates the program activities and processes that lead to program outcomes. It can be used as a visual way to demonstrate thinking and "articulate a theory of action" (Archibald et al., 2011, p. 14) and is often used as a tool in evaluation work. They have been used in many government, non-profit, and private sectors and various educational settings. For example, they are used to research teacher education programs because they help explicitly link programs and outcomes (Newton et al., 2013).

The logic model is a visual representation of a program that demonstrates the intended relationships between investments and results from the implementation of a program (Taylor-Powell & Henert, 2008). Logic models illustrate the activities or processes within a program that work to achieve outcomes and promote change (Newton et al., 2013). A logic model shows program plans and program outcomes. Evaluators using logic models identify key variables of a program to evaluate and use the logic models as a credible framework for reporting (Knowlton & Phillips, 2012). A logic model does not need to be an exact replica of the program but is considered a general "roadmap" of the program philosophies, resources, operations, and outcomes (Taylor-Powell & Henert, 2008).

A program logic model was employed for this study (see Figure 1). A program model communicates all the program's elements (resources, activities, people, etc.) and their desired outputs and outcomes. Once a logic model is fully developed, it assists the evaluator in looking at each piece of the model and determining what types of data must be collected and how to measure outcomes (Archibald et al., 2011). By developing a program logic model, I could effectively identify specific activities to evaluate the STEM/CT Professional Development Program and the desired outcomes.

Logic models should include four main components: inputs, activities, outputs, and outcomes listed in a linear order (Savaya & Waysman, 2005). *Inputs* are the resources (financial, human, organizational, etc.) that will be invested in a program. *Activities* are what the program does with the provided inputs and are key elements of program implementation. The direct results of activities are the *outputs* of the logic model. Lastly, *outcomes* are the changes based on the program activities and can build upon each other over time. For example, "it may be expected that new knowledge and increased skills (immediate outcomes) will lead to modified behavior (intermediate outcomes), which will lead, in turn, to improved condition (long-term outcome)" (Savaya & Waysman, 2005, p. 88).

In this dissertation, I evaluated the MSD STEM/CT Professional Development Program for paraprofessionals in terms of how it helps participants develop an understanding of computational thinking and prepares them to integrate and teach computational thinking in the STEM specialty. I developed the following problem statement after discussing the goals of the Computing Partnership Grant with the Utah STEM Action Center and the STEM/CT Professional Development Program with the Marshall School District. The logic model I developed outlines the STEM/CT Professional Development Program (see Figure 1) and the parts of the program that will be evaluated as part of this dissertation project (see Figure 2). The logic model elements included in this evaluation will be explained throughout this chapter. The assumptions below helped shape and determine the elements included within the logic model.

Assumptions

The MSD identified the following assumptions and taken from the MSD STEM Action Center grant application. The following paragraphs summarize these assumptions. One main objective of their professional development program is to build capacity and teacher confidence and abilities for the integration of computational thinking and STEM resources (i.e., the MSD lending lab) as well as connect paraprofessionals to the widely available resources to expand access to computing education within elementary schools across the district. This expansion will directly affect the equitable access given to students to be exposed to computing education and gain computing skills. The goal is for computing activities to shift from district-led to school-based programs and resources by the end of the third year of implementing the STEM/CT program.

Existing Resources

The district grant application indicated that they had several existing resources, including the lending lab, Digital Learning Specialists (DLS) team, and after-school programs. The MSD lending lab currently provides classroom sets of STEM tools such as Spheros, Ozobots, littleBits, MakeyMakey, ClassVR, and Micro:bits. Grant funds will also be used to purchase additional STEM tools for the lending lab as part of the PD and STEM specialty. The DLS team has several digital learning courses for teachers and paraprofessionals to earn endorsements and certificates for several digital learning platforms, including Nearpod, Adobe, and Canvas. Each school is also assigned a Digital Teacher Leader (DTL) who can serve as a resource for training and support with digital learning. Lastly, several elementary schools across the district currently participate in STEM/CS programs such as Girls Who Code, First Lego League Robotics, and MESA, where students are also given opportunities to participate in computing activities.

Identified Challenges

MSD identified several challenges as they begin the STEM/CT Professional Development Program. First, there was a lack of financial support for these tools and programs at each school. Second, in their grant application, the district reported that paraprofessionals had limited capacity and confidence in teaching computational thinking and STEM in an integrated way. Additionally, MSD reported that retaining STEM instructional assistants can be challenging.

STEM/CT Professional Development

The district grant application also outlined its timeline for implementation. Year one (2020-2021) and year two (2021-2022) of the PD program focus on skill-building and training on the STEM tools available through the MSD lending lab. There was a focus on free computer science resources (code.org, CS First) and contracted training through educational partners such as Skill Struck and Sphero. These resources directly built paraprofessionals' capacity and confidence in providing opportunities for students to engage in computational thinking. Additionally, paraprofessionals used Canvas to access resources, materials, and lesson plans. This usage included sharing their lesson plans and resources they created.

MSD STEM/CT Professional Development Program Logic Model

Inputs

There are three main inputs in this program logic model. The first input is the STEM/CT professional development provided by the MSD. The second input is the Elementary Professional Learning Community (PLC) Canvas Course developed by the MSD STEM & Computer Science consultant. The third input in the logic model is the MSD STEM Lending Lab. The STEM Lending Lab consists of several STEM tools and toys that will be used throughout the program. These tools and toys are described in more depth below.

Activities

There are four main activities for paraprofessionals to participate in to achieve the short-term, intermediate, and long-term outcomes outlined in this logic model. First is attending a multi-day summer professional development. Second is attending the monthly Elementary STEM PLC meetings with the MSD STEM & Computer Science consultant. Next is to participate in the Elementary STEM PLC Canvas course. The last is to teach the STEM specialty to MSD K-6 students. This teaching is where paraprofessionals can apply the skills and concepts they are learning in the previous three activities. Each activity is described in more depth below.

Short-Term Outcomes

The logic model presents four short-term outcomes that should result from the inputs and activities outlined within the model. Three of these four outcomes are included in this evaluation scope. The first desired short-term outcome is that paraprofessionals will have a positive learning experience participating in the STEM/CT program. Second, paraprofessionals will be able to develop an understanding of computational thinking. The next desired short-term outcome is that paraprofessionals can facilitate STEM/CT integrated lessons using computational thinking components. Lastly, MSD K-6 students will have hands-on experiences with STEM tools and toys during their STEM specialty class taught by the paraprofessionals (not included in the evaluation scope; see Figure 2). These short-term outcomes will lead to and help develop two intermediate outcomes.

Intermediate Outcomes

After paraprofessionals have established an understanding of computational thinking and can use the STEM tools and toys to facilitate STEM/CT integrated lessons using various CT components, the district expected that the paraprofessionals would be able to recognize CT in practice. This recognition will allow them to design and teach STEM/CT integrated lessons using various CT components. These two intermediate outcomes will help achieve three long-term outcomes.

Long-Term Outcomes

There are three long-term outcomes presented in this logic model. The first desired long-term outcome is that paraprofessionals can confidently teach CT components through integrated STEM activities for MSD K-6 students. Next, students will learn and reach proficiency in computational thinking through the various CT components. Last, the district desired that students demonstrate mastery of the Utah K-5 CS standards CT strands. These long-term outcomes will ultimately help all MSD K-6 students receive equitable access to opportunities to be exposed to and learn computing concepts and be taught the K-5 CS standards.

Figure 1

STEM/CT Professional Development Program Evaluation Model

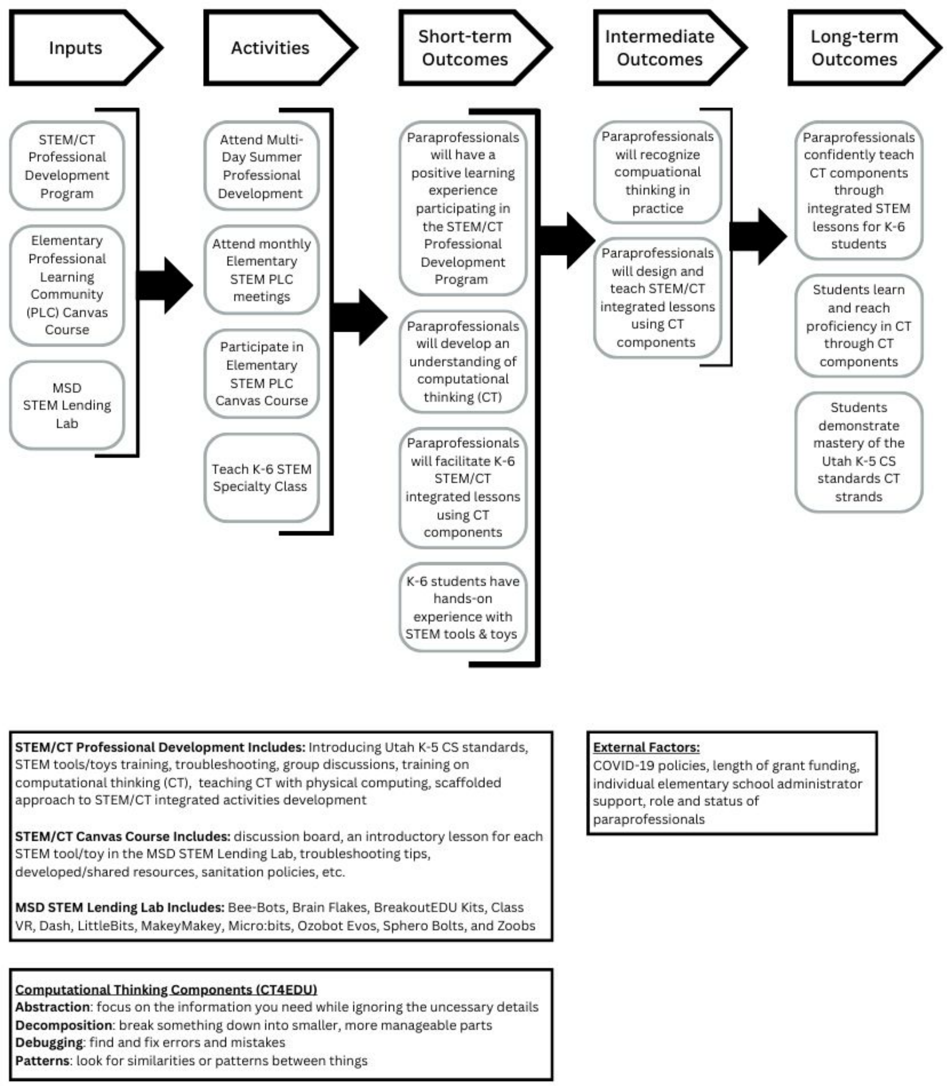
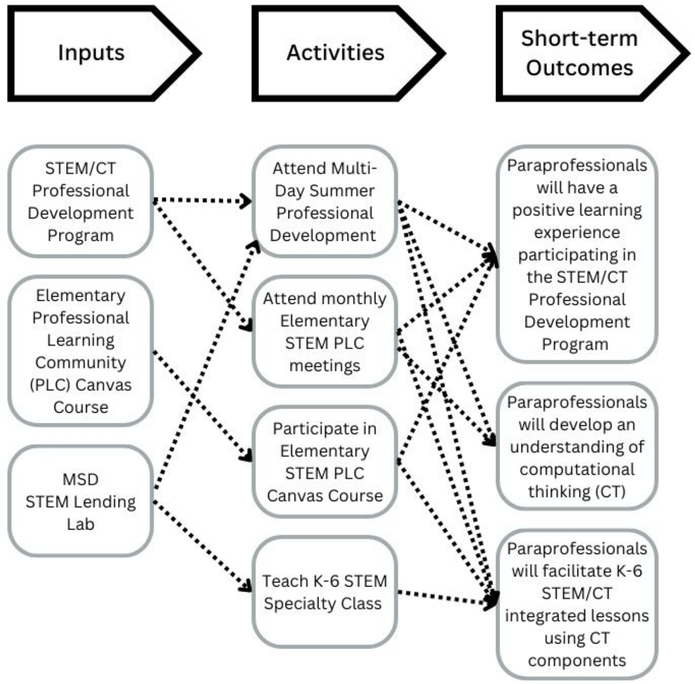


Figure 2

STEM/CT Professional Development Program Evaluation Scope



Context

STEM Action Center of Utah & the Computing Partnership Grant

The STEM Action Center of Utah created the Computing Partnership Grant (CPG) program to find gaps in computing education and opportunities for students throughout the state. Funds from the grant are used to provide solutions to computing education gaps, especially those that provide access to STEM and computing education in underserved populations (STEM Action Center, n.d.). In 2020, the Marshall School District was awarded a Computing Partnership Grant. This grant provides funding for three years (2021-2023) to be implemented in three phases (one per year). The district is using this grant to fund training and STEM tools to integrate computational thinking into a K-6 STEM specialty class in all elementary schools throughout the district.

Marshall School District

The Marshall School District is located in a large suburban area that serves PK-12 students, serving a student population from one of the fastest-growing regions in Utah (U.S. Census Bureau, 2021). The school district currently has 39 elementary schools with almost 28,000 K-6 students. There is an average 1 to 26 teacher/student ratio (schooldistrict.org). In addition, 75% of the students they serve are White/Caucasian, with 21% of economically disadvantaged families and 3% of students experiencing homelessness (USBE, 2020).

MSD STEM & Computer Science Consultant

The Marshall School District STEM & Computer Science Consultant oversees the K-6 STEM/CT Professional Development Program and the Computing Partnership Grant from the STEM Action Center. She is responsible for planning and teaching the summer and monthly professional development sessions and mentoring and coaching paraprofessionals who may be struggling. In addition, she answers questions any questions paraprofessionals may have, provides technical support, and coordinates the sharing of STEM tools throughout the district.

Professional Development Program

STEM/CT Professional Development Activities

As part of the elementary school day in the Marshall School District, students participate in a weekly (60 minutes) or bi-weekly (30 minutes) STEM specialty class.

Paraprofessionals teach this class. These paraprofessionals are now tasked with teaching Utah K-5 CS standards, focusing on the computational thinking (CT) strand for each grade level as part of the STEM specialty in elementary schools throughout the district. The school district is providing a STEM/CT Professional Development Program to support paraprofessionals in teaching CT in the STEM specialty using STEM tools. This program is being phased in over three years with the goal of all 39 elementary schools' STEM specialty paraprofessionals participating in the PD.

Professional Development Program

According to the logic model developed in conjunction with the MSD STEM & Computer Science Consultant, the goal of the STEM/CT Professional Development Program is for paraprofessionals to be able to effectively define computational thinking and design STEM/CT integrated lessons to engage elementary students in building STEM and computational thinking skills (K. Taylor, Personal Communication, 2020).

The STEM/CT Professional Development Program follows a continuous/extended professional development model. Cohort B paraprofessionals participated in more than 30 hours of summer professional development (see Table 6), where they learned all ten STEM tools in the district STEM Lending Lab. They also had a monthly three-hour professional development (see Table 6 below) where they were introduced to the Utah K-5 CS Standards, which focused on the computational thinking strand for each grade level and received content focused on computational thinking and the “technology” in STEM. In addition, there was a required Canvas course for these paraprofessionals with resources and information needed for PD sessions throughout the school year. The paraprofessionals also created STEM/CT integrated lessons and activities as part of the CPG. They will be added to a lesson plan bank to be used in future years. In the monthly PD sessions, paraprofessionals learned how to create STEM/CT integrated lessons incorporating tools from the STEM Lending Lab. Paraprofessionals were given opportunities in select professional development sessions and the Elementary STEM PLC Canvas Course to share their lesson ideas and receive formal feedback on their lesson plans.

Table 6

Professional Development Sessions

Date/hours	PD Goals	PD Activities	Participant Attendance
August 12, 2021 3 hours	Learn how to use the STEM Lending Lab tools.	Canvas Login Google Drive Access Ozobot EVO Ozobot Color Code	7
August 13, 2021 6 hours	Learn how to use the STEM Lending Lab tools. Learn how to write and submit a grant to the STEM Action Center.	STEM Mission/CT USBE CS Standards Ozobot Evo Ozoblockly Micro:bit Bee-Bot STEM Action Center	7
August 18, 2021 5 hours	Learn how to use the STEM Lending Lab tools.	BreakoutEDU Unplugged-“Littles” (KEVA, Zoob, Brain Flakes, Coji) Dash	8
August 19, 2021 5 hours	Learn how to use the STEM Lending Lab tools.	littleBits Makey Makey Breakout Digital	8

Table 6 (continued)

Date/hours	PD Goals	PD Activities	Participant Attendance
August 24, 2021 5 hours	Learn how to use the STEM Lending Lab tools.	Class VR Makey Makey Sphero Bolt Sphero RVR	8
August 25, 2021 3 hours	Learn how to use the STEM Lending Lab tools.	Nearpd ClassVR	7
August 26, 2021 3 hours	Learn how to use the STEM Lending Lab tools.	Drones Planning Activity Sharing Resource Creation	8
August 27, 2021 3 hours	A brief overview of the lesson plan, CS standards, and I can statements. Learn about computational thinking. Learn how to incorporate social-emotional learning into the classroom.	SEL Computational Thinking CS Standards I Can Statement	7
September 17, 2021 3 hours	Resolve any questions/concerns with STEM Lending Lab tools. Practice writing a STEM grant.	Review tech tools Troubleshooting Grant writing	7
October 2021	*Canceled*	*Canceled*	-
November 12, 2021 3 hours	Understand the fundamental concepts of computational thinking.	Presentation and discussion about CS standards and computational thinking	5

Table 6 (continued)

Date/hours	PD Goals	PD Activities	Participant Attendance
	Learn how to write a STEM/CT integrated lesson plan.	Teach the STEM/CT lesson plan format.	
December 3, 2021 3 hours	Walk through the lesson plan template.	Sharing lesson ideas and resources in Google Docs	5
	Sharing is Caring with Cohort A	Six unplugged STEM lessons -STEM Towers -Straw Airplanes -Perfect Square -Coding unplugged -Thaumatrope -Paper Bridges	
December 10, 2021 1 hour	Zoom Q & A for STEM/CT lesson plans		2
January 14, 2022	Optional day- lesson plans		4
February 18, 2022 4 hours	Review the computational thinking standards in depth.	Discuss "I can" statements.	7
	Receive training on Infini-D.	Discuss scaffolding for STEM activities.	
		Review CT standards Infini-D	
March 4, 2022 4 hours	Discuss why computer science is important.	Infini-D mission as a group	8
	Learn how to use Skill Struck.	STEM night debrief	
		Skill Struck training provided by Skill Struck	

Table 6 (continued)

Date/hours	PD Goals	PD Activities	Participant Attendance
March 11, 2022	Science Partners Training		7
April 1, 2022	Science Partners Training		6
April 8, 2022	Learn how to use Skill Struck.	Skill Struck training provided by Skill Struck	7
April 29, 2022	Learn how to use Skill Struck	Skill Struck training provided by Skill Struck	8

Professional Development Materials

Elementary STEM PLC Canvas Course

There is an Elementary STEM PLC Canvas Course for paraprofessionals (cohorts A and B) developed by the MSD STEM & Computer Science Consultant. The Canvas course is used to share PD recordings and materials, basic lesson plans for each STEM tool in the MSD STEM Lending Lab, unplugged computing activities, etc. It also is where paraprofessionals submit lesson plans written for the different STEM tools. Paraprofessionals were each required to submit at least one lesson plan. Finally, it is also a place for the paraprofessionals to share resources.

MSD STEM Lending Lab

School district and CPG funds were used to purchase several technology tools. The tools comprise the MSD STEM Lending Lab. The following technologies/tools are currently included in the MSD STEM Lending Lab: Bee-Bots, Brian Flakes, Breakout EDU Kits, Class VR, Dash, LittleBits, MakeyMakey, Micro:bits, Ozobot Evo, Sphero

Bolt, and Zoobs (see Table 7). These tools are assigned to each school in a rotation that paraprofessionals used in their STEM specialty class with K-6 students throughout the school district. One introduction lesson is provided for each STEM tool that introduces the students to the tool/technology and how to use it. As part of their school's participation in the MSD Lending Lab, paraprofessionals commit to the following:

1. Attend the STEM tool training sessions;
2. Use the STEM tools with students;
3. Incorporate CS/CT vocabulary and concepts; and
4. Write a lesson plan for the STEM tools.

Table 7

STEM Tool Descriptions

STEM Tool	Description
Bee-Bot	A Bee-Bot is a programmable robot designed for early childhood and lower elementary students. Students program the bee by pushing the buttons (forward, backward, turn right, turn left) on top and then pushing "Go" to execute the commands.
Brain Flakes	Brain Flakes are small, interlocking plastic discs students click together to build 2D and 3D objects, structures, etc. Brain Flakes aims to help students with spatial learning and explore engineering concepts.
Breakout EDU Kit	A Breakout EDU Kit has everything students need to play over 900+ educational puzzles on the Breakout EDU platform. The puzzles are designed to help students develop communication, collaboration, critical thinking, and creativity.
Class VR	Class VR uses virtual and augmented reality to help students visualize and understand various STEM, history, literacy, and arts topics. Students can also explore and research in multiple virtual worlds/scenes.

Table 7 (continued)

STEM Tool	Description
Dash	Dash is a robot designed for elementary students. Students use various scaffolded, age-appropriate Dash apps to code or give voice commands to the robot. In addition, the robot can interact and respond to the environment around it and other robots.
LittleBits	LittleBits are modular/electronic building blocks that snap together using magnets. LittleBits are used to help students explore circuitry and electronics. The blocks are color-coded by their function.
Makey Makey	The Makey Makey is an invention kit. Students can design their own controllers using every day, conductive materials. The MakeyMakey can also be used to create circuits and build sensors.
Micro:bits	The Micro:bit is a tiny computer. It helps students learn how software and hardware can work together. The Micro:bit has an LED light display, sensors, input/output features, and buttons. Students can use various programs to code the Micro:bit to interact and build new things.
Ozobot Evo	The Ozobot Evo is a small, round robot. It can be coded using colored markers or online with Blockly.
Sphero Bolt	The Sphero Bolt is a robotic ball designed for upper elementary students. It has several programmable sensors, including a compass, light sensor, gyroscope, accelerometer, motor encoders, and infrared communications. Students code the robot with the Sphero Edu app by drawing code or using block-based coding.
Zoobs	Zoobs are a building set designed for elementary students. Students snap together the pieces with swivel joints, sockets, and pivoting axles, allowing creative building.

Participants & Recruitment

Recrutiment of Paraprofessionals

The paraprofessionals participating in the STEM/CT Professional Development Program for this evaluation are part of the Elementary STEM Professional Learning

Communities (PLC) Cohort B. They are part of the Phase 2 implementation of the program. Cohort A is part of the Phase 1 implementation and will participate occasionally with Cohort B but are not participants in this evaluation. Paraprofessionals from Cohort B teach at fourteen different elementary schools throughout the district. Eighteen paraprofessionals started the STEM/CT Professional Development Program as part of Cohort B in the 2021-2022 school year. All paraprofessionals in Cohort B received an initial email and two follow-up emails with invitations to participate in the evaluation. Emails included a digital consent form and a pre-survey. Of the 18 paraprofessionals from Cohort B, 12 (67%) paraprofessionals consented to participate in this evaluation study. I obtained their consent for participation in this study before the evaluation's start, following approval by Marshall School District and Utah State University (USU) Institutional Review Board (IRB). All participants completed their pre-surveys before the first day of summer professional development. In February 2022, two paraprofessionals in this study quit their jobs, dropping the study down to ten participants. At the end of the year, only eight of the ten participants completed the post-survey and participated in the reflective interview. Therefore, only data from 8 of 18 paraprofessionals (44%) were included in this evaluation study.

Paraprofessional Participants' Backgrounds

Paraprofessionals are only required to have a high school diploma and six months of experience in a classroom environment to teach the STEM specialty class in the district. However, although they are not licensed teachers, more than half of the paraprofessionals hired for this position hold an associate or bachelor's degree. Of the eight paraprofessionals participating in this study, all were female; all but one were White, and all but two were more than 40 years old (see Table 8). Additionally, the range of prior teaching experience varied across participants (see Table 8). Some had never taught before, while others had prior teaching experience as a paraprofessional within the district in a different capacity (e.g., special education aide).

Table 8

Demographics, Education, and Experience of Participants

Pseudonym	Gender	Age	Race/Ethnicity	Education	Prior Teaching Experience
Marissa	Female	31-40	Native American/Pacific Islander	Bachelor of Science	Yes
Hailey	Female	41-50	White	Bachelor's degree	No
Sophia	Female	51-65+	White	n/a	No
Charlotte	Female	41-50	White	Bachelor of Science	Yes, preschool aide, special education aide
Ana	Female	51-65+	White	Associate degree	Yes, computer assistant
Liz	Female	18-30	White	n/a	Yes, art assistant, preschool teacher
Ellen	Female	51-65+	White	n/a	No
Gina	Female	51-65+	White	Associate degree	Yes, the technology lab

Data Collection

In the November 2021 PD session, paraprofessionals brainstormed and outlined their first STEM/CT integrated lesson using a STEM tool to teach during the K-6 STEM specialty class. This lesson plan used one of the MSD Lending Lab STEM tools to integrate CT into the lesson alongside the STEM tool. These lesson plans were uploaded to the Elementary PLC Canvas course in January. The evaluator was given access to the lesson plans of those paraprofessionals who consented to participate in the evaluation. In March 2022, paraprofessionals submitted their final lesson plans.

At the end of April 2022, the post-survey was sent out to the participants who filled out the pre-survey. Two reminder emails were sent. Additionally, all participants who participated in the pre-survey were invited to participate in a reflective interview about their experiences in phase 2 of the STEM/CT Professional Development Program. I conducted a semi-structured reflective interview via Zoom with eight participants that were recorded and transcribed.

Data Sources

Pre-post Surveys

I developed a 28-item pre-survey and 31-item post-survey, CT Professional Development Program (see Appendix C and Appendix D, respectively) from three existing surveys: the 2020 Gallup and Code with Google report (2020) on "Current Perspectives and Continuing Challenges in Computer Science Education in U.S. K-12 Schools," the Teacher Beliefs about Coding and Computational Thinking (TBaCCT) survey (P.J. Rich et al., 2020), and Teacher Computational Thinking Attitude questionnaire (Yadav et al., 2014). My pre-survey examines four constructs (see Table 9): computational thinking ability (4 items), computational thinking understanding (4 items), teaching CS/CT (7 items), and attitudes, values, and beliefs about CS/CT (12 items). The computational thinking ability was divided into two subconstructs with two items each: using a computer and problem solving. The post-survey adds a fifth construct, PD experience (3 items). Table 10 presents the evidence of reliability for the three surveys used to construct the survey used in this evaluation. The survey was written with a 6-level Likert scale, with response options in the following range:

(1) Strongly Disagree

(2) Disagree

(3) Somewhat Disagree

(4) Somewhat Agree

(5) Agree

(6) Strongly Agree

A section requesting demographic information (age, gender, race, and education) was included in the pre-survey.

Table 9

Pre-Post-Survey Constructs

Item #	Question	Construct	Source
8	When I am presented with a problem, I have difficulty breaking it down into smaller steps.	Computational Thinking Ability	TBaCCT (Rich et al., 2020)
26	I struggle to generalize solutions that can be applied to many different problems.	Computational Thinking Ability	TBaCCT (Rich et al., 2020)
13	I am NOT good at solving puzzles.	Computational Thinking Ability	TBaCCT (Rich et al., 2020)
8	I struggle to identify where and how to use variables in the solution of a problem.	Computational Thinking Ability	TBaCCT (Rich et al., 2020)

Table 9 (continued)

Item #	Question	Construct	Source
4	Computational thinking is understanding how computers work.	Computational Thinking Understanding: Using a Computer	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
2	Computational thinking involves thinking logically to solve problems.	Computational Thinking Understanding: Problem Solving	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
5	Computational thinking involves using computers to solve problems.	Computational Thinking Understanding: Using a Computer	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
14	Computational thinking involves abstracting general principles and applying them to other situations.	Computational Thinking Understanding: Problem Solving	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
25	I can explain basic computational thinking concepts to children (e.g., algorithms, loops, conditionals, functions, variables, debugging, pattern-finding).	Teaching CS/CT	TBaCCT (Rich et al., 2020)
18	I can recognize and appreciate computational thinking in all subject areas.	Teaching CS/CT	TBaCCT (Rich et al., 2020)
24	I can create computational thinking activities at the appropriate level for my students.	Teaching CS/CT	TBaCCT (Rich et al., 2020)

Table 9 (continued)

Item #	Question	Construct	Source
17	I can explain computational thinking well enough to be effective in teaching computational thinking.	Teaching CS/CT	TBaCCT (Rich et al., 2020)
11	I can explain how computational thinking concepts are connected to daily life.	Teaching CS/CT	TBaCCT (Rich et al., 2020)
1	I can develop and plan effective computational thinking lessons.	Teaching CS/CT	TBaCCT (Rich et al., 2020)
22	Computational thinking can be incorporated in the classroom by using computers in the lesson plan.	Teaching CS/CT	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
22	Computational thinking can be incorporated in the classroom by using computers in the lesson plan.	Teaching CS/CT	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
6	Computational thinking can be incorporated into the classroom by allowing students to problem-solve.	Teaching CS/CT	Teacher Computational Thinking Attitude Questionnaire (Yadav et al., 2014)
23	Computational thinking should be taught in elementary school.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)
9	Learning about computational thinking can help elementary students become more engaged in school.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)

Table 9 (continued)

Item #	Question	Construct	Source
16	Computational thinking content and principles CAN be understood by elementary school children.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)
20	My current teaching situation does NOT lend itself to teaching computational thinking concepts to my students.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)
3	Computational thinking is an important 21st-century literacy.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)
12	Computational thinking is an important part of today's science standards.	Value/beliefs about CS/CT	TBaCCT (Rich et al., 2020)
21	Offering opportunities to learn computational thinking is more important to a student's future success than other required courses like math, science, social studies/history, and English.	Value/beliefs about CS/CT	Gallup/Code with Google (2020)
15	Offering opportunities to learn computational thinking is just as important to a student's future success as other required courses like math, science, social studies/history, and English.	Value/beliefs about CS/CT	Gallup/Code with Google (2020)

Table 9 (continued)

Item #	Question	Construct	Source
7	It is important for my students to learn about computational thinking.	Value/beliefs about CS/CT	Gallup/Code with Google (2020)
19	It is important for me to learn about computational thinking.	Value/beliefs about CS/CT	Gallup/Code with Google (2020)
27	I am interested in learning computational thinking skills.	Value/beliefs about CS/CT	Gallup/Code with Google (2020)
29 Post	I had a positive experience in the Elementary STEM PLC Summer sessions.	PD Experience	
30 Post	I had a positive experience in the monthly Elementary STEM PLC sessions.	PD Experience	
31 Post	I was given ample opportunity to practice the computational thinking skills I was asked to learn and teach in the Elementary STEM PLC.	PD Experience	

Table 10

Evidence of Reliability for Survey Instruments

Instrument	Evidence of Reliability
Google/Gallup. (2020). Current perspectives and continuing challenges in computer science education in U.S. K-12 schools 2020 report	No reliability information was reported.

Table 10 (continued)

Instrument	Evidence of Reliability
Rich et al. (2020). Teacher beliefs about coding and computational thinking (TBaCCT) survey	Cronbach's alpha computed for each construct: *Values Beliefs about Computing* - pre-test=0.837 - post-test=0.817 *Coding* - pre-test=0.918 - post-test=0.915 *Computational Thinking* - pre-test=0.669 - post-test=0.707 *Teach* - pre-test=0.943 - post-test=0.942
Yadav et al. (2014). Teacher Computational Thinking Attitude questionnaire	Cronbach's alpha=0.76

Reflective Interviews

A semi-structured reflective interview protocol was developed to help probe participants for their understanding of computational thinking, their experiences with computational thinking, and their ability to design and teach STEM/CT integrated lessons (see Appendix E). These questions were written to gain insight into survey responses and address the three evaluation questions. Included in the interview questions were probes about a STEM/CT integrated lesson plan that each paraprofessional was asked to write. Paraprofessionals were compensated through the grant for their time creating their lesson plans. Each paraprofessional chose a grade level (K-6), a STEM tool from the MSD Lending Lab (see Table 7), and a CT component (Abstraction, Algorithms, Debugging, Decomposition, Patterns) to develop a STEM/CT integrated lesson plan. The participants' lesson plans will be discussed in the reflective interview.

Data Analysis

Survey Data Analysis

The pre-survey had 27 items and measured four constructs with the addition of three items and a fifth construct in the post-survey: computational thinking ability, computational thinking understanding (two subconstructs), teaching CS/CT, values/beliefs about CS/CT, and PD experience. After running descriptive statistics and checking the data for errors, I cleaned and prepared the data for analysis by replacing participants' names with pseudonyms. I reverse code items 8, 13, 20, 26, and 28. I then calculated a composite mean for each of the five constructs. I did this by adding the total rating (score) for each item associated with the construct and then dividing it by the number of items associated with it. I did this for each construct in the pre- and post-surveys. I then calculated item and participant means. Finally, I conducted descriptive statistics, where I computed frequencies of responses for each of the five constructs, looking at mean and standard deviation, and created bar graphs to visualize and summarize the tables.

Reflective Interview Data Analysis

Before beginning data analysis, I replaced participant names with pseudonyms and developed profiles for each participant, including age, education, experience, and general perceptions of CT. Otter.ai was then used to generate an initial transcript for each participant. I edited and refined each transcript for accuracy in MAXQDA. This process allowed me to read and re-read the interviews several times. In addition, taking the time to transcribe each interview helped me to begin recognizing potential codes and patterns among participants.

Due to the range of topics and responses collected in the reflective interviews, I used an eclectic coding (Saldaña, 2016) process. Data analysis of the reflective interview took place over four general phases. I describe these phases below.

Phase 1: Familiarization and Inductive Coding

During my first coding round, I used the evaluation questions and logic model to help identify and familiarize myself with the relevant segments from each interview and defined vague language within each one. For example, one participant says, “That’s really important to me.” “That” is vague language within the sentence. Therefore, I defined in brackets what “that” was referring to. After defining vague language, I began coding each interview using various coding methods, including Initial Coding, In Vivo Coding, and Emotion Coding (see Table 11 for an example of each method, Saldaña, 2016).

Phase 2: Deductive Coding

In this phase, I went back through the interview data and looked for mention of computational thinking (CT), algorithms, and the four CT components (practices) from CT4EDU.

Phase 3: Grouping & Pattern Finding

In this subsequent coding phase, I began to look for patterns within my codes and began to group, cluster, and combine codes using MAXQDA’s MAXMaps. Several maps were created as I refined my codes and clusters.

Phase 4: Generating and Refining Themes

In my last coding phase, I began theme naming and refining my codes and themes. For example, the theme “PD was a Great Experience” came from several codes, including: “helpful,” “memorable,” “enjoyed,” and “fun.” The generated themes were refined as I went through each interview, applying each theme to the various codes and groups of codes. In all, eleven themes were identified and will be presented throughout Chapter Four (See Table 12).

Table 11

Qualitative Coding Methods

Coding Method	**Code**	**Example**
In Vivo	“a great experience”	“But yeah, the process has been great…just the whole process is really smooth, and I feel so blessed…. So anyway, it’s been a great experience.”
	“intimidating”	“It’s so funny because it’s like it’s such a big word, and I think it’s really intimidating….”
Emotion	Discomfort	“For me, to start with, it was just being in a room with people I didn’t know.”
Initial Coding	Student Learning	“And can they do it? Are you kidding me? They can figure out anything you give them.”

Table 12

Qualitative Themes from Reflective Interviews

Theme	Definition
PD was "A Great Experience"	Examples of how participants perceived their experience (overall positive) in the STEM/CT Professional Development Program
Teaching CT was Overwhelming, Frustrating, and Hard	Participants shared examples of how hard teaching CT was and when they felt overwhelmed or frustrated teaching CT.
Need More CT Instruction & Practice	Examples of participants expressing the need for more CT instruction and practice
Uncertainty about Computational Thinking	Examples of how participants defined CT, often being unsure of what CT was
Computational Thinking as a Problem-Solving Process	Examples of participants describing CT as a problem-solving process
Computational Thinking as a New Vocabulary	Examples of participants sharing how CT provided them with a new vocabulary to talk about skills and processes being used
Computational Thinking is Learning to Use a Computer	Examples of when participants shared their view of CT as learning to use a computer
Support from Administrators	Examples of participants sharing experiences of when they were or were not supported in their role as the STEM/CT Specialty Teacher
Time, Money, & Materials	Examples from participants describing the need for more time, money, and/or materials
More Teaching Resources	Examples from participants describing the need for more teaching resources to support them in teaching STEM/CT integrated lessons
Additional Time to Collaborate with Other Paraprofessionals	Examples from participants describing the desire for and benefits of additional time to collaborate with other paraprofessionals

Analytical Strategy

The three evaluation questions guided this analysis. Table 13 provides the evaluation questions, the level of Guskey's model addressed, primary data sources used to answer them, and the form of data analysis.

Table 13

Evaluation Questions, Short-term Program Outcomes, Guskey Evaluation Level, Data Sources, and Data Analysis

Evaluation Question	Short-Term Program Outcome	Guskey (2000, 2002) Evaluation Level	Primary Data Source(s)	Data Analysis
Question #1 What were the participants' experiences in the STEM/CT Professional Development Program?	*Paraprofessionals will have a positive learning experience during the STEM/CT Professional Development Program.*	Level 1: Participants' reactions	Post Survey	Descriptive statistics
			Reflective Interview	Inductive coding
Question #2 How does participating in the STEM/CT program affect participants' understanding of computational thinking?	*Paraprofessionals will develop an understanding of computational thinking.*	Level 2: Participants' learning	Pre-Survey	Descriptive statistics
		Level 4: Participants' New Knowledge and Skills	Reflective Interviews	Inductive coding and deductive coding using CT components
			Post Survey	Descriptive statistics

Table 13 (Continued)

Evaluation Question	Short-Term Program Outcome	Guskey (2000, 2002) Evaluation Level	Primary Data Source(s)	Data Analysis
Question #3 What additional activities, resources, strategies, supports, and training do paraprofessionals still need to design and teach STEM/CT integrated lesson plans?	*Paraprofessionals will facilitate K-6 STEM/CT integrated lessons using computational thinking components.*	Level 4: Paraprofessionals' New Knowledge and Skills	Reflective Interview Post Survey	Inductive coding and deductive coding using CT components Descriptive statistics

Evaluation Question 1

I used data from three survey questions (#29, 30, and 31) and the reflective interview (questions # 13, 14, 15, and 16) with participants to answer evaluation question 1. The survey data gave me an overall view of their experience in the STEM/CT Professional Development Program. Their interview responses were coded inductively and analyzed to identify themes and commonalities among participants regarding their experiences. The interviews provided additional insights into their individual experiences in the STEM/CT Professional Development Program.

Evaluation Question 2

I used pre/post-survey data and analyses of responses to reflective interview questions (#2, 3, 4, 5, 6, 7) to answer the second evaluation question. Summary descriptive statistics from the pre-survey showed participants' current understandings of computational thinking. I then compared those descriptive statistics to the post-survey descriptive statistics to identify response trends about their understanding of computational thinking. Additionally, I compared pre- and post-survey scores using data visualizations and plots to determine if there was an improvement or growth.

I also used participants' reflective interview responses to help triangulate survey results. I deductively coded their answers about computational thinking, using the definition of computational thinking and CT components as the codes. The coding and analysis of the interview data helped generate themes describing participants' understanding of computational thinking, their attitudes and beliefs about computational thinking, and how they have evolved.

Evaluation Question 3

I used interview response data (questions #8, 9, 10, 11, and 12) to help answer this evaluation question. Interview responses were coded inductively, informed what elements of the professional development supported participants in designing STEM/CT-integrated lesson plans, and helped identify the areas that need improvement and more support.

Ethical Considerations

All USU IRB protocols were followed. In addition, several steps were taken to ensure the participants' privacy. Each participant signed a consent form before data collection began. All data was stored in an online Box folder with 2-factor authentication. Pseudonyms were used for each participant and elementary school they were assigned to.

Validity & Trustworthiness

Threats to Validity

There are several potential threats to internal and external validity with this evaluation. These threats include subjectivity/objectivity, transferability, social desirability issues, and instrumentation and measures issues. Although there are several potential threats, several strategies were implemented to mitigate them.

The first potential threat was that of subjectivity and objectivity. I was the sole coder, evaluator, and primary source for the data collected from the participants. Researcher involvement in data collection and analysis is inherently one of the challenges of qualitative research. I took several steps to help remedy this threat to validity. First, I was actively aware of my positionality and potential bias as a researcher/evaluator. Next, I attempted to confirm any findings or conclusions by triangulating the data between analyses of survey results and interview data.

A second potential threat to validity is the transferability of findings. The STEM/CT Professional Development Program is unique to the school district. Therefore, this context may not map well to other programs. However, I enhanced the transferability of any findings or conclusions by providing rich details of the context, program, assumptions, participants, etc.

Additionally, the threat of social desirability is always present. For example, participants may want to appear that they understand and can teach computational thinking when they may struggle because they want to appear that they can do their job well and are not struggling.

Lastly, the measures and instrumentation used in the evaluation could also potentially threaten validity. The survey I developed has never been used before, so there is no evidence of its reliability or validity. I ran a Cronbach's alpha to measure the reliability of the survey. The post-survey consisted of 31 items, and Cronbach's alpha value for the survey was $\alpha = 0.82$. Additionally, participants will already be familiar with the items that may impact the results because the same survey was used before and after the STEM/CT Professional Development Program.

CHAPTER IV

FINDINGS

Introduction

The purpose of this chapter is to present my findings from the evaluation of the Marshall School District STEM/CT Professional Development Program conducted during phase two (year two) of the implementation. The findings are organized by the evaluation questions. For evaluation question two regarding participants' understanding of computational thinking, I also draw upon the literature to examine various CT components learned and taught by participants.

Findings: Evaluation Question 1

What were the participants' experiences in the STEM/CT Professional Development Program?

Overall, the eight participants in my sample reported having a positive learning experience participating in the STEM/CT Professional Development Program (see Table 14). When asked on the post-survey if they had a positive learning experience in the summer PD sessions (post-survey item 29) and the monthly PD session (post-survey item 30), the responses were all "agree" or "strongly agree." However, when asked to respond to the statement, "I was given ample opportunity to practice the computational thinking skills I was asked to learn and teach in the Elementary STEM PLC," one participant reported that they disagreed (see item 31 Appendix F).

Table 14

Professional Development Experience Post Survey Responses (N=8)

Item #	Question	Mean	SD	Min	Max
29	I had a positive experience in the Elementary STEM PLC Summer sessions.	5.88	.33	5	6
30	I had a positive experience in the monthly Elementary STEM PLC sessions.	5.75	.43	5	6
31	I was given ample opportunity to practice the computational thinking skills I was asked to learn and teach in the Elementary STEM PLC.	4.75	1.20	2	6

Note: The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Inductive Qualitative Analysis of Reflective Interviews

In order to understand my participants' experiences further, I analyzed their reflective interview responses about their experience in the STEM/CT Professional Development Program using inductive qualitative analysis. Three main themes emerged that help support the survey findings above. These themes are: 1) PD was a great experience, 2) teaching CT was overwhelming, frustrating, and hard, and 3) participants need more CT instruction and practice. These three themes are explored in-depth below.

Theme 1: PD was A Great Experience

The first theme emerging from reflective interviews was that professional development was "a great experience" (Ana, Reflective Interview, April 2022) for the participants. Three aspects made the PD a "great experience" for participants: (1) the opportunity to participate in the PD and learn the content around STEM and computational thinking, (2) convenient access to STEM tools needed to implement STEM/CT integrated lessons with their students, and (3) the opportunity for participants to build relationships and receive support during the monthly PD sessions. Below, I further unpack each aspect that made the STEM/CT Professional Development Program a “great experience” for participants.

First, the opportunity to participate in the PD and learn the content around STEM and computational thinking contributed to the participants' great experience. Ana said, “I’ve written thank yous, to just thank her [MSD STEM & CS consultant] for teaching, you know, the things that I learned, I feel so grateful to have learned, you know, so anyway, it’s been a great experience” (Ana, Reflective Interview, April 2022). For many participants, the concept of computational thinking was entirely new, as was integrating CT into STEM. Ana had no prior experience with computer science or computational thinking, but she enjoyed learning all the STEM tools and teaching her students STEM/CT using the STEM tools. Moreover, she felt grateful she could learn the tools and feel comfortable with them so she could use them with her students.

Similarly, Ellen shared that “I’m really enjoying being able to learn the new resources” (e.g., various STEM tools, code.org, Skill Struck) and “It’s been really good personal growth and professional growth” because before the PD, she “really had no idea, the process of coding or the fact that things had to go in a certain order to make it work” (Ellen, Reflective Interview, April 2022). Additionally, learning how to write lesson plans made her “think a little differently” and “definitely made things easier” (Ellen, Reflective Interview, April 2022). For Ellen, the STEM/CT Professional Development Program was crucial for her in developing the knowledge and skills she needed to teach STEM/CT integrated lessons.

Like Ana and Ellen, Marissa also enjoyed the PD because of the learning opportunity. She said, "I really enjoyed it [PD] because I, you know, I didn't really have any previous experience, so everything I just kind of like soaked up and was like everything yeah, I didn't really have any negative effects with it at all. I was grateful to just learn all everything" (Marissa, Reflective Interview, April 2022). Unlike classroom teachers, paraprofessionals are often not afforded the same opportunities for professional development, so participants felt the opportunity to learn about STEM tools and computational thinking as part of the PD was part of what made it a "great experience."

The PD was a great experience for participants because of the ease of access to STEM tools needed to implement STEM/CT integrated lessons with their students. For instance, Hailey said:

> Oh, I love learning all the new tools, the way they provide everything we need, and they ship it to our school, we ship it to each other, they've had great support. If anything is broken or missing, we can ask for help. It's been just perfect, and it's been great. I don't have any complaints in it. (Hailey, Reflective Interview, April 2022)

In this quotation, Hailey emphasizes the value of having STEM tools easily accessible in her school as a part of the program because she is given very little prep time to prepare and gather materials herself. Ana also commented on the tools aspect of the program:

> I just, it [PD] has been amazing. I guess I was just like, how is this all gonna work? They're gonna be sending tools to us each week, and then we have to send them back, we have to do this and are like, I just didn't know how all this was going to really work. And it was, and it has just been so easy, you know, the process of it. (Ana, Reflective Interview, April 2022)

While Ana was initially unsure about the logistics of sharing STEM tools across schools, she ultimately described the process as “easy.” The ease of access to the STEM tools helped to contribute to the STEM/CT Professional Development Program being a great experience for participants because it did not add additional tasks to their plate. Some participants had very little prep time, so having the STEM tools organized and delivered to their school made implementing STEM/CT-integrated lessons possible.

A third aspect of participants having a “great experience” with the STEM/CT Professional Development program was its opportunity to build relationships and receive support. When referring to the monthly PD meetings, Sophia said, “I have loved the meetings, and I haven’t had any problems with them. Without them, I think I would have been a little lost. And I really look forward to them every month because I am getting new ideas” (Sophia, Reflective Interview, April 2022). Attending the STEM/CT Professional Development Program each month provided Sophia and other participants opportunities to gain ideas to take back and implement with their students in their classrooms.

Like Sophia, sharing resources and ideas was also an essential element of the PD experience for Charlotte and her success in the STEM/CT Professional Development Program. She described in her interview how it “felt impossible to teach without the anchor of the PLC with the connections that [she] made with the other group members to be like ‘Hey, did you have this already? What did you do in your classroom?’…I almost want to feel emotional about how much this PLC was the anchor to my success in this classroom.” (Charlotte, Reflective Interview, April 2022). Charlotte credits her experience in the STEM/CT Professional Development Program as the reason she could teach STEM/CT integrated lessons and feel successful in the classroom.

Overall, the participants appeared pleased with their experience in the STEM/CT Professional Development Program because it gave them opportunities to learn and share ideas about the STEM/CT content, build relationships, and receive support. The ease of access to STEM tools needed to implement STEM/CT integrated lessons with their students also contributed to their positive experience. If the participants had not had a positive experience attending the STEM/CT Professional Development Program, they may not have continued to attend and, therefore, not have learned the skills needed to teach STEM/CT integrated lessons. Additionally, if the coordination of the STEM tools was too complicated, it is doubtful that the STEM/CT integrated lessons using the STEM tools from the MSD Lending Library would have been taught. Because the STEM/CT Professional Development Program overall provided a positive experience for participants, access to computing education was possible for students.

Theme 2: Learning & Teaching CT was Overwhelming, Frustrating, and Hard

The second theme emerging from the reflective interviews was that the participants found learning and teaching STEM/CT overwhelming, frustrating, and hard. Although participants reported positive experiences in the summer and monthly professional development sessions, this did not necessarily translate to their learning and teaching experiences. All but one participant shared that teaching STEM/CT was overwhelming, frustrating, and hard. Some participants' feelings originated from learning about and teaching STEM/CT integrated lessons. For others, these feelings were caused by the STEM tools they were trying to use and engage their students with. Lastly, for others, student behaviors created feelings of being overwhelmed and frustrated.

For Ana and Gina, feeling overwhelmed or frustrated appeared to originate with learning and teaching STEM/CT. Gina shared her feelings of just wanting to "walk out because we [she & her co-teaching partner, [Ana] felt stupid" and that she "honestly didn't believe that [she] could do it" because they were struggling to understand something. Gina shared:

> I think we have to be aware that it's hard…. I mean, there was a time I saw tears in [Ana]'s eyes. And there was another time where I felt the same way. And we both said to you [sic], it's okay, you can do this, we can figure this out…. Don't feel bad that you can't do this (Gina, Reflective Interview, April 2022).

These quotes demonstrate the feelings that Ana and Gina experienced as they learned and began to teach STEM/CT concepts. Charlotte expressed similar sentiments when she reflected, "I walked into this deer in headlights [sic]" and "I was so overwhelmed with the concept of like teaching this" (Charlotte, Reflective Interview, April 2022). The sentiments expressed by these three participants are unsurprising. The literature on CS PD tells us that even certified teachers feel overwhelmed with the amount of content they must learn to teach CT (Celephkolu, O'Halloran, & Boyer, 2020), so it is unsurprising that paraprofessionals with little to no formal teacher preparation felt overwhelmed by the CT content.

Other times, it was the STEM tools themselves that caused frustration. Gina remembered getting frustrated when she had issues with students' Chromebooks or got locked out of online resources she planned to use in her lessons. Marissa described not liking certain STEM tools because they were too hard for the students to use and caused lots of frustration. For example, she explained she did not like using coding software with her students because “it was hard for the students to use and it kind of made the kids frustrated, so that’s what we want to stay away from is…leaving the kids frustrated and maybe having bad feelings towards them” (Marissa, Reflective Interview, April 2022).

Charlotte also had frustrating moments with the STEM tools and technology. She struggled with the VR headsets. She said, “None of this [sic] VR glasses goggles are working, I don’t remember how to make them work, and like she [STEM/CS Consultant] just shows up like a magic magician and makes them work” (Charlotte, Reflective Interview, April 2022). From her experience, you can see how frustrated Charlotte felt that she could not get the VR headsets to work.

Liz also had a similar experience of frustration with STEM tools not working. She learned from experience to take the STEM tools home to play with them because “if you’re fumbling around with it up there, the kids aren’t listening to you because you don’t know what you’re talking about” (Liz, Reflective Interview, April 2022). From her experience, I understood how frustrating it could be trying to teach a new STEM tool when you were still trying to learn it for yourself.

For others, like Marissa and Ellen, student behaviors created the most frustration while teaching STEM/CT integrated lessons. Marissa shared that “a lot of the kids aren’t awesome; they don’t behave very awesome” (Marissa, Reflective Interview, April 2022), so she found the behavioral tips in the STEM/CT Professional Development sessions helpful. Not being able to teach STEM lessons because of student behavior was frustrating.

In her reflective interview, Ellen also shared her struggles with behavioral issues and described her experience with her upper-grade students who “just sit there and talk, no matter what we say” and that she has experienced “discipline problems and disrespect…and it’s just been [sighs], been a doozy” and that overall “it has been a rough year” (Ellen, Reflective Interview, April 2022). These behavioral struggles with her students contributed to Ellen feeling overwhelmed and frustrated as she did not know how to deal with the challenging behaviors being exhibited by students in her classroom.

Throughout the STEM/CT Professional Development Program, participants were challenged by learning and teaching the STEM/CT content, finding it overwhelming, frustrating, and hard. These feelings are realities of learning new content, not feeling like an expert, working with technologies that sometimes fail, and dealing with student behaviors in the classroom.

Theme 3: Need More CT Instruction & Practice

The third theme from the analysis of participants’ reflective interviews was the need for more CT instruction and practice. A minority of the participants felt that they needed more opportunities to practice the CT concepts presented in the PD program,

In her reflective interview, Sophia expressed a desire to discuss computational thinking at each monthly PD session and to get help connecting her current teaching activities to CT. She reflected:

> To make this really happen, I think…you should have something about that [CT] at every meeting, like 20-30 minutes of computational thinking at every meeting, because we’re all adult learners, so it takes us a long time to learn. If I had like 20 minutes on it to reinforce it and say hey, what did you do this week? Let me tell you how you could tie that back to computational thinking; as an adult learner, that would be so helpful to me (Sophia, Reflective Interview, April 2022).

In this reflection, Sophia indicated that she needed more reinforcement of the CT concepts at each PLC meeting and help to connect the CT concepts to what she was already teaching.

Liz also expressed the desire to have a “good vocabulary lesson” to support her learning of computational thinking and model how to teach the CT vocabulary to her students. Liz’s request for a “good vocabulary lesson” suggests she did not understand the CT content and would benefit from additional CT instruction and practice to better support her in teaching her students STEM/CT integrated lessons.

In summary, two participants requested additional reinforcement around the computational thinking concepts they were asked to learn and teach. This additional practice could be centered around the CT vocabulary and concepts and how to connect them to what paraprofessionals are currently teaching in the STEM specialty.

Findings: Evaluation Question 2

How does participating in the STEM/CT program affect participants’ understanding of computational thinking?

To answer this question, I computed the composite means for the four constructs: addressing their *understanding* of CT (two subconstructs), *ability* to do CT, *beliefs about teaching* CT, and *values and beliefs* about CT for both the pre-and-post surveys by adding their rating for each item and dividing it by the number of items per construct as well as means for individuals participants’ responses to each of the survey items. I also examined their responses to the select questions from the reflective interviews. Inductive and deductive qualitative analyses using the CT components of abstraction, algorithm, debugging, decomposition, and patterns were used to analyze the reflective interviews. I will present my survey findings and qualitative analysis findings to evaluation question two, organized by the following four survey constructs:

1) Computational Thinking Understanding
    a. Problem Solving
    b. Using a Computer
2) Computational Thinking Ability
3) Beliefs about Teaching Computational Thinking
4) Values and Beliefs about Computer Science and Computational Thinking

Construct 1: Computational Thinking Understanding

To better understand how participants understood CT, two subconstructs were developed. The first subconstruct, *Problem Solving* (item #2, item #14), looked at participants' understanding of CT as abstraction and thinking logically to solve problems. The second construct, *Using a Computer* (item #4, item #5), looked at participants understanding as CT involves using a computer to solve problems and understanding how a computer works. After professional development, it is expected to see the *Problem Solving* subconstruct increase while the second subconstruct, *Using a Computer*, either declines or remains unchanged because while using a computer can help students engage in CT, knowing how/using a computer is *not* a requirement to be engaged in CT.

On average, all the participants reported beginning the STEM/CT Professional Development Program with an understanding of computational thinking (composite means exceed 5.00 on a 6-point scale; see Table 15 and Table 16). In the pre-survey, most participants (all but one) agreed that CT is understanding how computers work and that CT involves using computers to solve problems (see Figure 3, Figure 5). All participants self-reported that they agreed or strongly agreed that they understood computational thinking involved thinking logically to solve a problem and that CT involves abstraction (see Figure 4, Figure 5). Seven of the eight participants started the STEM/CT Professional Development Program with a mean of four (somewhat agree) or higher in both subconstructs (see Figure 3, Figure 4, Table 16).

Table 15

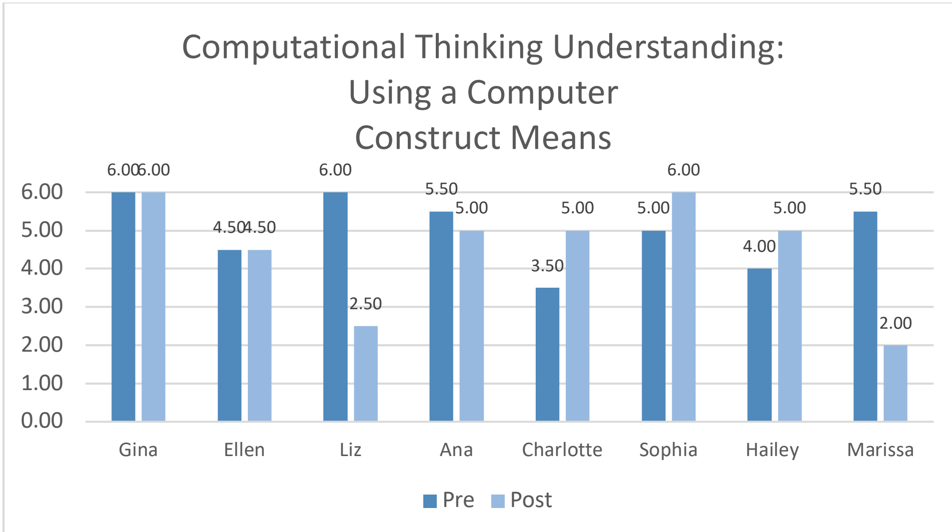
Composite Means for Four Constructs (and two subconstructs) from Responses to Pre- and Post-surveys (N=8)

Construct	**Pre**	**Post**
Computational Thinking Understanding: Using a Computer	5.00 *(SD 0.87)*	4.50 *(SD 1.39)*
Computational Thinking Understanding: Problem Solving	5.19 *(SD 0.24)*	5.50 *(SD 0.66)*
Computational Thinking Ability	3.94 *(SD 0.73)*	4.56 *(SD 0.62)*
Teaching Computer Science/Computational Thinking	4.44 *(SD 0.66)*	5.27 *(SD 0.45)*
Values/Beliefs about Computer Science/Computational Thinking	4.78 *(SD 0.22)*	5.29 *(SD 0.37)*

Note: The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Figure 3

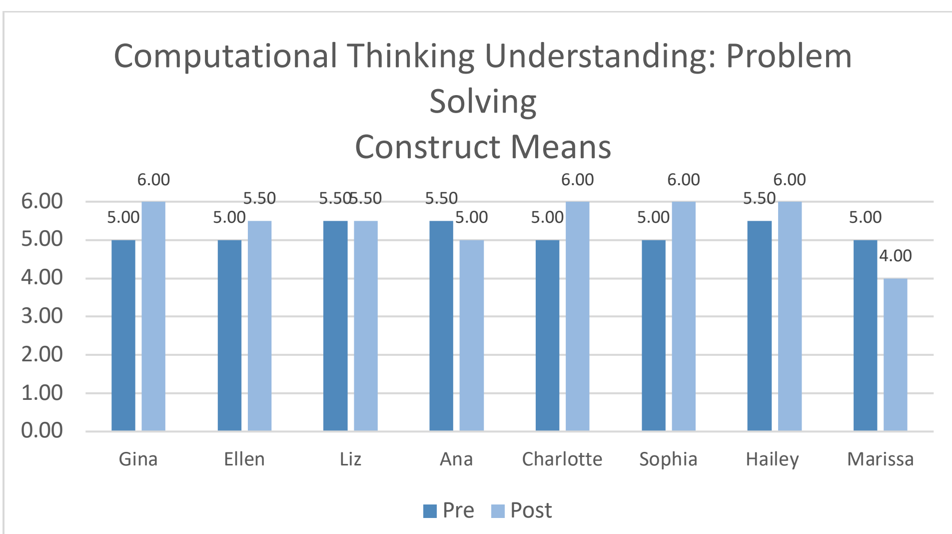
Composite Means for CT Understanding: Using a Computer from Two Items (4, 5) by Participant from Responses to Pre- and Post-surveys (N=8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Figure 4

Composite Means for CT Understanding: Problem Solving from Two Items (2, 14) by Participant from Responses to Pre- and Post-surveys (N=8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

The survey results (see Table 16) reveal that, on average, participants agreed that CT involves thinking logically to solve problems (Item # 2). However, some participants may have entered the STEM/CT Professional Development Program with potential misunderstandings about what computational thinking is. For example, on average (see Table 16, item 4, and Figure 3), participants entered the program believing that computational thinking is understanding how computers work (Item # 4) and that computational thinking involves using computers to solve problems (Item #5).

Between the pre- and post-survey, there was minimal change in the two subconstructs (see Table 15). This result was not surprising given that they entered the STEM/CT Professional Development Program with a high level of self-reported CT understanding (see Table 15, Table 16). In the post-survey, three participants reported growth (indicated by a lower average from pre- to post) in understanding the relationship between computers and computational thinking, and their self-reported scores suggest that some misunderstandings about computers and computational thinking were resolved (see Figure 3, Table 16). Additionally, both item 4 and item 5 saw a decrease in the reported score, which also indicates there may be a better understanding of the relationship between CT and computers as well. These small changes may indicate that their understanding of computational thinking has deepened, and they now understand that while computers ***may*** be used to solve problems, there are instructional programs that do not use a computer where students can engage in computational thinking and develop computational thinking skills without using a digital device (Caeli & Yadav, 2020).

In addition, five of the eight participants' responses from pre- to post-survey had a self-reported minimally higher average when reporting that their understanding of CT involves thinking logically to solve a problem and that CT involves abstraction (see Table 16, Figure 4). Overall, there was minimal change in most participants' self-reported understanding of computational thinking on both subconstructs between the pre- and post-surveys (see Table 15).

Table 16

Computational Thinking Understanding Pre/Post Survey Responses. Shaded cells indicate post-survey responses (N=8)

Item	Question	Mean	Min	Max	SD
	Computational Thinking Understanding: Using a Computer				
4	Computational thinking is understanding how computers work.	5.00	3	6	1.00
		4.38	2	6	1.49
5	Computational thinking involves using computers to solve problems.	5.00	4	6	0.87
		4.63	2	6	1.41
	Computational Thinking Understanding: Problem Solving				
2	Computational thinking involves thinking logically to solve problems.	5.38	5	6	0.48
		5.25	4	6	0.70
14	Computational thinking involves abstracting general principles and applying them to other situations.	5.00	5	5	0.00
		5.38	4	6	0.86

Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

A qualitative analysis was conducted on the following interview questions related to understanding computational thinking:

Interview Question 3:

What is computational thinking? How would you explain what computational thinking is to a colleague who has never heard of it?

Interview Question 4:

Can you give me an example of what computational thinking would look like in a classroom? How would you know your students were engaging in computational thinking?

Interview Question 5:

Has your understanding of CT changed as a result of participating in this project? Why or why not?

The analysis revealed four themes about how participants expressed their understanding of computational thinking: 1) uncertainty about computational thinking, 2) computational thinking as a problem-solving process, 3) computational thinking as a new vocabulary, and 4) computational thinking is learning how to use a computer.

Theme 1: Uncertainty about Computational Thinking

Despite reporting high levels of understanding of CT, for most participants, the interviews revealed that the concept of CT was new to them at the beginning of the STEM/CT Professional Development Program. As reported in their interviews, six of the eight participants had no prior experience with CS or CT before the STEM/CT Professional Development Program. After a year of exposure, some participants reported still being uncertain and not confident in their understanding of CT. Liz expressed that her understanding remained unchanged after participating in the STEM/CT Professional Development Program, as evidenced in the interview excerpt below.

Liz was asked, "Has your understanding of computational thinking changed as a result of participating in the elementary STEM PLC?" and she responded, "I would say no, just because I didn't really know much of it prior to this…I would say it didn't really change, and I feel like I am still learning about it" (Liz, Reflective Interview, April 2022). I then clarified with Liz, saying, "So you don't feel like you've hit that point where you can say confidently, like 'I know what computational thinking is, and I can do it.' You're still feeling like 'I need to learn more about this?' Liz responded, "Yeah, most definitely." (Liz, Reflective Interview, April 2022). Liz felt that her understanding of CT "didn't really change" over the year in the STEM/CT Professional Development Program and that she was "still learning about it" (Liz, Reflective Interview, April 2022).

This response was interesting because Liz answered "agree" or "strongly agree" on the pre-survey to every item measuring CT understanding. Liz's interview responses may indicate that CT was not a priority of the PD sessions and that she was not provided meaningful opportunities to engage with CT concepts.

When asked, "What is computational thinking?" in her reflective interview, Ellen responded, "Honestly, I can't answer that." However, Ellen added, "I would think more of thinking things through in a process" (Ellen, Reflective Interview, April 2022). Ellen's response demonstrates that she has a basic, simplified understanding of computational thinking.

When responding to the same question, Sophia said:

> …as little as I know about it [CT], and I don't know much about it [CT], I don't even want to say that I know anything about that. [I] only heard this term from you and a couple of other people in the last couple of months, to be honest with you. (Sophia, Reflective Interview, April 2022)

As demonstrated above, it seems that some participants were still very unsure of what CT is and did not feel comfortable being able to define CT when they were asked to do so directly. In Ana's reflective interview, she struggled to verbalize her understanding of CT. As she tried to answer the question, Ana took several long pauses and expressed how she should have reviewed CT before the interview. She described CT as "computational thinking is where we try to... [long pause, thinking] compute... figure out what, how the outcome might be ahead of time or sorting that out? Shoot" (Ana, Reflective Interview, April 2022). She then attempted another try at answering the interview question and explained CT as:

> Uh, computational thinking is where you [long pause, thinking] break down the parts of, of the project, I guess. [pause, thinking] I don't know. Maybe I needed like two seconds to just refresh what those are. And then I would remember better this morning, shoot. (Ana, Reflective Interview, April 2022)

Ana's hesitancy and uncertainty reflect her inability to verbalize her understanding of CT. This quotation suggests that Ana still does not have a solid understanding of computational thinking.

In the reflective interviews, many participants expressed uncertainty and a lack of confidence in their understanding of computational thinking. These commonalities across participants' responses suggest that participants felt they were not provided significant opportunities to engage in learning experiences with computational thinking through the STEM/CT Professional Development Program.

Theme 2: Computational Thinking as a Problem-Solving Process

When asked to define computational thinking, other participants communicated their understanding of it as a process. For example, in a later portion of her reflective interview, after discussing computational thinking further with the researcher, Ellen described computational thinking as "thinking things through in a process" and "the fact that things ha[ve] to go in a certain order to make it work" (Ellen, Reflective Interview, April 2022). Ellen reported an increase from "agree" to "strongly agree" when responding to the prompt, "Computational thinking involves thinking logically to solve problems" (Item #2). However, this contradicts what she said at the beginning of the interview when she said, "Honestly, I can't really answer that" when asked what CT is.

Liz described her understanding of CT, emphasizing identifying the problem that needed to be solved. She said:

> From what I understand, it is a lot about, kind of the way that they can handle problem-solving skills and also how they can logically kind of pinpoint certain things that need to be addressed. But also, a good amount of it is also just learning to be okay with like problem solving and learning that it's okay to like make mistakes and stuff like that. (Liz, Reflective Interview, April 2022).

Her response demonstrates an understanding that computational thinking is a problem-solving process. Liz's response is interesting because she reported a decrease from "strongly agree" to "agree" on the post-test when responding to the same prompt above (Item #2), although, in her interview, she expressed her understanding of CT with confidence.

Hailey started the STEM/CT Professional Development Program with high confidence in her understanding of CT. She responded, "strongly agree," when responding to the prompt, “Computational thinking involves thinking logically to solve problems” (Item #2). In her interview, Hailey also highlighted identifying the problem or issue that needs to be addressed in her response. When asked to describe CT, she said, “I think it's just working through a problem, taking steps figuring out what the problem is, taking steps to solve the problem” (Hailey, Reflective Interview, April 2022).

Sophia emphasized in her reflective interview that when teaching her students, they can use CT outside of just playing computer games and that CT is problem solving. She said, “They [her students] can use computational thinking to solve problems outside of just, you know, playing a game on the computer. You know it’s really problem solving…” (Sophia, Reflective Interview, April 2022). Sophia recognizes that CT has more applications beyond just “playing a game on the computer” and is trying to approach CT as a problem-solving process with her students.

Gina also conceptualized computational thinking as a process emphasizing the STEM element. Gina described CT as “just to try[ing] to get them to think about all the different processes that, you know, go on in, that involve STEM, whether it's science, or whether it's technology or whether it's engineering, you know?” (Gina, Reflective Interview, April 2022). In this quotation, Gina explained that problem-solving is used in almost all STEM subject areas and how CT can help students identify those processes. As another example, Sophia described CT as “…the step-by-step processes to make the computer do what you want to do” (Sophia, Reflective Interview, April 2022). Marissa also expressed a similar sentiment by describing CT as “computation[al] thinking is being able to problem-solve, figure out certain patterns, and being able to…working with computers and things like that on different programs” (Marissa, Reflective Interview, April 2022). The responses from Sophia and Marissa demonstrate their understanding of CT as a process used, as well as connecting the process to using a computer.

As the interview excerpts above demonstrated, the participants generally appeared to understand that computational thinking is a problem-solving process. However, some participants, like Marissa and Sophia, appear to have connected their understanding of CT to knowing how to use a computer, while others have potentially not yet conceptualized how problem-solving is used to engage students in developing solutions that can be represented as algorithms or as a series of steps. It is essential to recognize that there are varying definitions of CT within the field and that more in-depth probing is needed to understand how participants fully conceptualize CT.

Theme 3: Computational Thinking as a New Vocabul*ary*

When asked to define computational thinking, one participant expressed her understanding of computational thinking as a new vocabulary she was teaching her students. Gina emphasized that computational thinking is:

> …something that you're trying to implement, you know, different vocabulary and trying to break things down to introduce students [to] some of our main vocabulary words that we've been trying to introduce this year, like algorithms and abstractions, and decomposition, those kinds of words, debugging and pattern. (Gina, Reflective Interview, April 2022)

From my interpretation, Gina explained that she already viewed her students as engaging in CT through problem-solving. She was using computational thinking as a way to name the skills for her students and provide them with a new vocabulary in which to talk about them. Gina shared an example of how learning the correct vocabulary impacted her understanding of CT. She explained:

> Sometimes, a vocabulary word doesn't make sense to me, but when you break it down, like, you know, like, um, decomposition, it meant something different to me. Because all I could think it was compost pile [laughing], but then I thought it's still the same because you're breaking down the, you know, vegetables and the different things to enhance the soil. And so, in a way, I mean, it is the same thing. I just had to, like, switch my thinking a little bit to think, 'What does that have to do with STEM' other than, you know, science? (Gina, Reflective Interview, 2022)

Learning the vocabulary around CT allowed Gina to identify and connect to her understanding of CT.

Sophia also saw herself engaging in computational thinking without a vocabulary to describe it. She explained that in her job, she "did quite a bit of computational thinking, even though I wouldn't have labeled it that, right? I would never have labeled it that" (Sophia, Reflective Interview, April 2022). Learning about computational thinking has given her a new vocabulary to talk about how she understands and uses computational thinking. It has also changed how she thinks about tasks in her daily life. She shared a short anecdote demonstrating this change:

> Now I realize that you can use computational thinking in day-to-day things and make even your own life efficient. I am notorious for running around like a chicken with my head cut off. To do ten things at one time. … And it's been very helpful, even for me, to look back at what I am telling my students, who, you know, sometimes their mind races, and bringing them back to what the problem is. To coming back and saying, well, here's your problem [Sophia], you're trying to do 14 things at once, and you're only capable of doing one. So, let's write that down, and let's just write that down and see what you want to do. Don't run up and down the stairs 1000 times. You could probably run downstairs two times if you just think it through. So, I think the label has made me think. (Sophia, Reflective Interview, April 2022)

Sophia's response demonstrates how she conceptualizes CT as a problem-solving process that can be applied to everyday tasks and how she sees herself engaging with CT.

There is nuance in how participants use and interpret the term "computational thinking." For example, as Sophia described above, she would not have labeled her work "computational thinking." Conflicting literature exists on defining the terms and concepts that constitute CT, and this is reflected in how participants talk about CT.

Theme 4: Computational Thinking is Learning to Use a Computer

Sophia and Marissa connected their understanding of computational thinking with learning to use a computer. Marissa connected her understanding of CT to using/learning to use a computer when defining computational thinking. She expressed, "For me, I think computation [sic] thinking is being able to…. I guess, like, you know, working with computers and things like that on different programs" (Marissa, Reflective Interview, April 2022). From her response, working with a computer seems to be a key component of her understanding of computational thinking.

Sophia reflected on her understanding of computational thinking by saying, "it's learning not how to think like a computer and tell me if I'm wrong, but for me, it's more learning how to use a computer." (Sophia, Reflective Interview, April 2022). Sophia elaborated by saying:

> You are the ones that [are] telling the computer what to do so you can; you have to think about the step-by-step processes to make the computer do what you want to do. Whether it's a little handheld game or even for a kindergartner making the monkey eat one banana, four bananas, or five bananas, they still have to think about that. So, we do little games for the younger students, like I want you to tell me how I can get to my desk in the back corner of the room. So, they'll say, okay, take ten steps. So, I'll take ten baby steps, or take giant steps, so they start to think you know more along the lines, like wait, you have to be precise, to come to a solution to the problem with just getting me to go to my desk in the back of the room. (Sophia, Reflective Interview, April 2022)

As evidenced by this excerpt, Sophia appeared to have a beginning understanding of what computational thinking is and why it is used. Her response demonstrates that she incorporates CT concepts (such as algorithms) into her teaching. Still, she appears to hold incorrect assumptions. For example, she frequently conflates CT with learning to use a computer. However, CT involves more than just learning how to use a computer; her example demonstrates engaging her students in “unplugged” CT. Although she describes it as learning to use a computer, students can use the CT process to solve a problem without using technology or a computer.

In addition to the survey data analysis and inductive qualitative analysis of reflective interviews, I also conducted a deductive qualitative analysis using the computational thinking components to understand participants’ understanding of computational thinking to analyze the reflective interview responses from participants using the following computational thinking components as the codes.

Computational Thinking Components

According to the MSD STEM and Computer Science Specialist (K. Taylor, Personal Communication, 2022), five computational thinking components were taught during the STEM/CT Professional Development Program. The components include abstraction, algorithm, decomposition, debugging, and patterns (see Table 17 for participants’ definitions from their reflective interviews). Four of these five components were taken from the CT4EDU project (Yadav et al., 2019). Each paraprofessional was given posters with a definition and example of each CT practice to hang in their classrooms and use in their lessons. Each paraprofessional chose one CT component to highlight within their STEM/CT integrated lesson plan, which was discussed in their interview. Table 17 shows some understandings of the various CT components participants mentioned in their interview after a year of participation in the STEM/CT Professional Development Program.

Table 17

CT Component Definitions from Participants' Reflective Interviews

CT Component	Participants' Definition(s)
Abstraction	"Taking those parts, taking them apart more." (Ana) "Figure out what's important and what's not" (Ellen) "Break free from the minutia" (Sophia) "Focus" (Hailey)
Algorithm Design	"Thinking ahead, several steps ahead to try to get to that result" (Sophia) "Speak robot language" (Gina)
Debugging	"You have to study, maybe something that's working is something that's not working to try to figure out why — what's different" (Gina)
Decomposition	"Taking a more complex problem and breaking it down" (Sophia)
Patterns	"A sequence" (Gina)

Note. Because the lesson plans referred to below are/will be available in a public repository, it would be easy to identify the participants based on the lesson plan descriptions. I referred to participants without using pseudonyms to maintain confidentiality within this section.

In the reflective interview, I asked each participant a series of questions that have been used to write the following sections. The interview prompts included the following:

1) Can you give me an example of what computational thinking would look like in a classroom? How would you know your students were engaging in computational thinking?
2) Can you tell me a story about working with one of the STEM tools and technologies (e.g., Ozobot, Makey Makey, etc.)?
3) Tell me about a STEM/CT integrated lesson plan you designed. What was the implementation of that lesson plan like?

Abstraction.

A participant described how she used the ClassVR headsets to help 1st-grade students engage with CT and the practice of abstraction. The VR headsets were used to make observations about animals and the various habitats they live in. As various habitats were shown, the participant said she helped guide and direct students on various details important to the habitat, such as shelter, food, weather, physical characteristics, etc. She said this lesson helped her students engage in the practice of abstraction. She said,

> I'm thinking with first graders they can get easily distracted or get off topic. Or, if we're watching the VR, you know, they might focus on other things than what the topic is. So, abstraction, I chose it because we just want to focus. (Participant, Reflective Interview, April 2022).

Based on her response, I can conclude that the participant is beginning to develop an emerging understanding of abstraction because she is trying to help her students concentrate on the important details of each habitat and ignore or filter out the details that are not pertinent to each habitat.

Algorithm Design.

Another participant described how she used the Micro:bit and Zoobs to engage 1st graders in algorithm design. She described the implementation in-depth during her reflective interview. The lesson began with a discussion of the beginning, middle, and end, and they talked about how to make a peanut butter and jelly sandwich. The participant then showed her students MakeCode and the start, LED, and ICON blocks. As a class, they programmed the words “hi” and “bye” on the Micro:bit. Each student was then given the opportunity to push the start button on the Micro:bit to watch the program run. While she went around the room to individually teach each student how to engage with the Micro:bit, the other students were working in pairs to spell their names using the Zoobs. Then, they mixed up the order of their names and had their partner put their name back in the correct order. She ended the lesson with a discussion of the “funny order” of names mixed up and how their names do not make sense mixed up. She then reviewed the three coding blocks used with the Micro:bit. She ended her lesson by emphasizing how, just like it is essential for the letters of our names to be in order, the code blocks in MakeCode must be in order for the Micro:bit program they created to work correctly. When asked in her reflective interview how this lesson helped her 1st graders engage in algorithm design, the participant responded:

> I think, just like showing the kids like beginning, middle, and end and that there was an order to, an algorithm, that you can’t mix up the order or it won’t create what you’re trying to create. Like it has to be in the correct order of first things, middle things, and things. (Participant, Reflective Interview, April 2022)

Additionally, in the interview, the participant described how they were able to discuss how “algorithms are ordered” and that when you get a pile of letters (student names mixed up), “what do you do to make sense of them? How do you put them in order to make sense of the letters?” She then explained how this connected to algorithm design and the Micro:bits by saying, “If you don’t have the start button connected with the other building blocks, it won’t have its beginning, and like if you don’t end it, its’ just going to keep like going” (Participant, Reflective Interview, April 2022).

From the interview, I can identify that the participant understands how to model algorithm design in a developmentally appropriate way for young children. She has the 1st graders focus on putting things in order and a correct sequence to create a beginning, middle, and end.

Debugging.

Another participant described how she used the Sphero Bolt to engage students in debugging practice. She designed mazes for her 4th-grade students and provided them with specific tasks she wanted them to complete throughout the maze (e.g., turn, spin, flashlights, etc.). When I asked how this helped her students engage in debugging, she responded, “Because if it doesn’t work….or if it turns the wrong way and it doesn’t get up, then they have to go back through and figure out what went wrong and how to fix it.” (Participant, Reflective Interview, April 2022). Based on her interview response, it is apparent that she has a basic understanding of debugging (finding and fixing errors or mistakes) and is attempting to encourage her students to engage in debugging as they complete the maze. However, she has not created a specific plan to engage her students more deliberately in debugging, such as providing code for the Sphero Bolt that contains errors for the students to troubleshoot.

Decomposition.

One participant described how she used the STEM tool Dash to engage her 4th-grade students in computational thinking and decomposition. In the lesson, students worked in small groups to create a path to move through a maze provided to them. Students used rulers to measure how far the Dash traveled in centimeters in one "move forward" command. Then, they created a map or blueprint for how their Dash robot would travel through the maze (e.g., move 60 cm, make a 90-degree turn, move 30 cm, make cow noise). The participant supported her students in breaking the maze down into sections by requiring an animal sound to be made at each checkpoint in the maze. She encouraged students to focus on getting to one checkpoint at a time rather than the entire maze. She said this helped her students engage in decomposition. She said:

> I think I understood decomposition as taking a more complex problem and breaking it down to where you could go step-by-step, right? So, going along this grid, you would have to break down how many, how you get along that grid, right, no matter what the challenge was. (Participant, Reflective Interview, April 2022)

Based upon her interview response, this participant appears to have a basic understanding of decomposition and described an activity that included scaffolds to support her students in breaking down a larger problem (maze) into smaller chunks (checkpoints throughout the maze).

Patterns.

Lastly, a participant who was a former choreographer described how she used the Sphero Bolt to engage students in computational thinking and pattern recognition. She realized how "choreography is coding" (Participant, Reflective Interview, April 2022). She decided to have students "choreograph" or "code" the Sphero Bolt. She had her students begin by learning a short line dance in class. Then, the students had to program their robots to do the same “line dance” or movement pattern using block coding on the SpheroEDU site. She created visuals to match the Sphero Bolt to use on the whiteboard with her students. She reviewed the various commands students needed to create the line dance, such as roll, spin, turn, etc. She also introduced the repeat command so students could repeat the line dance pattern. She then used the large versions of the coding blocks she had made on her whiteboard to help students build the pattern on SpheroEDU. She clearly demonstrated that she understood and could engage her students in recognizing, building, and repeating patterns.

Based on the analyses of the interviews, there is currently a wide range of understanding of computational thinking and CT practices among participants. As evidenced by the analysis above, there is a range of understanding across the various CT components, from one participant only demonstrating an emerging understanding of abstraction to another demonstrating a clear understanding of how to engage students in pattern recognition.

Construct 2: Computational Thinking Ability

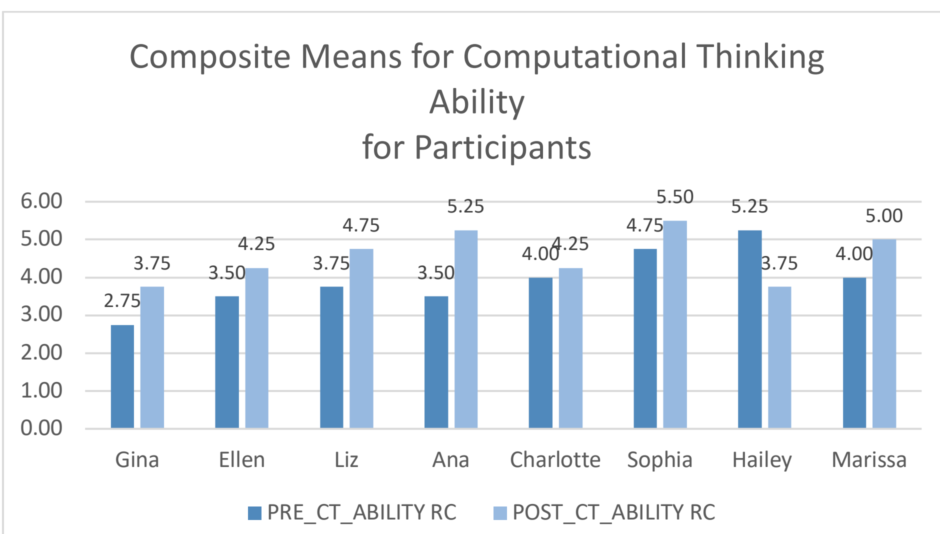
Regarding participants' computational thinking ability, there was a small increase in the composite mean (see Table 15). When looking at survey results, seven of the eight participants’ confidence in their CT ability increased from the pre-to-post survey (see Figure 5, Table 18). Ana, Liz, and Marissa demonstrated the most gains in their CT abilities, while Hailey reported a decrease in her CT abilities from pre- to post-survey (see Figure 5, Table 18).

Individual item responses provide information about the areas where they were not confident in their abilities (see Table 18). For example, a few participants, like Charlotte and Sophia (see Item 8 Appendix F), did not report changes between pre- and post-survey responses in their ability to break a problem down into smaller steps. However, Marissa reported a significant growth in her ability to break down a problem into smaller steps (see Figure 6).

Six of the eight participants self-reported that they struggled less with their ability to identify where and how to use variables in the solution of a problem at the end of the PD program than before (see Figure 7). These findings may indicate that the STEM/CT Professional Development Program may have helped to improve participants' CT abilities as there was limited growth in participant composite means, and only three participants had a self-reported composite mean of 5 (agree) or higher.

Figure 5

Composite Means for CT Ability from Four Items (8, 26, 13, 28) for Participants from Responses to Pre- and Post-surveys (N=8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Table 18

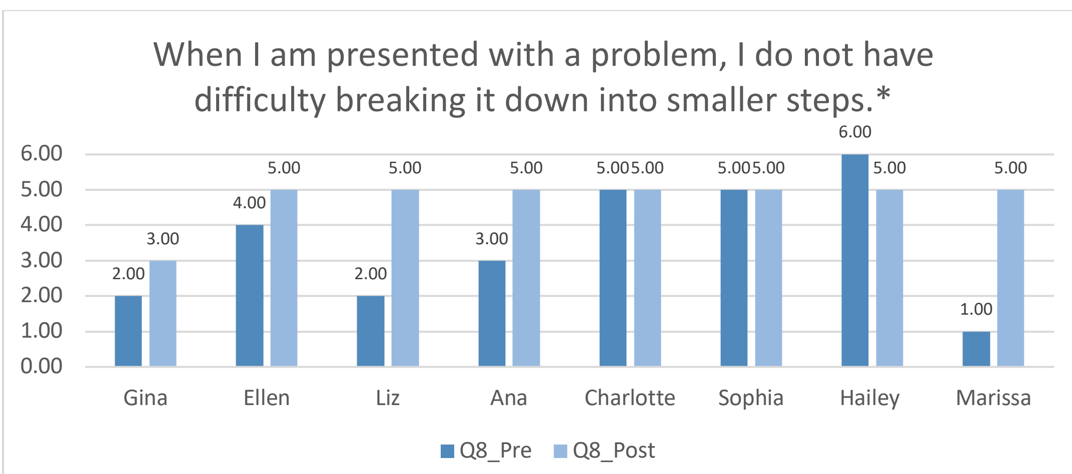
Computational Thinking Ability Pre/Post Survey Responses. Shaded row indicates post-survey responses (N=8). RC indicates the item was reverse-coded.

Item	Question	Mean	Min	Max	SD
8 RC	When I am presented with a problem, I have difficulty breaking it down into smaller steps.	3.50	1	6	1.66
		4.75	3	5	0.66
26 RC	I struggle to generalize solutions that can be applied to many different problems	4.00	2	5	1.12
		4.38	3	6	1.11
13 RC	I am not good at solving puzzles.	4.75	4	5	0.46
		4.50	1	6	1.50
28 RC	I struggle to identify where and how to use variables in the solution of a problem.	3.50	2	5	1.32
		2.38	1	3	0.70

Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Figure 6

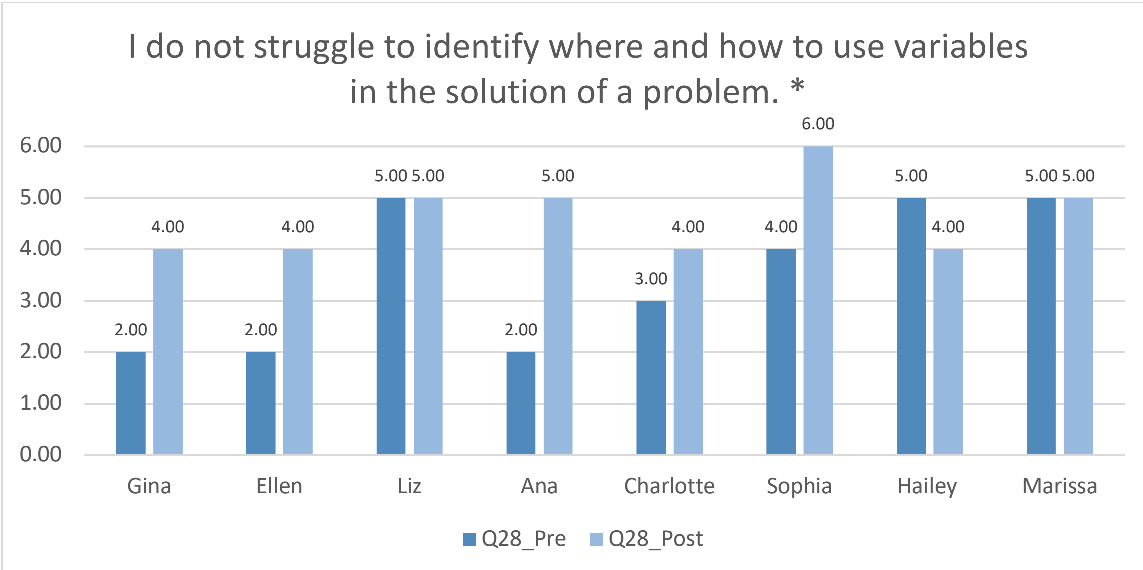
Participants Responses to Item #8 (N= 8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree. * *The item has been re-written to reflect the reverse coding.*

Figure 7

Participants Responses to Item #28 (N =8)



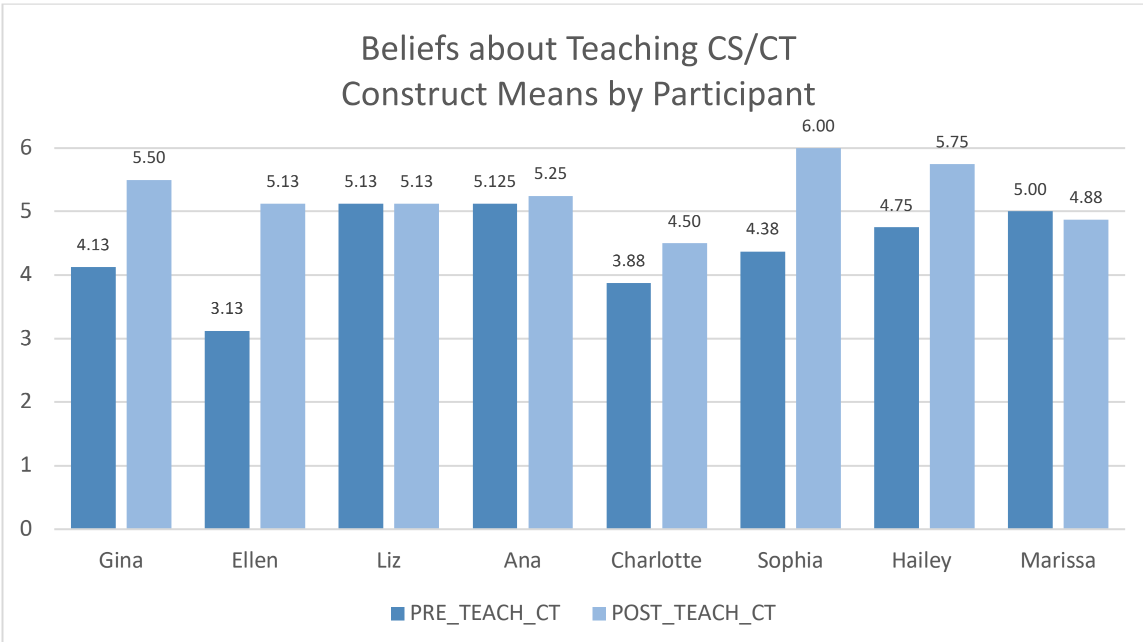
Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree. * *The item has been rewritten to reflect the reverse coding.*

Construct 3: Beliefs about Teaching Computer Science/Computational Thinking

Overall, participants reported beliefs about teaching CS/CT were positively impacted by their participation in the STEM/CT Professional Development Program (see Table 15 shows the composite mean for this construct, Table 19 shows item means, and Figure 8 shows the composite means for participants for this construct). Only one participant reported a decrease in their average belief about teaching CS/CT (see Figure 8).

Figure 8

Composite Means for Beliefs Teaching CS/CT from Seven Items (25,18, 24, 17, 11, 1, 22, 6) for Participants from Responses to Pre- and Post-surveys (N=8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

All participants reported on the post-survey that they at least "somewhat agree" that they can explain basic computational thinking concepts to children, and five of the eight participants self-reported growth from pre- to post-test in their ability to do so (see Item 25 in Appendix F). Additionally, all but one participant (Ana) self-reported gains in their abilities to create computational thinking activities at the appropriate level for their students (see item 24 in Appendix F). However, only half (4 of 8) of the participants self-reported growth in their ability to develop and plan effective CT lesson plans (see item 1 in Appendix F). Three participants (Liz, Anna, and Charlotte) reported less ability to develop and plan effective CT lessons after the year-long professional development compared to the beginning of the PD program (see item 1 in Appendix F). The reflective interviews also expressed the desire for more support and resources for STEM/CT-integrated lessons (see Evaluation Question 3, Theme 3).

Table 19

Beliefs about Teaching CS/CT Pre/Post Survey Responses. Shaded row indicates post-survey responses shaded (N=8)

Item	Question	Mean	Min	Max	SD
25	I can explain basic computational thinking concepts to children (e.g., algorithms, loops, conditionals, functions, variables, debugging, pattern-finding).	3.63	2	5	1.41
		5.13	4	6	0.60
18	I can recognize and appreciate computational thinking in all subject areas.	4.25	2	5	0.97
		5.38	4	6	0.70
24	I can create computational thinking activities at the appropriate level for my students.	4.00	2	6	1.41
		5.25	4	6	0.66
17	I can explain computational thinking well enough to be effective in teaching computational thinking.	4.38	3	5	0.70
		5.38	4	6	0.86
11	I can explain how computational thinking concepts are connected to daily life.	4.13	2	5	1.27
		5.25	5	6	0.43
1	I can develop and plan effective computational thinking lessons.	4.88	4	6	0.78
		5.25	4	6	0.66
6	Computational thinking can be incorporated into the classroom by allowing students to problem-solve.	5.38	5	6	0.48
		5.88	5	6	0.33
22	Computational thinking can be incorporated in the classroom by using computers in the lesson plan.	4.88	4	6	0.60
		4.75	2	6	0.99

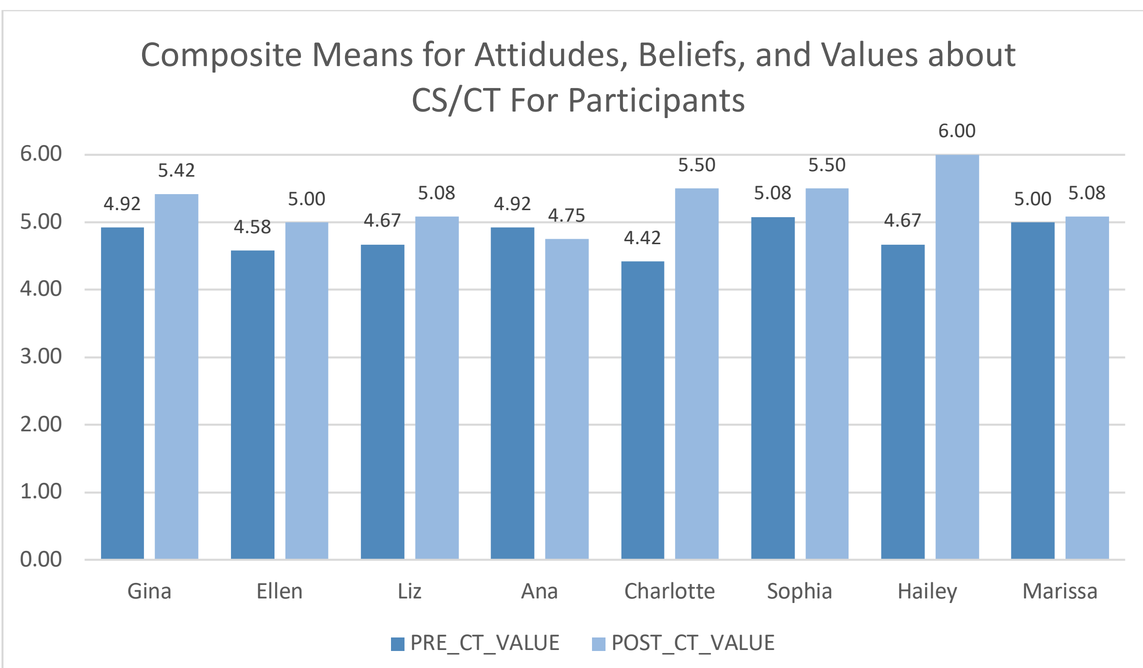
Note. The scale ranged from 1 = strongly disagree to 6 =strongly agree.

Construct 4: Values and Beliefs about Computer Science and Computational Thinking

Overall, the STEM/CT Professional Development Program participants reported a change in their reported attitudes, beliefs, and values about computer science and computational thinking as measured by 12 items on the pre/post survey (see Table 15, Figure 9, Table 20). All but one participant's composite mean for this construct increased from pre- to post-test (see Figure 9). Given that most participants started the STEM/CT Professional Development Program reporting the attitude and beliefs that CS/CT was important, there was little room for growth in this construct.

Figure 9

Composite Means for Attitudes, Beliefs, and Values about CS/CT from 12 items (23, 9, 16, 20, 3, 12, 21, 15, 10, 7, 19, 27) for Participants from Responses to Pre-and Post-survey (N= 8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

However, given the limited ability to demonstrate growth, it appears that participating in the STEM/CT Professional Development Program helped some participants see the benefit of CT for their students' learning. For example, the mean for item 9 increased from 4.38 to 5.38 (see Table 20), indicating that the STEM/CT PD helped participants learn that CT can help students become more engaged in school. Additionally, all participants reported on item 16 (see Table 20, Appendix F) of the post-survey that they agree or strongly agree that elementary students can learn CT concepts. Lastly, the mean for item 23 increased on the post-survey, why they were agreeing to strongly agreeing (see Table 20, Item 23 in Appendix F) that CT should be taught in the elementary school setting. Although participants began the STEM/CT Professional Development Program with high self-reported attitudes, beliefs, and values about CS/CT, there was an indication, evidenced by increased composite mean for this construct, that they experienced growth from the beginning to the end of the program.

Table 20

Attitudes, Beliefs, and Values about CS/CT Pre/Post Survey Responses. Shaded Boxes Indicate Post-Survey (N=8)

Item	Question	Mean	Min	Max	SD
23	Computational thinking should be taught in elementary school.	5.00	4	6	0.50
		5.63	5	6	0.48
9	Learning about computational thinking can help elementary students become more engaged in school.	4.38	3	5	0.70
		5.38	3	6	0.70
16	Computational thinking content and principles CAN be understood by elementary school children.	4.88	4	6	0.60
		5.75	5	6	0.43

Table 20 (continued)

Item	Question	Mean	Min	Max	SD
20	My current teaching situation does NOT lend itself to teaching computational thinking concepts to my students.	2.75	1	6	1.09
		1.88	1	2	0.33
20 RC	My current teaching situation lends itself to teaching computational thinking concepts to my students.	4.25	3	6	1.16
		5.13	5	6	0.35
3	Computational thinking is an important 21st-century literacy.	5.38	5	6	0.48
		5.63	4	6	0.70
12	Computational thinking is an important part of today's science standards.	5.13	4	6	0.60
		5.88	5	6	0.33
21	Offering opportunities to learn computational thinking is more important to a student's future success than other required courses like math, science, social studies/history, and English.	4.00	3	5	0.50
		4.63	3	6	0.99
15	Offering opportunities to learn computational thinking is just as important to a student's future success as other required courses like math, science, social studies/history, and English.	5.13	4	6	0.60
		5.75	5	6	0.43
10	Offering opportunities to learn computational thinking is less important to a student's future success than other required courses like math, science, social studies/history, and English.	2.50	2	4	0.71
		2.50	1	6	1.41
7	It is important for my students to learn about computational thinking.	5.38	5	6	0.48
		6.00	6	6	0.00
19	It is important for me to learn about computational thinking.	5.25	5	6	0.43
		5.75	5	6	0.43
27	I am interested in learning computational thinking skills.	5.50	5	6	0.50
		5.50	5	6	0.50

Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

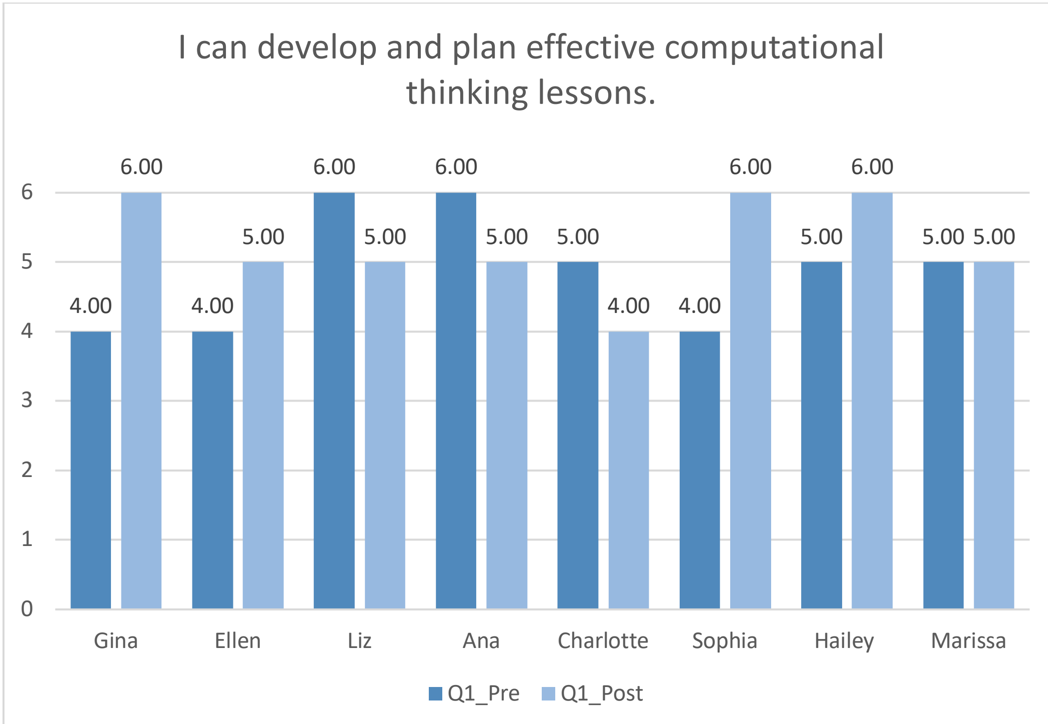
Findings: Evaluation Question 3

What additional activities, resources, strategies, supports, and training do participants still need to design and teach STEM/CT integrated lessons?

In order to answer this question, I looked at participants' survey responses (item 1) and the reflective interviews. Participants were confident in the pre-survey when asked to respond to whether they could develop and plan effective CT lessons. All participants agreed or strongly agreed (see Figure 10). However, in the post-survey, only four of the participant's ratings increased, three participants' confidence decreased, and one remained the same (see Figure 10). I examined the reflective interviews to understand participants' experiences with STEM/CT integrated lesson plans.

Figure 10

Participants Responses to Pre- and Post-Survey Item # 1 (N=8)



Note. The scale ranged from 1 = strongly disagree to 6 = strongly agree.

Inductive Qualitative Analysis of Reflective Interviews

To better understand what additional activities, resources, strategies, supports, and training paraprofessionals may need to teach STEM/CT integrated lessons, I analyzed the reflective interview responses to questions 8, 9, 10, 11, and 12, which asked about participants' experiences with STEM/CT integrated lesson and lesson planning (see Appendix E) using inductive qualitative analysis. Four themes were identified. These were 1) support from administrators, 2) time, money, and materials, 3) more teaching resources, and 4) additional time to collaborate with other paraprofessionals.

Theme 1: Support from Administrators

In the analysis of reflective interviews, a common theme that emerged in participants' responses was support from administrators. However, how they talked about their support varied. Each paraprofessional works at a different elementary school, and the level of support they receive from the administration varies. Some participants indicated they received great support, while others felt abandoned and unsupported. For example, Gina felt highly supported by the administration in her school building in two ways. First, there is a school-wide reward system for specialty classes where students/classes earn points during each specialty. There is a school-wide "rewards week" where students are rewarded for their good behavior and get different activities based on the number of points they earn. These rewards help her feel supported by the administration regarding classroom management because behavior expectations have been set in place.

Additionally, her specialty class could occasionally get loud. A few teachers have complained about the noise level in her classroom, but Gina felt supported in what she was doing. In her interview, Gina said,

> I think you have got to have the backing from your administrators. Because it's [STEM lab] become the best place to be in the school… Because if we didn't have backing by the administrators, we wouldn't be making noise in the STEM lab. You, they're the ones that tell the teachers, that's what they're doing. (Gina, Reflective Interview, April 2022).

Based on her experience shared above, Gina felt supported by her school administration in having a loud STEM lab even though it appears that other teachers in the school did not like the noise generated from the STEM lab.

Sophia and Liz also felt highly supported by their school's administration. In her interview, Sophia said, "I have a fantastic principal who basically, you know, supports whatever I say" (Sophia, Reflective Interview, April 2022). Liz, who had been struggling with behavior issues in her specialty classes, explained that the issues she was having with students were "being helped and addressed by my principal and assistant principal and a few of the other behavioral instructors" (Liz, Reflective Interview, April 2022). The behavioral support that Liz received from her school administrators helped her to feel supported as a paraprofessional. Administration involvement helps participants feel supported in their jobs as STEM/CT specialty teachers.

However, other participants did not feel supported at all. For example, Hailey expressed that her biggest problem is not getting much support from her school. She provides several examples, such as asking for tape and being told she cannot have any, making a big deal about getting paper for her classroom, not having anyone come to check in and see the activities she is doing in the classroom, and feeling like the teachers are not interested or supporting what she teaches in the classroom. She said:

> …but I just don't feel that support with the school. Like they don't even seem interested in what kind of lessons I'm doing or anything. Like, well, I could just be showing movies every day, and they would never even know. (Hailey, Reflective Interview, April 2022)

The lack of support from her school often makes her "feel like a babysitter" (Hailey, Reflective Interview, April 2022).

Also, Hailey's administration does not support her attending the monthly PD meetings. Hailey explained in her interview that her school has her teaching on Fridays, so she does not even finish teaching before the PD meeting begins. Then, after school, she still has to clean up and put away her entire classroom (desks, chairs, etc.) because she shares a room with the drama teacher. She had missed most of the PD by the time she had cleaned up her classroom. These experiences shared by Hailey illustrate why she did not feel supported as the STEM/CT specialty teacher in her school.

Marissa also began the school year missing out on the monthly PD sessions. She was unaware that they were happening because it was never communicated to her by the school administrators. She said, “I was kind of like just thrown in, so I think I would like more direction….” (Marissa, Reflective Interview, April 2022). Charlotte shared a similar sentiment of feeling like she was just “thrown in.” She shared, “…it was overwhelming, it was still frustrating, I didn’t feel like I had any support from my administration” (Charlotte, Reflective Interview, April 2022). She joked that if it were not for the guidance she received at the monthly PD, she was “gonna go work at Starbucks, hey, I can make coffee” (Charlotte, Reflective Interview, April 2022). From the interviews, it was clear that the level of support provided to the participants was extremely important, with participants feeling valued and supported in teaching the STEM/CT specialty class.

Theme 2: Time, Money, & Materials

Additionally, nearly every participant interview discussed the lack of time, money, and/or materials. Many paraprofessionals do not receive paid prep time; if they do, there is still not enough time to prepare lessons and activities for seven grade levels each week. Gina can only work 17 hours per week, so she spends much of her time at home planning for the school day. In her interview, she said:

> So, we maybe have an hour a week that we’re paid for prep time. I would, it would be nice if we were paid for more prep time because it takes more time than

> an hour for that many classes. You know, we have six to seven classes every day…and so that would be helpful. (Gina, Reflective Interview, April 2022)

From her experience, Gina needs more than one hour per week to prepare for six to seven classes daily, so she does not spend personal time at home preparing. Additional prep time would benefit Gina in planning and preparing to teach the STEM/CT specialty class. Like Gina, Sophia also does much prep time and learning at home. Although Sophia is given two hours of preparation time per week, it is not enough. She will often go into the school on the days she does not teach to prepare and prepare for the days she is teaching. Similarly to Gina and Sophia, Hailey is left with the choice of being unprepared or planning and preparing on her own time at home. In her interview, Hailey explained:

> I don't get very much preparation time to do it. Like this year, I get more time because I go every other week, so I get a week off to think about my plan. But next year, I'm going to do it every week, and pretty much the only time they give me is the time that I'm here at school, so it doesn't really give me any extra time to plan and prepare. So, I just have to kind of, I either I want to do it at home, on my own time, or just come up with something quickly you know. I mean, I just have to show up and do it…. I don't think they realize it takes a little bit of time to come up with this thoughtful lesson plan. (Hailey, Reflective Interview, April 2022)

Marissa expressed how stressed she felt not having any preparation time. She said, “It was kind of stressful because…when you have 1000 students, you have to plan in advance, like all supplies and everything, because if you don’t, you’re in trouble” (Marissa, Reflective Interview, April 2022). She described spending her own time learning things on YouTube after school that she can implement in her classroom. For example, when working with the Sphero Bolts, she went to YouTube to figure out how to build her own Sphero course in her classroom and stayed after school to build it for the next day.

On the other hand, Ellen receives an hour of preparation time each day. In her interview, she says, “I just couldn’t imagine not having any prep time.” Ellen uses her prep time to play and tinker with the STEM tool for the week, find resources to use in her lessons, make copies, and learn new skills to teach her students. Having this prep time has helped her feel prepared to teach her students. Based upon the experiences of each participant described above, it is evident that preparation time is extremely important to each participant, and the lack thereof has negatively affected many of them as they strive to teach the STEM/CT specialty class.

Next, a recurring theme in the interviews was money and materials. These issues included lacking pay, using personal funds and materials, and spending personal time finding additional materials to use in their classrooms. Most participants are asked to do a lot with little other than the STEM tools that come on rotation from the MSD STEM Lending Lab. In her reflective interview, Marissa described how she began the school year. “I had nothing. Like, I don't have, I just, all I had is a classroom and chairs and desks. Like, I didn't even have like a pencil or paper or nothing at all, so I started from like nothing” (Marissa, Reflective Interview, April 2022). It was overwhelming not knowing that she was going into the classroom with basically only desks and chairs and no other materials or supplies. This lack of resources was a point of conversation amongst paraprofessionals at the monthly PD sessions when they would share what resources were available (or not available) to them at their schools.

Participants like Liz and Marissa also found themselves attending additional training or reaching out to other organizations to help supply their classrooms with the materials they needed for lessons on weeks they did not have a STEM tool. Marissa described signing up with the STEM Action Center to do a presentation for her class because they had "kits they delivered to your school with all the materials and everything" (Marissa, Reflective Interview, April 2022). Liz attended additional science training through the school district to have extra lessons and materials to use on weeks without the STEM tools.

Gina and Anna team teach at the same school. They were given $500 for the entire school year. They both strive to be very frugal with the budget they have. They constantly ask for donations, go to the DI to find things for the classroom, and spend time writing grants to get money and supplies for their classroom. They find creative ways to use what is donated to them. For example, they were given a large bag of straws. So, during a week that they did not have STEM tools, they taught a unit on air and tried to see who could blow the paper off the straw the furthest. In her interview, Gina explained:

> We also use a lot of resources for [sic] my home because the school gave us a $500 budget for the whole year. And to buy the things that are thrown away like cups and spoons and forks, you know what I mean, the things that you're building with- tape and Play-Doh, those kinds of things, eat that up…. We're always bring things from home…. I'm always raiding my craft closet…and sometimes, I'll put a shout-out to my kids. (Gina, Reflective Interview, April 2022)

In her interview, Ellen shared that she uses her personal funds to purchase lessons and activities from Teachers Pay Teachers. She also pays for a monthly subscription to the STEM Teacher's Club, which hosts online professional development, lesson plans, STEM challenges, etc., that she uses to help her with lessons and activities in her classroom. Participants are doing their best with what they are given and try to make the money and materials stretch however they can, sometimes even using their personal funds and supplies as necessary.

Additionally, for all they are asked to do, they get paid very little for their time and effort. In her interview, Hailey noted:

> I could literally get paid more at McDonald's or Walmart, and that's really kind of discouraging…. I have a degree…. I could really go get a job in the industry, and I don't mean to say that this isn't a real job, but getting paid nothing…. I realize I'm kind of being in a charitable position. So, I might do it next year, but I really, it would really help if…I got paid a little bit better. (Hailey, Reflective Interview, April 2022)

Charlotte also expressed the frustration she has experienced the past year with her title and compensation. In her interview, she shared:

> I think the title of "classroom assistant" has been challenging because that's not what we do in this [STEM specialty] room. Like we are creating lesson plans, we are meeting state standards, we are teaching these vocabulary words, and like, and we're doing it for all developmental ages. So, I feel like what we do in this room is not necessarily assistant. We are the teacher, and like the process of what we do in this room is not, I don't know what the right word is, like, honored in our title or in our pay. (Charlotte, Reflective Interview, April 2022)

She continued to express how the district needs to realize that they are not classroom assistants, need more prep time, and are not like "say PE…I don't walk into the classroom and be like well, let's just grab something out of the closet, you know." These STEM specialty paraprofessionals are asked to do a classroom teacher's job without a classroom teacher's knowledge, training, experience, or pay. This obligation is evident to them and frustrating for many participants.

Another "time" theme across the reflective interviews was more time with the STEM tools. Paraprofessionals received all of their training on STEM tools before school started in August of 2021. However, some participants only received the STEM tools in February 2022. In her interview, Charlotte explained, "It was a little intimidating to be like, wait, I learned about this seven months ago, and I have to teach it on Monday." Although videos were available on Canvas, participants wanted more hands-on time with the tools. Marissa also requested more time with the STEM tools. She explained how often in the monthly PD sessions, "we just kind of talk about them…but we don't really use them" (Marissa, Reflective Interview, April 2022). She would like more hands-on experiences with the STEM tools because "when you actually use them in your classroom, it's way different than the PLC [monthly PD]."

Sophia also requested more time with the STEM tools to prepare and use them with her students in the classroom. She explained, "You get these tech tools literally two days before you have to introduce them to students you learned them six months ago" (Sophia, Reflective Interview, April 2022), so having additional time would be beneficial to preparing to teach. She also expressed how using the tools is fun. She said, "I wish you could have them for longer; you could do so much more with them" (Sophia, Reflective Interview, April 2022). They only get the STEM tools for one or two 40-minute class sessions and never see them again for the rest of the year. This made it difficult to teach students how to use the tool *and* be able to teach and engage students in computational thinking activities with a tool.

Theme 3: More Teaching Resources

Writing their lesson plans and planning activities for the weeks they did not have a STEM tool delivered to their school was difficult for most participants. In the interviews, several participants mentioned that having more lessons and teaching resources available would be very helpful, especially if they were "unplugged" lessons that teach CT skills. In her interview, Marissa explained how "all the STEM tools [were] planned out for the year, but then filling those lessons and my first year doing this, that was really hard for me to come up with lessons without STEM tools" (Marissa, Reflective Interview, April 2022). Liz shared in her interview that "We were having difficulties with finding lesson plans and stuff …it also felt like we were just kind of like thrown out on the job with like, not knowing kind of still not what to teach and stuff" (Liz, Reflective Interview, April 2022). Feeling unprepared was often because no lessons or resources were available outside the lessons for the STEM tools that came on rotation.

Charlotte also mentioned in her interview that she felt she had to choose between being prepared and not getting paid or being unprepared. If there were more lessons available for paraprofessionals to use during the weeks they did not have STEM tools, they would not have to make a choice to work for free, so they would be prepared to teach. In her interview, Sophia mentioned how she is "YouTube's best friend now" and that "all I do is watch YouTube" to get ideas of what she can do and teach her students in the classroom.

Theme 4: Additional Time to Collaborate with Other Paraprofessionals

Lastly, participants want and need additional time to collaborate; many spend their own time collaborating with their peers throughout the year. In her interview, Charlotte expressed how having 30 additional minutes dedicated to collaboration added to the monthly PD session would be beneficial. She mentioned how all the 3rd-grade teachers at her school get together to discuss the math program and share ideas of how they make it work for their students. She would like time to talk with the other STEM specialty instructional assistants to discuss what lessons have worked well, what they are doing in their classrooms, etc. She expressed how it would be nice "to be able to have a little bit more time to connect with other people teaching the same thing that we're teaching" (Charlotte, Reflective Interview, April 2022).

Hailey shared how the only support and encouragement she receives is from the other paraprofessionals in the STEM/CT Professional Development Program. In her interview, she describes how she has collaborated and received support and encouragement throughout the year. "We can share emails; I can go visit. I've been to their schools before to see what they're doing" (Hailey, Reflective Interview, April 2022). Liz also felt highly supported by those in the STEM/CT Professional Development Program. She shared in her interview how often, during break moments at the monthly PD, she "learned that [she] wasn't the only one that was struggling with students being absolutely crazy, so it helps [her] not feel so alone" (Liz, Reflective Interview, April 2022). During the break moments or staying a bit longer after the monthly PD session, Liz could talk with the other paraprofessionals and bounce around ideas, get tips on classroom management, and receive encouragement.

In her interview, Marissa described "parking lot meetings." After the monthly PD meeting, she and a few other paraprofessionals would meet in the parking lot to share what they were doing and get ideas from one another. Marissa also went to their schools to help her understand what she could do in her classroom. Sophia also expressed the desire to have more time to share ideas. She said, “If I had other people that have already done it [a lesson] if we can do a lot of that collaboration, I think it would inspire more ideas” (Sophia, Reflective Interview, April 2022). Gina also expressed the desire for more community in her interview. When asked what additional support or help she needed, she responded, “Having a group of people that are going through the same thing…and just share with each other...because there’s some things that go on within schools that get frustrating…so it’s just nice to have people that you can talk to that are doing the same things” (Gina, Reflective Interview, April 2022). There is a strong desire to share and learn from one another, and the participants would like the time to collaborate and do so.

Summary

In summary, participants reported having a positive experience participating in the STEM/CT Professional Development Program. Some also reported that teaching the STEM specialty could often feel overwhelming and hard. Findings suggested that after one year of participation in the STEM/CT Professional Development Program, they had varying levels of understanding of computational thinking, as evidenced in their survey and reflective interview responses. After participating in the STEM/CT Professional Development Program, some participants indicated they did not feel prepared to integrate CT, while others indicated they could do so successfully.

Findings also identified several additional activities, resources, strategies, support, and training paraprofessionals still need, including more support from administration, preparation time, money, materials to use within the classroom, more CT instruction and practice, more time to use the STEM tools, additional teaching resources, and additional time to collaborate.

CHAPTER V

DISCUSSION AND CONCLUSIONS

Overview

As more states emphasize CS education and adopt CS standards, there is an increasing need to provide equitable access to CS education (Gretter et al., 2019). This dissertation set out to evaluate one school district's efforts to provide equitable access to CS education through integrating computational thinking into a STEM specialty class for K-6 students taught by paraprofessionals. In this evaluation, I evaluated a STEM/CT Professional Development Program for paraprofessionals teaching this STEM specialty class. I aimed to answer three evaluation questions that addressed participants' experiences, understanding, and additional needs moving forward. I present a summary of findings for each of the evaluation questions followed by a conclusion of my evaluation findings, discussion, limitations, and finally, recommendations for practice and further research.

Summary of Findings and Discussion

Evaluation Question 1

Evaluation question one examined the experiences of paraprofessionals participating in the STEM/CT Professional Development Program. Using Guskey's Five Levels of Professional Development Evaluation Model (Guskey, 2002), I answered evaluation question one using Level One of Guskey's Model, which examines participants' reactions to the professional development program. This examination includes information such as their enjoyment, comfort in the space, and whether the PD was helpful. My findings suggest that participants had an overwhelmingly positive experience participating in the STEM/CT Professional Development Program. Many participants felt that the STEM/PD Professional Development Program was vital to their success in the classroom. Participants felt that the PD Program supported their learning and provided opportunities and resources to help them teach with the STEM tools provided in the MSD STEM Lending Lab. Participants also found support from each other in the PD Program. Many participants shared that they sometimes felt frustrated or overwhelmed with teaching STEM/CT integrated lessons using the STEM tools or the day-to-day struggles of working with students. The STEM/CT Professional Development Program monthly meetings greatly supported the participants. Recommended STEM/CT Professional Development Program improvements will be discussed later in Evaluation Question 3.

Evaluation Question 2

Evaluation question two examined how participating in the STEM/CT Professional Development Program affected paraprofessionals’ understanding of computational thinking. I answered this evaluation question using Levels Two and Four of Guskey’s Five Levels of Professional Development Evaluation Model (Guskey, 2002). Level Two assesses whether or not participants developed an understanding of CT during the year-long STEM/CT Professional Development Program. Level Four helps to determine whether or not new knowledge was applied or if the new skills and concepts that were taught were implemented. Overall, there were only minimal increases in group composite means for the two subconstructs measuring participants' understanding of computational thinking. Most participants self-reported incremental increases in their understanding of computational thinking, and interview responses explained how new skills and concepts regarding computational thinking were implemented using the STEM tools within their classrooms. My sample was too small to run any inferential statistics, but these findings are promising and suggest that the STEM/CT Professional Development Program was successful in helping participants develop their understanding of CT.

Understanding of Computational Thinking

Participants' self-reported understanding of computational thinking on the two sub-constructs and self-reported computational thinking ability increased throughout the year-long STEM/CT Professional Development Program. On the post-survey compared to the pre-survey, participants, on average, self-reported less difficulty breaking a problem down into smaller steps, demonstrating they developed their decomposition and abstraction skills. Additionally, on average, from pre- to post-survey, they self-reported less struggle generalizing solutions that can be applied to various problems. Participants also self-reported, on average, that they struggled less when identifying where and how to use variables to solve a problem and agreed that CT involved thinking logically to solve problems. These self-reported changes from pre- to post-survey indicate that participants increased their understanding of computational thinking and their computational thinking abilities.

Although there was a self-reported increase in participants' understanding of computational thinking based on survey results, some participants still expressed uncertainty regarding computational thinking in the reflective interviews. This result aligns with Ouyang et al.'s (2021) findings that even after a full year of professional development, teachers in their study still did not feel confident in certain aspects of CT. In contrast, in the reflective interviews, several participants shared anecdotes of how they understood computational thinking as a problem-solving process and engaged their students in various problem-solving tasks to help them engage in computational thinking. This understanding aligns with much of the literature that generally describes CT as an approach to problem solving (Grover & Pea, 2013) and the USBE (2019) definition of CT. A few participants also expressed how learning about CT gave them the vocabulary to talk about the skills and concepts they were trying to teach regarding computational thinking. These findings align with the current literature. For example, Hunsaker and West's (2020) study also found that educators participating in their study demonstrated an increased understanding of CT after participating in the study's professional development (badging) program. Yadav et al. (2018) also found that participants in their study conceptualize CT in various ways.

Implementing and Teaching Computational Thinking

Overall, the group mean measuring the teaching CS/CT construct increased. From pre- to post-survey, participants also self-reported an increased change in their abilities to explain basic computational thinking concepts to children. Additionally, they self-reported an increase in their ability to create computational thinking lessons at the appropriate level for their students and explain CT well enough to be effective in teaching CT to their students. Participants also self-reported an increased understanding that CT can be incorporated into the classroom by engaging students in problem-solving tasks while helping students understand how CT concepts are connected to their daily lives.

These self-reported changes from pre- to post-survey demonstrate, on average, an increase in participants' ability to implement and teach CT using the STEM tools provided to them. Additionally, participants self-reported an increase in their attitudes, values, and beliefs about computational thinking. All participants agreed that students can understand computational thinking concepts and should be taught to elementary students.

In the reflective interviews, participants also shared their experiences creating a STEM/CT integrated lesson plan and integrating CT into the STEM specialty using the STEM tools from the MSD STEM Lending Lab. Two participants did not feel ready to fully implement CT into their STEM specialty this year; however, they acknowledged and recognized the importance and benefits of integrating CT. Again, many participants shared anecdotes of using various tasks and challenges, such as guiding a robot through a maze, to engage their students in CT.

Evaluation Question 3

Evaluation question three examined the additional activities, resources, support, and training participants felt they still needed to design and teach STEM/CT integrated lessons using the STEM tools as part of the STEM specialty. I answered this evaluation question using four of Guskey's Five Levels of Professional Development Evaluation Model (Guskey, 2002). Overall, participants expressed several needs, including additional support, more time, money, and resources, additional CT instruction and practice, and lastly, many expressed a desire for more collaboration amongst paraprofessionals teaching CT as part of the STEM specialty.

Participants Needs

Support.

Some participants felt highly supported by their administration, while others participating in the program expressed the need for increased support from their administrator to teach STEM/CT integrated lessons better. This support includes allowing paraprofessionals time to regularly attend the STEM/CT Professional Development Program monthly PD sessions and support from the teachers to reinforce what they are trying to teach in the STEM specialty. I argue that this ultimately comes back to the participants' roles within their schools and who should support them. According to USBE (2023), the role of these "paraprofessionals" is a "paraeducator," and as such, they should be receiving instructional support from a licensed educator. Participants expressed the desire for more support at their individual schools. In the future, I recommend explicit discussion between MSD and individual elementary school administrators about who will provide the instructional support the paraprofessionals need.

Time, Money & Materials.

In the reflective interviews, the lack of preparation time was brought up in almost every interview. This finding aligns with what Ketelhut et al. (2020) also found: teachers need more time to plan CT-integrated lessons. In order to plan and create STEM/CT integrated lessons, paraprofessionals need the time (and resources) to do so. Many participants do not receive any preparation time, making preparing to integrate CT extremely difficult. Israel et al. (2015) found that teachers had more success integrating CT when there were opportunities to co-plan and co-teach. Due to the current lack of preparation time, MSD paraprofessionals spend their personal time planning or are unable to adequately prepare, adversely affecting their ability to integrate CT into the STEM specialty.

In conjunction with the lack of preparation time, most participants expressed their struggles with the lack of lessons and activities available to them to teach on the weeks they did not have a STEM tool from the MSD STEM Lending Lab. Figuring out what to teach and where to find the resources was difficult and stressful. Yadav et al. (2016) found that teachers must have PD that connects computational thinking and their current curriculum. While the STEM/CT Professional Development Program provided paraprofessionals with one introductory lesson for each STEM tool, participants would benefit from more lessons integrated with the STEM tools available to them. A few participants expressed the desire for more "unplugged" CT lessons that they could use before or after the STEM tools to reinforce CT concepts or prepare students to engage with CT and the STEM tool.

Additionally, several participants in their interviews expressed a lack of materials or budget to get the materials needed to teach STEM/CT integrated lessons in the classroom. This left participants feeling unprepared, so they contacted other organizations for supplies and teaching resources. Several participants shared that they spent their personal funds to purchase teaching resources or supplies for their students so they could feel prepared to be able to teach.

Another challenge regarding resources was the lack of access to STEM tools. Participants learned how to use the STEM tools in the summer but may not have access to them again for six or seven months and only have a day or two before teaching with the STEM tool to practice using it again and plan their lessons. Additionally, paraprofessionals were only allotted one or two weeks with each STEM tool. Participants shared in their interviews that this was not adequate time with the tool in the classroom to engage their students in computational thinking in as meaningful ways as they would have liked to. Additional time with the tool would allow them to do more STEM/CT-integrated activities in the classroom with the tools.

Lastly, a few participants expressed how the lack of pay for what they do and how much effort and time they put into teaching STEM/CT integrated lessons was discouraging. They are not just "helping" in a classroom; they plan and teach lessons daily and feel that additional compensation is well-deserved. Adequate compensation is vital in being able to retain paraprofessionals who are teaching the STEM specialty.

Additional CT Instruction & Practice.

In the post-survey, a few participants reported that they did not receive ample opportunities to practice the computational thinking skills they had been asked to learn and teach. Additionally, in the reflective interview, two participants expressed the desire to have more time dedicated to learning about and engaging in computational thinking as part of the monthly STEM/CT Professional Development Program. Providing additional time as part of the PD program spent on computational thinking would benefit all paraprofessionals and most likely continue to help them improve their understanding of and ability to integrate computational thinking in the STEM specialty.

Additional Time to Collaborate.

Lastly, several participants wanted more time to collaborate to share ideas and learn from each other. In the reflective interviews, participants shared how they would hold "parking lot" meetings, visit each other's classrooms, and correspond via email to share ideas and resources. Many expressed the desire to have additional time built into the STEM/CT Professional Development Program for them to collaborate and share.

Conclusions of Evaluation

Given the findings above, I can conclude that the STEM/CT Professional Development Program positively impacted most participants' understanding of computational thinking. That being said, it is important to note that gains were small (e.g., agree to strongly agree), and the reflective interviews revealed several challenges that participants faced. Participants had a positive experience and generally reported increased gains (albeit small) in their understanding of computational thinking. There are a few observations about the PD program and recommendations for the STEM/CT Professional Development Program that will enhance the experience and potentially improve paraprofessionals' understanding of computational thinking in the future.

Emphasis on CT Shifted Throughout the Year

When discussing the PD experience with participants in their reflective interviews and examining the PD monthly goals and activities, the focus on CT gradually shifted throughout the school year. This shift is due to several factors, many of which were to help address paraprofessionals' needs or further the school district's goals to provide equitable access to computing. For example, the school district began a pilot with the coding company Skill Struck. Therefore, a few of the month's PD sessions were with Skill Struck, training paraprofessionals on how to use the software. Additionally, paraprofessionals were given the opportunity to earn grant funds and gather new resources for their classroom by participating in another program, Science Partners, where community organizations such as the zoo, museums, botanical gardens, etc., would come to give a presentation and training on science lessons that they could teach in the STEM specialty class. These additional opportunities took up the monthly PD sessions and shifted the primary focus away from computational thinking. Therefore, there was no time to dedicate to computational thinking instruction and practice.

Computational Thinking Implementation

Although participants reported that their individual understanding improved throughout the STEM/CT Professional Development Program, their ability to implement CT more widely varied, as evidenced in their reflective interviews. This variation may be because paraprofessionals were provided little or no time to prepare STEM/CT integrated lessons. This conclusion aligns with prior research. For example, Ketelhut et al. (2020) also found that teaching CT through integration was also a struggle for the teachers in their study due to the lack of time to plan CT-integrated lessons. Li and colleagues (2019) and Sands and colleagues (2018) also found that teachers struggled to know how to integrate CT within their curriculum. In the reflective interviews I conducted, many participants expressed feeling overwhelmed and not knowing what to do with the lack of resources available to teach computational thinking.

CT was not the Main Focus of Teaching

As the STEM specialty class teacher, paraprofessionals were responsible for integrating CT with the STEM tools and exposing students to other science, technology, engineering, and math learning experiences. One participant was also expected to teach art because it was the Science, Technology, Engineering, Art, and Math (STEAM) specialty at her school. When paraprofessionals are only with their students for 30-40 minutes once per week, that does not leave much time for instruction with so many different topics to teach.

Limitations

This evaluation has several limitations, including evaluation scope, lack of classroom observations, lack of measures of student learning, having a sole evaluator, small sample size, non-diverse demographics, and the COVID-19 pandemic.

Scope of the Evaluation

First, I only evaluated a part of the logic model. This meant there might be areas of the STEM/CT Professional Development Program that were not evaluated because they were not a part of the logic model selected to be evaluated. The PD program is also specific to the STEM Action Center grant received and the Marshall School District. Therefore, the findings may not broadly apply in other situations and contexts.

Lack of Classroom Observations

Secondly, due to the COVID-19 pandemic and the evaluation scope, I had limited opportunities to observe participants teaching STEM/CT integrated lessons using the STEM tools in the classroom. These limited opportunities greatly reduced the ability to evaluate how well the participants understood CT. It was also a limitation in determining paraprofessionals' additional needs when preparing and teaching STEM/CT integrated lessons using the STEM tools.

Sole Evaluator

I was the primary source for the data collected from participants. I strived to eliminate the impact of any bias that I may have. One challenge of qualitative research is the researcher's involvement in the data collection and analysis process (Creswell, 2014). I attempted to confirm the data by triangulating the data between survey results, lesson plans, and interview data. For example, if the survey showed that most participants were unsure what computational thinking is, that should also be reflected in the answers to the interview questions. Another example could be that if the participants demonstrate an understanding of computational thinking in their description of a STEM/CT integrated lesson plan they have taught, the post-survey results should also demonstrate an increased understanding of CT.

Small Sample Size

Additionally, the sample for this evaluation was small (n=8). Because there were so few participants, it was difficult to pinpoint patterns and trends in the descriptive statistics and qualitative data. To help counteract the small sample size, I developed an in-depth interview protocol to garner rich descriptions to supplement survey data.

Non-Diverse Demographics

Also, the sample size was not diverse in age, gender, or race/ethnicity. The entire sample size was females, with the majority over the age of 50. The majority identified as White/Caucasian. Many races and ethnicities were not represented in the sample, limiting my ability to generalize.

COVID-19 Pandemic

Lastly, this evaluation was conducted during the COVID-19 pandemic. COVID-19 impacted professional development attendance. Professional development sessions were canceled due to illness, and participants' attendance was affected by quarantine, illness, etc. Additionally, virtual learning days may have interrupted participants' implementation of what was learned in professional development sessions. Lastly, some

STEM tech tools could not be used in the classroom because they could not be appropriately sanitized between classes (e.g., they could not wipe or spray the Makey Makey with disinfectant).

Recommendations for Practice

MSD District STEM/CS Consultant & STEM/CT Professional Development Program

The MSD District STEM and Computer Science Consultant was responsible for designing, implementing, and delivering the STEM/CT Professional Development Program. Below, I provide five recommendations to consider in the future design and implementation of the STEM/CT Professional Development Program and other similar programs that will help to continue to provide a positive experience for paraprofessionals participating in the program, further paraprofessional knowledge of computational thinking, and address the current needs of participants participating in the STEM/CT Professional Development Program.

Recommendation 1: Create a Community of Practice for Paraprofessionals

First, one recommendation for the future STEM/CT Professional Development Program and similar programs is to implement a professional development model that enables paraprofessionals to collaborate more. One possible solution is centered around a Community of Practice (Lave & Wenger, 1991) professional development model. A Community of Practice is when novices and experts come together and learn from each other and support each other as the novices acquire learning.

This model has been used or recommended in prior work and research on teaching CT to teachers (Hestness et al., 2018; Killen et al., 2020; Sands et al., 2018). Computational thinking is a complex topic and sometimes difficult to develop and integrate into current teaching practice. Yadav et al. (2016) recommended that Communities of Practice be used as part of CT professional development to provide additional practice, support, and reinforce teacher learning. This conclusion is supported by Israel et al.'s (2015) findings that for teachers to plan and teach CT-integrated lessons, they need support and coaching. Killen and colleagues (2020) implemented a Community of Practice as part of the CT PD and found great success. Teachers could share expertise and ask for help, fill gaps in their knowledge, receive mentorship, and teachers reported an easier time implementing CT into their lessons.

Those participating in the STEM/CT Professional Development Program have requested additional time to collaborate with the other paraprofessionals and actively sought out opportunities on their own to do so. Providing paraprofessionals with a structured Community of Practice can greatly enhance their learning of CT and their ability to implement STEM/CT integrated activities and lessons into the STEM specialty class, thus helping to provide equitable access to computing for all students they serve.

Recommendation 2: More Explicit CT Instruction

Second, greater emphasis should be placed on explicit CT instruction at every monthly PD session. Table 1 in Chapter Three reveals that most professional development sessions were not explicitly focused on CT instruction. Two participants requested an increased focus on CT in the reflective interviews. Additionally, more explicit CT instruction will better support paraprofessionals' understanding and implementation of CT. This recommendation aligns with the literature illustrated in Chapter Two regarding extended/continuous PD as well as with Li et al. (2019) findings that, due to the difficult nature of CT, PD needs to engage educators in adequate hands-on experiences and learning opportunities with CT to support them in becoming proficient in CT concepts. The following paragraph explains how the monthly PD session could be modified to include more explicit CT instruction.

The first 20 minutes of each PD session could be dedicated to developing an understanding of CT or implementing a CT practice. Several learning activities could be used during this time. For example, a mini-lesson where paraprofessionals are the "students" and the STEM/CS consultant models a short lesson around the CT practice. Alternatively, a short video of the skill taught could be shown and discussed with the paraprofessionals. Potential discussion questions could include a) What did they notice? b) How were students engaging in a specific CT component? c) How did the educator support students? d) What could be improved? Additionally, the STEM/CS consultant could use a STEM tool and engage the paraprofessionals in a task that requires them to engage in CT, for example, providing them with a Sphero Bolt and code that needed to be debugged to get the Sphero Bolt through a maze correctly followed by a short discussion of how they each engaged in the CT practice.

Recommendation 3: STEM Tools Connected to CT Components

After evaluating this STEM/CT Professional Development Program, I determined that a greater effort could be made to connect the STEM tools to specific CT components to support paraprofessionals with STEM/CT integration. This more explicit connection when learning how to use the STEM tools in the summer PD sessions could support paraprofessionals in integrating CT using the STEM tools during the school year. For example, the Sphero Bolt is perhaps a better tool for engaging students in debugging tasks than ClassVR. Providing paraprofessionals with ideas of how specific CT components can be taught using each STEM tool would be beneficial in helping them connect their understanding of CT to the STEM tools and support them in integrating CT when using the STEM tools in the STEM specialty class. The research supporting this recommendation is still emerging. In 2020, Li et al.'s review found a lack of research on STEM education and CT integration. Additionally, Sands et al. (2018) found that how educators conceptualized CT suggests “that many educators have very little knowledge about what these skills are and lack awareness of how these skills can be implemented in their classrooms” (p. 161). Therefore, having explicit instruction around STEM tools and the CT components has the potential to be extremely beneficial for paraprofessionals learning to teach STEM/CT integrated lessons.

Recommendation 4: Provide Additional Resources and Materials

A challenge for participants participating in the STEM/CT Professional Development Program was the lack of additional resources and materials to use outside of the single lesson provided with each STEM tool. I recommend providing several additional mini-lessons. The mini-lessons should include several “unplugged” activities that teach the various CT components and several “plugged” mini-lessons for each STEM tool that emphasize a particular CT component. Additionally, these mini-lessons should be provided for each grade level (K-6) or can be easily modified or adapted for the various grade levels.

This recommendation also supports the current guidelines from the Utah State Board of Education that state the paraprofessionals “may not be solely responsible for designing lesson plans” (USBE, 2023, p.9) and “may not be responsible for selecting programming or prescribing educational activities or materials for the students without the supervision and guidance of an appropriately licensed teacher” (USBE, 2023, p.9). Additionally, current guidelines state that “the supervising educator prepares the lesson and plans the instructional support activities the paraeducator [paraprofessional] will implement” (USBE, 2023, p. 10). By providing additional resources and materials, paraprofessionals will be better equipped to integrate CT into the STEM specialty class using the MSD STEM Lending Lab tools and technologies.

Recommendation 5: Fewer STEM tools and More Time with the STEM Tools (Quality over Quantity)

I recommend reducing the number of STEM tools training provided during the STEM/CT Professional Development Program. Then, I suggest decreasing the number of STEM tools and increasing the amount of the chosen STEM tools so that more tools are available within the district. When selecting which STEM tools to purchase additional quantities of and to focus design instruction for, it may be helpful to consult work done by Hamilton et al. (2020) and Yu and Rouque (2019), who have examined various computational toys and kits for children. Lastly, I recommend increasing the time each paraprofessional(s)/schools get with each STEM tool. Many participants expressed the desire for more time with each STEM tool. With only seeing their students once per week for 30-40 minutes and only having the STEM tool for one or two weeks, participants did not feel they had sufficient time with the STEM tool to engage students deeply in computational thinking. Students are so excited about the STEM tools that the first class is usually just spent exploring the STEM tool and how to use it. An introductory lesson is taught in the next class, but the tool must be returned. If paraprofessionals and students had access to the STEM tool for a longer period, more CT integration would be able to occur.

Many of these five recommendations are feasible within the current structure of the STEM/CT Professional Development Program. For example, recommendation #1 to create a Community of Practice is highly feasible. Paraprofessionals could be assigned to groups to collaborate with paraprofessionals from both Phase 1 and Phase 2 of the STEM/CT Professional Development Program at the monthly Elementary STEM PLC meetings and asynchronously using a messaging tool such as Slack or WhatsApp. This would allow paraprofessionals to collaborate and share ideas more easily with each other throughout the school year. Additionally, a Community of Practice has the potential to provide more time in the Elementary STEM PLC meetings to focus on CT instruction (recommendation #2) since other concerns paraprofessionals may have would have been addressed asynchronously throughout the month. Recommendation #4 and Recommendation #5 would require additional funding for curriculum development and the purchase of additional STEM tools. While recommendations #4 and #5 are feasible, the cost may be prohibitive for the school district.

Administrators

Level three of Guskey's Five Levels of Professional Development Evaluation model (Guskey, 2002) examines organizational support and change. Although level three was not part of this evaluation, two key themes emerged from the reflective interviews that require additional organizational support and change outside of what the STEM/CT Professional Development Program can provide. These two themes are addressed below. During the year-long evaluation, I observed significant attrition of STEM specialty instructional assistants within MSD. The first two recommendations below may also help to lessen the attrition rate in future years, as paraprofessionals will have the support and time they need to do their jobs adequately.

Preparation Time

First, some STEM/CT Professional Development Program participants received little or no preparation time. This lack of preparation time directly impacted their ability to plan and implement STEM/CT integrated lessons using the STEM tools because they did not have the time to do so. If some paraprofessionals receive preparation time to plan and implement STEM/CT integrated lessons while others do not, this also directly impacts the ability to provide equitable access to computing education for all students within MSD. Thus, I recommend that MSD adopt a policy that provides all their STEM specialty instructional assistants within the district adequate preparation time to plan and prepare to teach STEM/CT integrated lessons and activities or that sufficient pre-prepared resources, lessons, and activities are provided to paraprofessionals, so they can adequately do their job with the limited (or none) preparation time currently provided.

Compensation

Additionally, many STEM/CT Professional Development Program participants desired improved compensation. Participants expressed that they felt they were not adequately compensated for their work and often put in hours that were not compensated because they wanted to be prepared to teach their lessons to their students. One participant joked that she could make more money working at McDonald's or Walmart, which she found discouraging. Another participant expressed that she often does the work of a teacher for the pay of an assistant, which she found frustrating. Increasing the compensation or decreasing the expectations of paraprofessionals will benefit them greatly in feeling they are being compensated fairly for their time. Unfortunately, providing adequate compensation presents budgetary implications that MSD may be unable to overcome. Therefore, as the literature has shown (Ketelhut et al., 2020; Israel et al., 2015; P.J. Rich et al., 2021), integration within the general education classroom may be a more feasible, appropriate model to consider for the future.

Integration in the General Education Classroom with a Teacher

Lastly, the administration must consider the most appropriate placement for computer science and computational thinking instruction. Is the STEM specialty, where an unlicensed teacher teaches it, the most appropriate place for the Utah K-5 CS standards to be taught? On top of the difficult subject matter, paraprofessionals often lack the skills and abilities of a licensed classroom teacher. For example, several participants expressed difficulty with classroom management, planning lessons and activities, scaffolding instruction for the various grade levels, etc. These are often skills licensed classroom teachers have mastered and do not pose as much of a challenge. There is a lot of current research and materials being developed to support the integration of CS/CT in the elementary classroom (Chipps et al., 2022; Century et al., 2020; Shumway et al., 2021; Silvis et al., 2022; Waterman et al., 2020) and it may be considered as a more viable, long-term solution to providing equitable access to computing education for all students within Marshall School District.

Recommendations for Research

Improved Research on Computational Thinking in Practice

Elementary teachers often lack the understanding of how to integrate computational thinking into their classrooms and what it may look like (Yadav et al., 2019), and there are limited studies that look at how to teach computational thinking to elementary students (Ottenbreit-Leftwich & Yadav, 2022). In educational research, researchers do not agree on how to best approach teaching and integrating computational thinking in the K-5 context. For example, researchers such as P.J. Rich (personal communication) and Williams (2022) argue that elementary students should be taught coding to learn computational thinking. However, other researchers, such as Yadav et al. (2019), argue that beginning with coding in elementary education is not the best place to start, and computational thinking instruction should begin through “unplugged” instruction in the classroom.

There is a large body of research that defines computational thinking (e.g., Wing, 2008; Selby & Woollard, 2014; Grover & Pea, 2013), identifies various CT practices and concepts (Brennan & Resnick, 2012; Weintrop et al., 2016), provides frameworks or conceptual models (Kotsopoulos et al., 2017), identifies barriers (Stokke, 2019) or describes *in theory* what K-5 computer science and computational thinking integrated instruction *may* look like. For example, K.M. Rich and colleagues (2020) describe the common practices that mathematical thinking and computational thinking share and the implications for developing integrated instruction of math and CT. Additionally, several recent studies or published research have looked at pre-service elementary teachers learning of computational thinking (Kaya et al., 2019; Moran et al., 2020).

However, limited research demonstrates what CT practices look like in an elementary school classroom, as much of the current research has focused on secondary education. Luo et al. (2022) describe research and current practices around CT in the elementary as being in the “infant stage” (p.19:5). Additionally, Luo et al. (2022) found that the majority of CT literature has primarily examined CT as a “product” of student learning rather than research that examines *how* students develop CT ability. Ottenbreit-Leftwich and Yadav (2022) have determined that the research field “still needs to identify developmentally appropriate practices and learning goals for elementary students” (p. vi). Several approaches to teaching CT include integration, stand-alone CS courses, unplugged v. plugged, through coding toys, etc. There is a need for additional research that examines these approaches, models what age-appropriate CT instruction looks like, and how to better prepare elementary teachers and paraprofessionals teaching computer science and computational thinking.

Teaching Computational Thinking to Paraprofessionals

Lastly, there is very limited, if any, research on professional development for paraprofessionals (outside of special education). Paraprofessionals may face challenges when participating in professional development as they may lack pedagogical knowledge (how to teach) and the specific content knowledge that many teachers already possess. If professional development is not designed to address these knowledge gaps, paraprofessionals may remain unprepared to teach after participating in the professional development. Douglas et al. (2019) found that the current training materials for paraprofessionals often do not use evidence-based practices for adult learners. There is a lack of research on how to teach computational thinking to paraprofessionals as they have unique needs compared to pre-service or classroom teachers. There is a need for more understanding on how to best prepare paraprofessionals to understand and teach computational thinking and how to best design professional development opportunities to meet the needs of paraprofessionals tasked with teaching CS/CT.

Conclusion

As part of the Marshall School District's efforts to provide equitable access to computing education for all students, it was determined that paraprofessionals would teach computational thinking as part of the STEM specialty class. The STEM Action Center of Utah funded this effort. Paraprofessionals were provided training and support to teach STEM/CT integrated activities and lessons through the STEM/CT Professional Development Program. This evaluation found that the PD program was well received by participants and was generally successful in helping participants develop an understanding of computational thinking. The evaluation also found that there are several additional supports that paraprofessionals need, including more CT instruction and practice, additional time with the STEM tools, more teaching resources, additional time to collaborate with other paraprofessionals, preparation time, and additional money and materials for the classroom, and increased support from administration. There is a need in the research field to identify and develop age-appropriate teaching strategies and learning objectives for computational thinking at the elementary level. Additionally, more research is needed to demonstrate and understand what learning computational thinking looks like in the elementary setting. Lastly, there is a need for more research on professional development for paraprofessionals and how to prepare them best to teach and integrate computational thinking.

References

Aho, A. V. (2012). Computation and computational thinking. *Computer Journal*, *55*(7), 832–835. https://doi.org/10.1093/comjnl/bxs074

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking curriculum framework: Implications for teacher knowledge. *Journal of Educational Technology & Society*, *19*(3), 47–57. http://www.jstor.org/stable/jeductechsoci.19.3.47

Archibald, S., Coggshall, J., Croft, A., & Goe, L. (2011). High-quality professional development for all teachers: Effectively allocating resources. *National Comprehensive Center for Teacher Quality.* https://files.eric.ed.gov/fulltext/ED520732.pdf

Ashbaker, B. Y., & Morgan, J. (2001). Paraeducators: A powerful human resource. *Streamlined Seminar*, *19*(2), 1-5. https://eric.ed.gov/?id=ED453573

Barefoot Computing. (n.d.). *Computational thinking*. https://www.barefootcomputing.org/resources/computational-thinking-poster

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved, and what is the role of the computer science education community? *ACM Inroads*, *2*(1), 48–54. https://doi.org/10.1145/1929887.1929905

Bell, T., & Lodi, M. (2019). Constructing computational thinking without using computers. *Constructivist Foundations*, *14*(3), 342–351.

Bonilla, S., & Paul, B. (2019). *Utah Computer Science Education Master Plan*. https://schools.utah.gov/file/abb4c31f-f599-4ec5-b57e-3a51404506ba

Brennan, K., & Resnick, M. (2012, April 13-17). *New frameworks for studying and assessing the development of computational thinking*. American Educational Research Association Conference, Vancouver, Canada.

Cansu, S. K., & Cansu, F. K. (2019). An Overview of Computational Thinking. *International Journal of Computer Science Education in Schools*, *3*(1). http://files.eric.ed.gov/fulltext/EJ1214682.pdf

Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & Education*, *109*, 162–175. https://doi.org/10.1016/j.compedu.2017.03.001

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

Chipps, J., Dangol, A., Fasy, B.T., Hancock, S., Jiang, M., Rogowski, A., Searle, K.A., & Tofel-Grehl, C. (2022). *Culturally responsive storytelling across content areas using American Indian ledger art and physical computing*. American Society for Engineering Education Annual Conference (Paper ID # 38431). Minneapolis, MN. American Society of Engineering Education.

Clarke, D., & Hollingsworth, H. (2002). Elaborating a model of teacher professional growth. *Teaching and Teacher Education*, *18*(8), 947–967. https://doi.org/10.1016/S0742-051X(02)00053-7

Clarke-Midura, J., Shumway, J., Beck, K., Shehzad, U., & Recker, M. (2022). Cache Code Math September Unit: Repeats in Mathematics and Programming. *Instructional resources.* Paper 2. https://digitalcommons.usu.edu/eled_support_instructional/2

Computer Science Teacher Association (CSTA), & International Society for Technology in Education (ISTE). (2011). *Operational definition of computational thinking for K-12 education*. Retrieved from CSTA website http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf.

Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches.* SAGE Publishing.

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). Computational thinking: A guide for teachers. *Computing at School.*

Cuny, J., Snyder, L., & Wing, J.M. (2010). Demystifying computational thinking for non-computer scientists. Unpublished manuscript in progress, referenced in http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf

Davis School District (DSD). (n.d.). *What's happening.* https://mhs.davis.k12.ut.us/parents-students/cte-career-and-technical-education/information-technology/utah-k-12-computer-science-plan/whats-happening

Darling-Hammond, L., Hyler, M. E., & Gardner, M. (2017). *Effective teacher professional development*. Palo Alto, CA: Learning Policy Institute. https://learningpolicyinstitute.org/product/teacher-prof-dev

Denzin, N. K., & Lincoln, Y. S. (2003). *The landscape of qualitative research theories and issues*. SAGE Publishing

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

Desimone, L. M., & Pak, K. (2017). Instructional coaching as high-quality professional development. *Theory into Practice*, *56*(1), 3–12. https://doi.org/10.1080/00405841.2016.1241947

Dickson, S., & Voorhies, L. (2023). *Paraeducator manual*. Utah State Board of Education. Retrieved July 24, 2023, from https://www.schools.utah.gov/file/0ecaf200-77b2-4f85-aa44-3f17af07c9e4

Douglas, S. N., Uitto, D. J., Reinfelds, C. L., & D'Agostino, S. (2019). A systematic review of paraprofessional training materials. *The Journal of Special Education*, *52*(4), 195–207. https://doi.org/10.1177/0022466918771707

Etscheidt, S. (2005). Paraprofessional services for students with disabilities: A legal analysis of issues. *Research and Practice for Persons with Severe Disabilities: The Journal of TASH*, *30*(2), 60–80. https://doi.org/10.2511/rpsd.30.2.60

Gallup & Code with Google. (2020). *Current perspectives and continuing challenges in computer science education in U.S. K-12 Schools.* Gallup, Inc. Retrieved July 26, 2023, from https://services.google.com/fh/files/misc/computer-science-education-in-us-k12schools-2020-report.pdf

Google (n.d.). *Computational Thinking Concepts Guide*. https://docs.google.com/document/d/1i0wg-BMG3TdwsShAyH_0Z1xpFnpVcMvpYJceHGWex_c/edit

Google for Education (n.d.). *Computational Thinking*. Retrieved July 26, 2023, from https://edu.google.com/latest-news/future-of-the-classroom/computational-thinking/

Gretter, S., Yadav, A., Sands, P., & Hambrusch, S. (2019). Equitable learning environments in K-12 computing. *ACM Transactions on Computing Education*, *19*(3), 1–16. https://doi.org/10.1145/3282939

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

Grover, S., & Pea, R. (2018). Computational thinking: A competency whose time has come. In S. Sentance, E. Barendsen, & C. Schulte (Eds.), *Computer science education: Perspectives on teaching and learning in school* (pp.19-38). Bloomsbury Academic. https://doi.org/10.1080/03004279.2019.1617329

Guskey, T. R. (1986). Staff development and the process of teacher change. *Educational Researcher*, *15*(5), 5–12. https://doi.org/10.2307/1174780

Guskey, T. R. (2000). *Evaluating professional development*. Corwin Press.

Guskey, T. R. (2002). Does it make a difference? Evaluating professional development. *Educational Leadership*, *59*(6), 45–51.

Guskey, T. R. (2005). Taking a second look at accountability. *Journal of Staff Development*, *26*(1), 10–18.

https://search.ebscohost.com/login.aspx?direct=true&db=eue&AN=507849349&site=ehost-live

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

Guzdial, M., Kay, A., Norris, C., & Soloway, E. (2019). Computational thinking should just be good thinking. *Communications of the ACM*, *62*(11), 28–30. https://doi.org/10.1145/3363181

Hamilton, M., Clarke-Midura, J., Shumway, J. F., & Lee, V. R. (2020). An emerging technology report on computational toys in early childhood. *Technology, Knowledge and Learning*, *25*(1), 213–224. https://doi.org/10.1007/s10758-019-09423-8

Hanover Research. (2014, June). *Professional Development Program Evaluation: Frameworks and Tools*. Retrieved July 26, 2023, from https://www.apsva.us/wp-content/uploads/2018/10/Professional-Development-Program-Evaluation-Frameworks-and-Tools-Arlington-Public-Schools.pdf

Harris, C. (2018). *Computational thinking unplugged: Comparing the impact on confidence and competence from analog and digital resources in computer science professional development for elementary teachers* [Doctoral dissertation, St. John Fisher College]. https://fisherpub.sjfc.edu/education_etd/374/

Hauerwas, L. B., & Goessling, D. P. (2008). Who are the interventionists? Guidelines for paraeducators in RTI. *Teaching Exceptional Children Plus*, *4*(3), 2-13. https://files.eric.ed.gov/fulltext/EJ967479.pdf

Hestness, E., Ketelhut, D. J., Randy McGinnis, J., & Plane, J. (2018). Professional knowledge building within an elementary teacher professional development experience on computational thinking in science education. *Journal of Technology and Teacher Education*, *26*(3), 411–435. https://www.learntechlib.org/p/181431/

Hunsaker, E. (2020). Computational thinking. In A. Ottenbreit-Leftwich & R. Kimmons (Eds.), *The K-12 educational technology handbook*. EdTech Books. https://edtechbooks.org/k12handbook/computational_thinking

Hunsaker, E., & West, R. E. (2020). Designing computational thinking and coding badges for early childhood educators. *TechTrends: Linking Research and Practice to Improve Learning*, *64*(1), 7–16. https://doi.org/10.1007/s11528-019-00420-3

ISTE (*n.d.*). *Computational thinking competencies* https://iste.org/standards/computational-thinking-competencies

Israel, M., Pearson, J. N., Tapia, T., Wherfel, Q. M., & Reese, G. (2015). Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. *Computers & Education*, *82*, 263–279. https://doi.org/10.1016/j.compedu.2014.11.022

K-12 Computer Science Framework Steering Committee. (2016). *K-12 Computer Science Framework* [Technical Report]. Association for Computing Machinery. https://dl.acm.org/doi/abs/10.1145/3079760

Kale, U., Akcaoglu, M., Cullen, T., Goh, D., Devine, L., Calvert, N., & Grise, K. (2018). Computational what? Relating computational thinking to teaching. *TechTrends*, *62*(6), 574–584. https://doi.org/10.1007/s11528-018-0290-9

Kafai, Y. B. (2016). From computational thinking to computational participation in K--12 education. *Communications of the ACM*, *59*(8), 26–27. https://doi.org/10.1145/2955114

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

Ketelhut, D. J., Mills, K., Hestness, E., Cabrera, L., Plane, J., & McGinnis, J. R. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science. *Journal of Science Education & Technology*, *29*(1), 174–188. https://doi.org/10.1007/s10956-019-09798-4

Killen, H., Coenraad, M., Byrne, V., Cabrera, L., & Ketelhut, D. J. (2020). Reimagining computational thinking professional development: Benefits of a community of practice model. In *Proceedings of the International Conference of the Learning Sciences (ICLS 2020)* (pp. 2125-2132. https://par.nsf.gov/servlets/purl/10204993

Kirkpatrick, J. D., & Kirkpatrick, W. K. (2016). *Kirkpatrick's four levels of training evaluation*. Association for Talent Development.

Knowlton, L. W., & Phillips, C. C. (2012). *The logic model guidebook: Better strategies for great results*. SAGE Publishing.

Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A pedagogical framework for computational thinking. *Digital Experiences in Mathematics Education*, *3*(2), 154–171. https://doi.org/10.1007/s40751-017-0031-2

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, *2*(1), 32–37. https://doi.org/10.1145/1929887.1929902

Lewis, K. C. (2004). Instructional aides: Colleagues or cultural brokers? *School Community Journal*, *14*(1), 91–112. https://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=EJ794830&site=ehost-live

Li, Q., Richman, L., Haines, S., & McNary, S. (2019). Computational thinking in classrooms: A study of a PD for STEM teachers in high needs schools. *Canadian Journal of Learning and Technology*, *45*(3), 1-21. http://files.eric.ed.gov/fulltext/EJ1252335.pdf

Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D., & Duschl, R. A. (2020). On computational thinking and STEM education. *Journal for STEM Education Research*, *3*(2), 147–166. https://doi.org/10.1007/s41979-020-00044-w

Luo, F., Israel, M., & Gane, B. (2022). Elementary computational thinking instruction and assessment: A learning trajectory perspective. *ACM Transactions on Computing Education*, *22*(2), 1–26. https://doi.org/10.1145/3494579

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, *41*, 51–61. https://doi.org/10.1016/j.chb.2014.09.012

Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014). Computational thinking in K-9 education. *Proceedings of the Working*

Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference, 1–29. https://doi.org/10.1145/2713609.2713610

Mishra, P., & Yadav, A. (2013). Rethinking technology & creativity in the 21st century. *TechTrends*, *57*(3), 10–14. https://doi.org/10.1007/s11528-013-0655-z

Moran, R., Roberston, L., Tai, C., Keith, K. J., Price, J., Meier, L. T., & Hong, H. (2020). Preparing pre-service teachers for the future: Computational thinking as a scaffold for critical thinking. In J. Keengwe & P. Wachira (Eds.), *Handbook of research on integrating computer science and computational thinking in K-12 education* (pp. 17–29). IGI Global. https://doi.org/10.4018/978-1-7998-1479-5.ch002

Newton, X. A., Poon, R. C., Nunes, N. L., & Stone, E. M. (2013). Research on teacher education programs: Logic model approach. *Evaluation and Program Planning*, *36*(1), 88–96. https://doi.org/10.1016/j.evalprogplan.2012.08.001

NGSS Lead States. 2013. *Next Generation Science Standards: For States, By State*s. The National Academies Press

Ottenbreit-Leftwich, A., & Yadav, A. (2022). Introduction: Computational thinking in preK-5: Empirical evidence for integration and future directions. In A. Ottenbreit-Leftwich & A. Yadav (Eds.), *Computational thinking in preK-5: Empirical evidence for integration and future directions* (pp. iii – vi). Association for Computing Machinery. https://doi.org/10.1145/3507951.3519281

Ouyang, Y., Hayden, K. L., & Remold, J. (2018). Preparing upper elementary school teachers for integrating computational thinking into regular classroom activities. *Journal of Computing Sciences in Colleges*, *33*(4), 165–171. https://dl.acm.org/doi/abs/10.5555/3199572.3199594

Perković, L., Settle, A., Hwang, S., & Jones, J. (2010). A framework for computational thinking across the curriculum. *Proceedings of the Fifteenth Annual Conference on Innovation and Technology in Computer Science Education*, 123–127. https://doi.org/10.1145/1822090.1822126

Rich, K. M., Yadav, A., & Larimore, R. A. (2020). Teacher implementation profiles for integrating computational thinking into elementary mathematics and science instruction. *Education and Information Technologies*, *25*(4), 3161–3188. https://doi.org/10.1007/s10639-020-10115-5

Rich, K. M., Yadav, A., & Schwarz, C. V. (2019). Computational thinking, mathematics, and science: Elementary teachers' perspectives on integration. *Journal of Technology and Teacher Education*, *27*(2), 165–205. http://dx.doi.org/

Rich, P. J., Hu, H., Christensen, J., & Ellsworth, J. (2019). *The landscape of computing education in Utah*. https://eric.ed.gov/?id=ED604662

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

Rich, P. J., Mason, S. L., & O'Leary, J. (2021). Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking. *Computers & Education*, *168*, 104196. https://www.sciencedirect.com/science/article/pii/S0360131521000737

Saldaña, J. (2016). *The coding manual for qualitative researchers.* SAGE Publishing.

Sands, P., Yadav, A., & Good, J. (2018). Computational thinking in K-12: In-service teacher perceptions of computational thinking. In M. S. Khine (Ed.), *Computational thinking in the STEM disciplines: Foundations and research highlights* (pp. 151–164). Springer International Publishing. https://doi.org/10.1007/978-3-319-93566-9_8

Savaya, R., & Waysman, M. (2005). The logic model: A tool for incorporating theory in development and evaluation of programs. *Administration in Social Work*, *29*(2), 85–103. https://doi.org/10.1300/J147v29n02_06

Selby, C., & Woollard, J. (2014). *Refining an understanding of computational thinking*. https://eprints.soton.ac.uk/372410/1/372410UnderstdCT.pdf

Shumway, J. F., Welch, L. E., Kozlowski, J. S., Clarke-Midura, J., & Lee, V. R. (2021). Kindergarten students' mathematics knowledge at work: The mathematics for programming robot toys. *Mathematical Thinking and Learning*, 1–29. https://doi.org/10.1080/10986065.2021.1982666

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

Silvis, D., Clarke-Midura, J., Shumway, J. F., Lee, V. R., & Mullen, S. (2022). Children caring for robots: Expanding computational thinking frameworks to include a technological ethic of care. *International Journal of Child-Computer Interaction*, *33*, 100491. https://doi.org/10.1016/j.ijcci.2022.100491

STEM Action Center (n.d.) *STEM Computing Partnership Grant*. https://stem.utah.gov/educators/funding/computing-partnership-grant/

Stokke, T. L. (2019). *An exploratory study to identify barriers to implementation of computational thinking* [Doctoral dissertation, The University of North Dakota]. https://commons.und.edu/cgi/viewcontent.cgi?article=3591&context=theses

Taslibeyaz, E., Kursun, E., & Karaman, S. (2020). How to develop computational thinking: A systematic review of empirical studies. *Informatics in Education*, *19*(4), 701–719. https://doi.org/10.15388/infedu.2020.30

Taylor-Powell, E., & Henert, E. (2008). *Developing a logic model: Teaching and training guide*. University of Wisconsin-Extension Cooperative Program Development and Evaluation. Retrieved July 26, 2023, from https://fyi.extension.wisc.edu/programdevelopment/files/2016/03/lmguidecomplete.pdf

Tsarava, K., Moeller, K., Román-González, M., Golle, J., Leifheit, L., Butz, M. V., & Ninaus, M. (2022). A cognitive definition of computational thinking in primary education. *Computers & Education*, *179*, 104425. https://doi.org/10.1016/j.compedu.2021.104425

U.S. Bureau of Labor Statistics. (2022, October 4). *Teacher assistants: Occupational Outlook Handbook*. U.S. Bureau of Labor Statistics. Retrieved April 10, 2023, from https://www.bls.gov/ooh/education-training-and-library/teacher-assistants.html

U.S. Census Bureau. (2021). *Utah Was Fastest-Growing State From 2010 to 2020*. Retrieved August 28, 2023, from https://www.census.gov/library/stories/state-by-state/utah-population-change-between-census-decade.html

Utah Code Annotated § 53F-2-411. (2022). Available at: https://www.lawserver.com/law/state/utah/archv-2022-ut-code/2022_utah_code_53f-2-411

Utah State Board of Education. (n.d.). *Paraeducator to Teacher Scholarship (PETTS) Program Information Sheet. Utah State Board of Education Rules R277-321* https://www.schools.utah.gov/file/734220e3-8896-450a-8bfb-a0b60496943a

Utah State Board of Education (2023). *USBE Paraeducator Manual.* https://www.schools.utah.gov/file/0ecaf200-77b2-4f85-aa44-3f17af07c9e4

Utah State Board of Education. (2019). *Utah K-5 computer science standards.* https://www.utah.gov/pmn/files/500299.pdf

Warner, J., Fletcher C., & Garbrecht, L. (2019). *Better data, better progress: Methods for measuring inequities in computer science education.* AERA, Toronto, Canada

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

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Weiss, C. H. (2004). Rooting for evaluation: A CliffNotes version of my work. In M.C. Alkin (Ed.) *Evaluation roots: Tracing theorists' views and influences*, (pp. 154-168). SAGE Publishing.

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

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, *366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends: Linking Research & Practice to Improve Learning*, *60*(6), 565–568. https://doi.org/10.1007/s11528-016-0087-7

Yadav, A., Krist, C., Good, J., & Caeli, E. N. (2018). Computational thinking in elementary classrooms: Measuring teacher understanding of computational ideas for teaching science. *Computer Science Education*, *28*(4), 371–400. https://doi.org/10.1080/08993408.2018.1560550

Yadav, A., Larimore, R., Rich, K., & Schwarz, C. (2019). Integrating computational thinking in elementary classrooms: Introducing a toolkit to support teachers. *Society for Information Technology & Teacher Education International Conference* (pp. 347–350). https://www.learntechlib.org/p/208366/

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J.T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education*, *14*(1), 1-16. https://doi.org/10.1145/2576872

Yadav, A., Ocak, C., & Oliver, A. (2022). Computational Thinking and Metacognition. *TechTrends*. https://doi.org/10.1007/s11528-022-00695-z

Yadav, A., Stephenson, C., & Hong, H. (2017). Computational thinking for teacher education. *Communications of the ACM*, *60*(4), 55–62. https://doi.org/10.1145/2994591

Yu, J., & Roque, R. (2019). A review of computational toys and kits for young children. *International Journal of Child-Computer Interaction*, *21*, 17–36. https://doi.org/10.1016/j.ijcci.2019.04.001

APPENDICES

Appendix A: List of Select Journals and all Databases Searched

Database	Select Included Journals
Academic Search Ultimate	*Communications of the ACM* *Constructivist Foundations* *Educational Technology Research & Development* *Journal of Cognitive Science* *Journal of College Science Teaching* *Journal of Science Education & Technology* *Proceedings of the ASEE Annual Conference & Exposition* *TechTrends: Linking Research & Practice to Improve Learning*
Computer Source	*Communications of the ACM* *Computer Applications in Engineering Education* *Computer Science Education* *Interactive Learning Environments* *Journal of Educational Computing Research*
Education Source	*Computers & Education* *Constructivist Foundations* *Education & Information Technologies* *Educational Research Review* *Educational Researcher* *Interactive Learning Environments* *Journal of Research on Technology in Education* *Journal of Science Education & Technology* *Journal of the Learning Sciences* *Science & Education* *TechTrends: Linking Research & Practice to Improve Learning*
ERIC	*Advances in Engineering Education* *Association for Educational Communications and Technology* *Cambridge Journal of Education* *Cognitive Science* *Curriculum Journal* *Design and Technology Education* *Education* *Education Leadership Review* *Education Sciences* *Educational Action Research* *Educational Leadership* *Educational Technology & Society* *Educational Technology Research and Development* *Informatics in Education*

	International Association for Development of the Information Society *International Journal of Computer Science Education in Schools* *Journal of College Science Teaching* *Journal of Computers in Mathematics and Science Teaching* *Journal of Education and Practice* *Journal of Engineering Education* *Journal of Science Education and Technology* *Journal of Staff Development* *Leadership and Policy in Schools* *Learning Professional* *Research in Science Education* *Research in Social Sciences and Technology* *School Community Journal* *Science and Children* *Science Teacher* *Teacher Development* *TechTrends: Linking Research & Practice to Improve Learning*
JSTOR	*American Educational Research Journal* *Cognition and Instruction* *Educational Technology & Society* *Educational Technology Research & Development* *Instructional Science* *Journal for Research in Mathematics Education* *Journal of College Science Teaching* *Journal of College Science Teaching* *Journal of Educational Technology & Society* *Journal of Science Education and Technology* *Journal of Science Teacher Education* *Review of Educational Research* *Science Scope* *Teacher Education Quarterly* *The Elementary School Journal* *The Journal of Educational Research* *The Journal of Technology Studies*
Professional Development Collection	*American Journal of Education* *American Journal of Evaluation* *British Journal of Educational Technology* *Cambridge Journal of Education* *Computer Science Education* *Curriculum Journal* *Early Childhood Education Journal* *Education* *Education Journal*

	Educational Research *Educational Research Evaluation* *Educational Research Quarterly* *Educational Technology Research & Development* *TechTrends: Linking Research & Practice to Improve Learning*
PsycINFO	*Cognitive Science* *Computational Intelligence* *Constructivist Educational Technology: Re-Examining the Foundations and State of the Literature* *Educational Research* *Educational Technology Research & Development* *International Journal of Computer-Supported Collaborative Learning* *International Journal of Human-Computer Interaction* *Journal of Educational Computing Research* *Journal of Research in Science Teaching* *Journal of Science Education and Technology* *Science* *Topics in Cognitive Science*

Appendix B: Literature Review Systematic Search & Findings

For each database:

- A Boolean Search was used.
- A thesaurus was used in each database to determine search terms for each specific database search.
- I read the title and abstract to determine if the article was relevant to the research question guiding the literature review.

Note: *The following research questions were later combined and revised and do not reflect the final guiding questions presented in Chapter Two. However, the tables below account for all the literature searched for this review.*

Research Question #1

Research Question	Evaluation Question(s) Addressed:
What is computational thinking?	#2 How does participating in the STEM/CT professional development program affect paraprofessionals' definition and understanding of computational thinking?

Search terms: computational thinking, define, definition, meaning, description

Inclusion/Exclusion Criteria:

Inclusion Criteria	*Exclusion Criteria*
Peer-reviewed or published books/chapters	
Full-text available	No full text available
Defines and explains computational thinking	Studies about developing, implementing, or designing computational thinking activities
Available in English	Not available in English

Searches:

Database	Search	# of Articles	# of relevant articles
Academic Search Ultimate	computational thinking AND (definition or define or meaning or description)	222	3 Bell & Lodi, 2019 Lee et al., 2020 Weintrop et al., 2016
Computer Source	computational thinking AND (definition or define or meaning or description)	15	3 Rich et al., 2020 Aho, 2012

			Yadav et al., 2017
Education Source	computational thinking AND (define or definition or meaning)	110	9 Tsarava et al., 2022 Lee et al., 2020 Grover & Pea, 2013 Lodi & Martini, 2021 Bell & Lodi, 2019 Weintrop et al., 2016 Shute et al., 2017 Wilkerson et al., 2020 Tekdal, 2021
ERIC	computational thinking AND (definition or define or meaning or description)	44	5 Cansu et al., 2019 Lee et al., 2020 Haseski et al., 2018 Weintrop et al., 2016 Taslibeyaz et al., 2020
Google Scholar	computational thinking AND (define or definition or meaning) *Went ten pages deep*	17,700	8 Selby, 2013 Denning, 2017 Shute et al., 2017 Yadav et al., 2014 Weintrop et al., 2016 Grover & Pea, 2013 Lee et al., 2011 Mannila et al., 2014
	Computational thinking *Went ten pages deep*	453,000	31 Grover & Pea, 2013 NRC, 2010 Shute et al., 2017 Selby, 2013 NRC, 2011 Mannila et al., 2014 Lee et al., 2011 Wing, 2011 Aho, 2012 Barr et al., 2011 Brennan & Resnick, 2012 Yadav et al., 2017 Kafai, 2016 Hsu, 2018 Hu, 2011 Weintrop et al., 2016 Tang et al., 2020

			Lee et al., 2014 Lye & Koh, 2014 Yadav et al., 2014 Kale et al., 2018 Barr & Stephenson, 2011 Cansu & Cansu, 2019 Grover & Pea, 2018 Perkovic et al., 2010 Lockwood & Mooney, 2017 Guzdial et al., 2019 Haseski et al., 2018 Hunsaker, 2020 Weinberg, 2013 Li et al., 2020
PsycInfo	computational thinking AND (define or definition or meaning)	5	2 Lee et al., 2020 Weintrop et al., 2016

Research Question #2

Research Question	Evaluation Question(s)
What is the role of a K-6 paraprofessional, and what is known about their professional development?	#1 What were the experiences of paraprofessionals participating in the STEM/CT professional development program? What additional activities and supports do paraprofessionals still need? #3 To what extent does participation in the STEM/CT professional development program provide paraprofessionals with the strategies and supports needed to create STEM/CT integrated lessons using the Utah K-5 CS standards? What additional strategies and supports do paraprofessionals still need?

Search terms: paraprofessional, paraeducator, teacher's assistant, teacher aides, assistants, school aide, school personnel, paraprofessional personnel, professional development, professional learning, professional education, elementary school or primary school or grade school

Inclusion/Exclusion Criteria:

Inclusion Criteria	*Exclusion Criteria*
United States	Outside of the U.S.
Published since 2000	Published prior to 2000
Peer Reviewed	Not peer-reviewed

Full-text available	Secondary or Higher Education
English	
Elementary school	

Searches:

Database	Search	# of Articles	# of relevant articles
Education Source	(professional development or professional learning or professional education) AND (paraprofessional or paraeducator or assistant or aide) AND (elementary school or primary school or grade school) NOT (special education or special needs or disabilities)	185	0
ERIC	(teacher aides or assistants) AND professional development AND (elementary school or primary school or grade school)	84	5 Manz et al., 2010 Anderson, et al., 2015 Lewis, 2004 Jolly & Evans, 2005 Hauerwas & Goessling, 2008
PsycINFO	(paraprofessional personnel) AND (professional development) AND (elementary school)	4	0
Professional Development Collection	(paraprofessionals or teachers' assistants or paraeducators or aides) AND (professional development) AND (elementary school or primary school or grade school) NOT (special education or special needs or disabilities)	31	0
Academic Search Ultimate	(professional development or professional education) AND (teacher assistant) NOT (special education or special needs or disabilities) AND (elementary)	7	0
ProQuest Dissertations & Theses Global	ab(paraprofessionals) AND ab(professional development)	57	2 Romano, 2015 Julian, 2020

Google Scholar	paraprofessionals AND professional development **Went ten pages deep* ** Only kept systematic reviews/seminal pieces that included special education*	14, 700	7 McKenzie, 2011 Jones et al., 2012 Brock & Carter, 2013 Brock & Anderson, 2021 Douglas et al., 2019 Christenson, 2013 Brown & Devecchi, 2013

Research Question #3

Research Question	Evaluation Question(s)
What is known about computational thinking professional development for K-6 paraprofessionals?	#2 How does participating in the STEM/CT professional development program affect paraprofessionals' definition and understanding of computational thinking? #3 To what extent does participation in the STEM/CT professional development program provide paraprofessionals with the strategies and supports needed to create STEM/CT integrated lessons using the Utah K-5 CS standards? What additional strategies and supports do paraprofessionals still need?

Search terms: paraprofessional, teacher's aide, aide, teacher's assistant, professional development, professional learning, K-6, elementary, computational thinking, computer science

Inclusion/Exclusion Criteria:

Inclusion Criteria	*Exclusion Criteria*
United States	Outside of the U.S.
Published since 2010	Published prior to 2010
Peer Reviewed	Not peer-reviewed
Full-text available	No full-text available
Published in English	Published in a language other than English
Elementary school	Secondary or Higher Education
Paraprofessionals	Teachers

Professional development/training	Not about professional development/training

Searches:

Database	**Search Terms**	**# of Articles**	**# of Relevant Articles**
ACM Digital Library	"paraprofessional" OR "teacher's assistant" OR "aide" AND "professional development" AND "computational thinking" "paraprofessional" OR "teacher's assistant" OR "aide" AND "professional development" OR “professional learning” AND "computer science" AND “elementary”	61 107	0 0
Education Source	“Teachers' assistants” AND “Professional education” AND “computational thinking” OR “computer science”	No results found	0
ERIC	(paraprofessional school personnel OR school aides OR Teacher aides+) AND professional development AND computational thinking	No results found	0
PsycInfo	(Teacher Aides) AND (professional development) AND (computational thinking OR computer science)	No results found	0
Computer Source	(paraprofessionals or teachers' assistants or paraeducators or aides) AND professional development AND computational thinking	No results found	0
Google Scholar	"paraprofessional" OR "teacher's assistant" OR "aide" AND "professional development" AND "computational thinking"	176	2 Smith, 2020; Prado et al., 2021

Research Question #4

Research Question	Evaluation Question(s)
What is known about computational thinking professional development for elementary educators?	#2 How does participating in the STEM/CT professional development program affect paraprofessionals' definition and understanding of computational thinking? #3 To what extent does participation in the STEM/CT professional development program provide paraprofessionals with the strategies and supports needed to create STEM/CT integrated lessons using the Utah K-5 CS standards? What additional strategies and supports do paraprofessionals still need?

Search terms: professional development, professional education, professional learning, K-6, elementary, elementary education, elementary education research, computational thinking, computer science

Inclusion/Exclusion Criteria:

Inclusion Criteria	*Exclusion Criteria*
Published since 2010	Published prior to 2010
Peer Reviewed	Not peer-reviewed
Full-text available	No full-text available
Published in English	Published in a language other than English
Elementary school	Secondary or Higher Education
Teachers/educators	Not teachers
Professional development/training	Not about professional development/training

Searches:

Database	Search	# of Articles	# of Relevant Articles
Academic Search Ultimate	professional education AND computational thinking AND elementary education research professional education AND computational thinking	2 32	1 Ketelhut et al., 2020 4 Ketelhut et al., 2020 Yadav et al., 2016 Lee et al., 2020 Gilbert et al., 2016

Education Source	Professional education AND computational thinking AND elementary education Professional education AND computational thinking OR computer science AND elementary education	4 10	1 Ketelhut et al., 2020 1 Rich et al., 2021
Computer Source	professional education AND computational thinking AND elementary education research professional education AND computational thinking	No results found No results found	0 0
ERIC	Professional development AND computational thinking AND elementary education	10	4 Ketelhut et al., 2020 Bower et al., 2017 Li et al., 2019 Yadav et al., 2016
PsycInfo	Professional development AND computational thinking OR computer science AND elementary education Professional development AND computational thinking OR computer science	2 25	1 Ketelhut et al., 2020 1 Ketelhut et al., 2020
ProQuest Dissertations & Theses	ab(professional development) AND ab(computational thinking)	22	1 Bain, 2021
JSTOR	(((professional development) AND (computational thinking)) AND (elementary education)) AND la:(eng OR en)	576 (Stopped on page 7 of results)	2 Angeli et al., 2016 Jocius et al., 2021
Professional Development Collection	Professional development AND computational thinking	4	1 Yadav et al., 2016
Google Scholar	Professional development AND computational thinking AND elementary education *Went 15 pages deep	16,300	16 Ketelhut et al., 2020 Hestness et al., 2018 Yadav et al., 2016 Jocius et al., 2020 Kong et al., 2020 Yadav et al., 2017 Sands et al., 2018

			Lamprou & Repenning, 2018 Mason & Rich, 2019 Bower et al., 2017 Yadav et al., 2019 Kotsopoulous et al., 2017 Hunsaker, 2020 Li et al., 2020 Lenoard et al., 2018 Meneskse, 2015

Appendix C: Pre-Survey Questions

Computational Thinking Survey

Notes:
Questions with a * signify that the question was from or modified from the 2020 Gallup/Code with Google Report.

Questions with ** signify that the question is from the TBaCCT.

Questions with a *** signify that the question is from the Teacher Computational Thinking Attitude Questionnaire.

1) Name

2) What is your gender?

3) What is your age?

- ○ 18-30
- ○ 31-40
- ○ 41-50
- ○ 51-65+

4) Explain any formal education you've obtained beyond high school.

5) I am a...

- ○ Licensed teacher
- ○ A teacher who is no longer licensed
- ○ Instructional assistant

6) Before becoming a STEM specialist, did you have any prior teaching experience? If so, explain.

7) When I am presented with a problem, I have difficulty breaking it down into smaller steps. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

8) I struggle to generalize solutions that can be applied to many different problems. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

9) I am NOT good at solving puzzles. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree

◯ Disagree

◯ Strongly disagree

10) I struggle to identify where and how to use variables in the solution of a problem. **

◯ Strongly agree

◯ Agree

◯ Somewhat agree

◯ Somewhat disagree

◯ Disagree

◯ Strongly disagree

11) Computational thinking is understanding how computers work. ***

◯ Strongly agree

◯ Agree

◯ Somewhat agree

◯ Somewhat disagree

◯ Disagree

◯ Strongly disagree

12) Computational thinking involves thinking logically to solve problems. ***

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

13) Computational thinking involves using computers to solve problems. ***

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

14) Computational thinking involves abstracting general principles and applying them to other situations. ***

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

15) I can explain basic computing concepts to children (e.g., algorithms, loops, conditionals, functions, variables, debugging, pattern-finding). **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

16) I can recognize and appreciate computing concepts in all subject areas. **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

17) I can create computing activities at the appropriate level for my students. **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

18) I can explain computing concepts well enough to be effective in teaching computing. **

- Strongly agree
- Agree
- Somewhat agree
- Somewhat disagree
- Disagree
- Strongly disagree

19) I can explain how computing concepts are connected to daily life. **

- Strongly agree
- Agree
- Somewhat agree
- Somewhat disagree
- Disagree
- Strongly disagree

20) I can develop and plan effective computing lessons. **

- Strongly agree
- Agree
- Somewhat agree
- Somewhat disagree
- Disagree
- Strongly disagree

21) Computational thinking can be incorporated into the classroom using computers in the lesson plan. ***

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

22) Computational thinking can be incorporated into the classroom by allowing students to problem-solve. ***

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

23) Computing should be taught in elementary school. **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

24) Learning about computing can help elementary students become more engaged in school. **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

25) Computing content and principles CAN be understood by elementary school children. **

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

26) My current teaching situation does NOT lend itself to teaching computing concepts to my students. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

27) Computing is an important 21st-century literacy. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

28) Computational thinking is an important part of today's science standards. **

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

29) Do you think offering opportunities to learn computer science is more important, just as important, or less important to a student's future success than other required courses like math, science, social studies/history, and English? *

◯ More important

◯ Just as important

◯ Less important

30) How important is it for your students to learn computer science? *

◯ Very important

◯ Important

◯ Not important

31) How important is it for YOU to learn computer science? *

◯ Very important

◯ Important

◯ Not important

32) How interested are you in learning computational thinking and computer science skills? *

◯ Very interested

◯ Interested

◯ Somewhat interested

◯ Not at all interested

Appendix D: Post-Survey Questions

Same as pre–survey questions (minus demographic information) with the addition of a few questions at the end of the survey asking about their experience with the STEM/CT program (PD).

Additional questions:

33) I had a positive learning experience in the summer professional development.

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

34) I had a positive learning experience in the monthly professional development sessions.

- ○ Strongly agree
- ○ Agree
- ○ Somewhat agree
- ○ Somewhat disagree
- ○ Disagree
- ○ Strongly disagree

35) I was given ample opportunity to practice the computational thinking skills I was asked to learn and teach.

- ◯ Strongly agree
- ◯ Agree
- ◯ Somewhat agree
- ◯ Somewhat disagree
- ◯ Disagree
- ◯ Strongly disagree

Appendix E: STEM/CT Professional Development Program Interview Protocol

Note: MSD calls the STEM/CT program the "Elementary STEM PLC." This term is used in the interview protocol so the paraprofessionals understand what I am asking questions about.

Introduction

1. Tell me about how you came to be involved with the STEM specialty as an instructional assistant.
2. Did you have any experience with computer science or computational thinking before becoming a STEM specialty teacher? Tell me about it…

Defining & Recognizing Computational Thinking

3. What is computational thinking? How would you explain what computational thinking is to a colleague who has never heard of it?
4. Can you give me an example of what computational thinking would look like in a classroom? How would you know your students were engaging in computational thinking?
5. Has your understanding of CT changed as a result of participating in this project? Why or why not?

Importance, Values & Beliefs about Computational Thinking

6. Do you think it is important for students to engage in computational thinking? Why or why not?
7. Do you think it is important for your students to learn computational thinking? Why or why not? Have your ideas about this changed throughout the school year?

STEM/CT Integrated Lessons & Lesson Planning

8. Do you have any prior experience with creating lesson plans as an instructional assistant or elementary school teacher? If yes, explain.
9. What aspect of the monthly Elementary STEM PLC was most beneficial in helping you learn how to design STEM/CT integrated lesson plans using the CT skills and approaches?
10. Can you tell me a story about working with one of the STEM tools and technologies (e.g., Ozobot, Makey Makey, etc.)?
11. Tell me about a STEM/CT integrated lesson plan you designed.
    a. Why did you choose (CT skill) and (CT approach) for this lesson?
    b. How is this activity connected to (*CT skill selected)?*
    c. How is this activity connected to *(the CT approach selected)?*
    d. What was the implementation of that lesson plan like?
12. What additional strategies and supports do you still need to successfully design STEM/CT integrated lesson plans using the CT skills and approaches?

Elementary STEM PLC (STEM/CT Program)

13. Tell me about your experiences in the Elementary STEM PLC. What was the most memorable part of the STEM PLC program for you? What aspect was the most challenging? Where do you feel like you still need support?
14. Is there anything you would change about your experience with the Elementary STEM PLC (STEM/CT program)?

15. What advice do you have for other educators who might participate in the Elementary STEM PLC in the future?
16. Do you plan on continuing as a STEM instructional assistant? Why or why not?

Conclusion

17. Is there anything else you'd like to share with me?

Appendix F: Participant Survey Responses Organized by Construct

CT Understanding: Using a Computer				
	Item 4: *Computational thinking is understanding how computers work*		***Item 5:*** *Computational thinking involves using computers to solve problems.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	strongly agree	strongly agree	strongly agree	strongly agree
Ellen	agree	agree	somewhat agree	somewhat agree
Liz	strongly agree	disagree	strongly agree	somewhat disagree
Ana	agree	agree	strongly agree	agree
Charlotte	somewhat disagree	somewhat agree	somewhat agree	strongly agree
Sophia	agree	strongly agree	agree	strongly agree
Hailey	somewhat agree	agree	somewhat agree	agree
Marissa	strongly agree	disagree	agree	disagree
CT Understanding: Problem Solving				
	Item 2: *Computational thinking involves thinking logically to solve problems.*		***Item 14:*** *Computational thinking involves abstracting general principles and applying them to other situations.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	agree	strongly agree	agree	strongly agree
Ellen	agree	strongly agree	agree	agree
Liz	strongly agree	agree	agree	strongly agree
Ana	strongly agree	strongly agree	agree	somewhat agree
Charlotte	agree	strongly agree	agree	strongly agree
Sophia	agree	strongly agree	agree	strongly agree
Hailey	strongly agree	strongly agree	agree	strongly agree
Marissa	agree	somewhat agree	agree	somewhat agree
CT Ability				
	Item 8: *When I am presented with a problem, I have difficulty breaking it down into smaller steps*		***Item 13:*** *I am NOT good at solving puzzles.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	disagree	somewhat disagree	disagree	disagree
Ellen	somewhat agree	agree	disagree	disagree
Liz	disagree	agree	disagree	somewhat disagree
Ana	somewhat disagree	agree	somewhat disagree	strongly disagree
Charlotte	agree	agree	somewhat disagree	disagree
Sophia	agree	agree	disagree	strongly disagree
Hailey	strongly agree	agree	disagree	strongly agree
Marissa	strongly disagree	agree	disagree	somewhat disagree
	Item 26: *I struggle to generalize solutions that can be applied to many different problems*		***Item 28:*** *I struggle to identify where and how to use variables in the solution of a problem.*	
Participant	Pre	Post	Pre	Post
Gina	disagree	somewhat disagree	disagree	somewhat agree
Ellen	somewhat disagree	somewhat disagree	disagree	somewhat agree
Liz	somewhat disagree	agree	agree	agree
Ana	agree	agree	disagree	agree

Charlotte	somewhat agree	somewhat disagree	somewhat disagree	somewhat agree
Sophia	agree	agree	somewhat agree	strongly agree
Hailey	agree	agree	agree	somewhat agree
Marissa	agree	strongly agree	agree	agree
Teaching Computer Science/Computational Thinking				
	Item 1: *I can develop and plan effective computational thinking lessons.*		***Item 6:*** *Computational thinking can be incorporated into the classroom by allowing students to problem-solve.*	
Participant	Pre	Post	Pre	Post
Gina	somewhat agree	strongly agree	strongly agree	strongly agree
Ellen	somewhat agree	agree	agree	strongly agree
Liz	strongly agree	agree	strongly agree	strongly agree
Ana	strongly agree	agree	agree	strongly agree
Charlotte	agree	somewhat agree	agree	strongly agree
Sophia	somewhat agree	strongly agree	strongly agree	strongly agree
Hailey	agree	strongly agree	agree	strongly agree
Marissa	agree	agree	agree	agree
	Item 11: *I can explain how computational thinking concepts are connected to daily life*		***Item 17:*** *I can explain computational thinking well enough to be effective in teaching computational thinking.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	agree	agree	somewhat agree	agree
Ellen	disagree	agree	somewhat agree	agree
Liz	agree	agree	somewhat agree	agree
Ana	somewhat agree	agree	agree	strongly agree
Charlotte	disagree	agree	agree	somewhat agree
Sophia	agree	strongly agree	somewhat disagree	strongly agree
Hailey	agree	strongly agree	agree	strongly agree
Marissa	agree	agree	agree	agree
	Item 18: *I can recognize and appreciate computational thinking in all subject areas.*		***Item 22:*** *Computational thinking can be incorporated in the classroom by using computers in the lesson plan.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	somewhat agree	strongly agree	agree	strongly agree
Ellen	disagree	agree	somewhat agree	agree
Liz	agree	agree	agree	agree
Ana	somewhat agree	agree	strongly agree	agree
Charlotte	agree	somewhat agree	agree	agree
Sophia	agree	strongly agree	agree	strongly agree
Hailey	somewhat agree	strongly agree	somewhat agree	somewhat agree
Marissa	agree	strongly agree	agree	disagree
	Item 24: *I can create computational thinking activities at the appropriate level for my students.*		***Item 25:*** *I can explain basic computational thinking concepts to children (e.g., algorithms, loops, conditionals, functions, variables, debugging, pattern-finding).*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	somewhat disagree	agree	disagree	agree
Ellen	disagree	agree	disagree	agree
Liz	agree	agree	agree	agree
Ana	strongly agree	agree	agree	agree
Charlotte	disagree	somewhat agree	disagree	somewhat agree
Sophia	somewhat agree	strongly agree	somewhat disagree	strongly agree

Hailey	agree	strongly agree	agree	strongly agree
Marissa	agree	strongly agree	agree	agree
Attitudes, Beliefs, and Values about Computational Thinking				
	***Item 3**: Computational thinking is an important 21st-century literacy*		***Item 7**: It is important for my students to learn about computational thinking.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	strongly agree	strongly agree	strongly agree	strongly agree
Ellen	agree	strongly agree	agree	strongly agree
Liz	strongly agree	strongly agree	agree	strongly agree
Ana	agree	somewhat agree	agree	strongly agree
Charlotte	agree	strongly agree	agree	strongly agree
Sophia	agree	strongly agree	strongly agree	strongly agree
Hailey	agree	strongly agree	agree	strongly agree
Marissa	strongly agree	agree	strongly agree	strongly agree
	***Item 9**: Learning about computational thinking can help elementary students become more engaged in school.*		***Item 10**: Offering opportunities to learn computational thinking is less important to a student's future success than other required courses like math, science, social studies/history, and English.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	agree	strongly agree	disagree	disagree
Ellen	agree	agree	disagree	disagree
Liz	agree	agree	somewhat disagree	disagree
Ana	somewhat agree	somewhat agree	somewhat agree	somewhat disagree
Charlotte	somewhat agree	strongly agree	disagree	strongly disagree
Sophia	strongly agree	strongly agree	disagree	disagree
Hailey	agree	strongly agree	disagree	strongly agree
Marissa	strongly agree	agree	somewhat disagree	disagree
	***Item 12**: Computational thinking is an important part of today's science standards*		***Item 15**: Offering opportunities to learn computational thinking is just as important to a student's future success as other required courses like math, science, social studies/history, and English.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	strongly agree	strongly agree	strongly agree	strongly agree
Ellen	agree	agree	agree	strongly agree
Liz	agree	strongly agree	somewhat agree	agree
Ana	agree	strongly agree	agree	agree
Charlotte	somewhat agree	strongly agree	agree	strongly agree
Sophia	agree	strongly agree	strongly agree	strongly agree
Hailey	strongly agree	strongly agree	agree	strongly agree
Marissa	agree	strongly agree	agree	strongly agree
	***Item 16**: Computational thinking content and principles CAN be understood by elementary school children.*		***Item 19**: It is important for me to learn about computational thinking.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	strongly agree	strongly agree	agree	strongly agree
Ellen	agree	strongly agree	agree	strongly agree
Liz	agree	strongly agree	agree	agree
Ana	somewhat agree	agree	strongly agree	agree
Charlotte	agree	strongly agree	agree	strongly agree
Sophia	agree	strongly agree	agree	strongly agree
Hailey	somewhat agree	strongly agree	agree	strongly agree

Marissa	agree	agree	strongly agree	strongly agree
	***Item 20 RC**: My current teaching situation lends itself to teaching computational thinking concepts to my students*		***Item 21**: Offering opportunities to learn computational thinking is more important to a student's future success than other required courses like math, science, social studies/history, and English.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	somewhat disagree	agree	somewhat agree	somewhat agree
Ellen	somewhat disagree	agree	agree	somewhat disagree
Liz	somewhat agree	agree	agree	somewhat agree
Ana	agree	agree	strongly agree	somewhat agree
Charlotte	somewhat disagree	agree	agree	strongly agree
Sophia	strongly agree	agree	agree	agree
Hailey	agree	strongly agree	agree	strongly agree
Marissa	agree	agree	agree	agree
	***Item 23**: Computational thinking should be taught in elementary school*		***Item 27**: I am interested in learning computational thinking skills.*	
Participant	**Pre**	**Post**	**Pre**	**Post**
Gina	somewhat agree	strongly agree	strongly agree	strongly agree
Ellen	agree	agree	strongly agree	agree
Liz	agree	strongly agree	agree	agree
Ana	strongly agree	agree	strongly agree	agree
Charlotte	agree	strongly agree	agree	strongly agree
Sophia	agree	strongly agree	strongly agree	strongly agree
Hailey	agree	strongly agree	agree	strongly agree
Marissa	agree	agree	agree	agree
Professional Development Experience				
	***Item 29:** I had a positive experience in the Elementary STEM PLC Summer sessions*	***Item 30:** I had a positive experience in the monthly Elementary STEM PLC sessions.*	***Item 31:** I was given ample opportunity to practice the computational thinking skills I was asked to learn and teach in the Elementary STEM PLC.*	
Participant	Post	Post	Post	
Gina	strongly agree	strongly agree	somewhat agree	
Ellen	agree	agree	agree	
Liz	strongly agree	agree	strongly agree	
Ana	strongly agree	strongly agree	agree	
Charlotte	strongly agree	strongly agree	disagree	
Sophia	strongly agree	strongly agree	agree	
Hailey	strongly agree	strongly agree	strongly agree	
Marissa	strongly agree	strongly agree	agree	

Curriculum Vitae

Aubrey A. Rogowski

18020 Boardtree Drive
Elgin, Texas 78621

aubrey.rogowski@gmail.com

Education

Ph.D. Instructional Technology & Learning Sciences, Utah State University, 2024

Dissertation: *Evaluating an Integrated Science, Technology, Engineering, Math, and Computational Thinking Professional Development Program for Elementary Level Paraprofessional Educators*

Chair: Dr. Mimi Recker

M.Ed. in Instructional Technology & Learning Sciences, Utah State University, 2018

B.S. Elementary Education (1-8), Southern Utah University, 2014
B.S. Mild/Moderate Special Education (PK-12) Southern Utah University, 2014

Licenses & Certificates

Professional Educator License (expires 06/30/2027)
Elementary Education
Special Education K-12
Utah State Board of Education

Texas Educator Certificate (expires 5/31/2029)
Core Subjects with STR Grades (EC-6)
Special Education Grades (EC-12)

Master Teacher Certificate
Utah State University
Office of Empowering Teaching Excellence

Teaching Scholar Certificate
Utah State University
Office of Empowering Teaching Excellence

Research Experience

Graduate Research Assistant | USU, Instructional Technology & Learning Sciences
August 2022- Current
Integrating Elementary-Level Mathematics Curricula with Expansively-Framed Computer Science Instruction (NSF)
Collaboratively designing adapted and cross-curricular units for 5th-grade math teachers and computer lab paraprofessionals to teach math and CS concepts. Collect and analyze qualitative and quantitative data. Collaborate with district partners, researchers, and other graduate research assistants; use qualitative research methods; observe teachers' and paraprofessionals' teaching, etc.

Researcher | Inter-institutional Collaborative Research Group
January 2022- present
Scoping Review: Adult Learning in Distance Education
A collaborative inter-institutional research group made up of graduate students, university instructors, and professors working together to conduct a scoping review on adult learning and distance education.

Graduate Research Assistant | USU, Instructional Technology & Learning Sciences
May 2021-August 2021; December 2021-January 2022; May 2022- August 2022
Indian Education and Computing for All: A Montana Story (NSF)
Developed an integrated, curricular units for 4th-8th grade teachers to teach the Indian Education for All standards, Language Arts, Science, Social Studies, and Computer Science standards.

Graduate Research Assistant | USU, Instructional Technology & Learning Sciences
September 2017-August 2019
Situated Librarianship Learning Infrastructure for Making (IMLS)
Assist with the creation, design, and development of STEM-oriented maker program training materials for rural librarians to use in running Makerspace programs in their libraries; Collaborate with other researchers, librarians, and graduate research assistants; use qualitative research methods; Observe librarians and youth interacting in library spaces to guide material design; Collaboratively write conference proposals, journal articles, book chapters, etc.

Publications

Journal Articles

Searle, K. A., **Rogowski, A.,** Tofel-Grehl, C., & Jiang, M. (2023). A Critical Computing Curriculum Design Case: Exploring Tribal Sovereignty for Middle School Students. *Journal of Computer Science Integration*, 6(1), 8.DOI: https://doi.org/10.26716/jcsi.2023.12.27.45

Lee, V. R., **Rogowski**, **A.**, Shehzad, U., & Recker, M. (2021). Unplugged-to-plugged computer science at the library. *Teacher Librarian. 48(3),* 34-39.

Rogowski, A., Recker, M., & Lee, V. R. (2018). Designing online support guides for librarians managing STEM maker activities. *International Journal on Innovations in Online Education, 2*(4). https://doi.org/10.1615/IntJInnovOnlineEdu.2019029566

Book Chapters

Rogowski, A. (in press). TikTok flop. In R. E. West (Ed), *Failing Forward*.

Vincent, H., Lee, V. R., **Rogowski, A**., & Recker, M. (2022). Looming code: A model, learning activity, and professional development approach for CS educators. In C. Mouza, A. Ottenbreit-Leftwich, & A. Yadav (Eds.), *Professional development for in-service teachers: Research and practices in computing education.* Information Age Publishing.

Rogowski, A., Lee, V. R., & Recker, M. (2020). In support of making in libraries rather than makerspaces: Rethinking the (maker)space for rural libraries. In M. Melo & J. Nichols (Eds.), *Re-making the library makerspace: Critical theories, reflections, and practices* (pp. 167-182). Litwin Books.

Lee, V. R., Recker, M., & **Rogowski, A**. (2020). Researchers or service providers? A case of renegotiating expectations in a research-practice partnership. In T. Ruecker & V. Svihla (Eds.), *Navigating challenges in qualitative educational research: Research, interrupted* (pp. 80-94). Routledge.

Book Reviews

Cuthbert, J. M., **Rogowski, A.**, Vakula, M., Aguilar, J., & Kesler, K. (2021). The missing course: An introduction to college teaching for graduate instructors. *Journal on Empowering Teaching Excellence, 5*(2). https://doi.org/10.26077/866d-0cdd

Conference Proceedings

Chipps, J., Dangol, A., Fasy, B. T., Hancock, S., Jiang, M., **Rogowski, A.,** Searle, K. A., & Tofel-Grehl, C. (2022). *Culturally responsive storytelling across content areas using American Indian ledger art and physical computing*. American Society for Engineering Education Annual Conference (Paper ID # 38431). Minneapolis, MN. American Society of Engineering Education.

Conferences

Rogowski, A., & Recker, M. (2023, October). *Findings and future steps from a STEM/computational thinking professional development program evaluation for paraprofessionals*. Concurrent session at the 2023 Association for Educational Communications and Technology (AECT) International Convention, Orlando, FL.

Rogowski, A., Bellnier, K., & Fensie, A. (2023, October). *Assessing Learning & Measurement for Online Adult Learners.* Concurrent session at the 2023 Association for Educational Communications and Technology (AECT) International Convention, Orlando, FL.

Rogowski, A., & Recker, M. (2023, May). *Towards equity in CS education: The various ways K-6 paraprofessionals conceptualize computational thinking*. Paper session at the 2023 American Educational Research Association (AERA) Annual Meeting (virtual).

Searle, K. S., **Rogowski, A**., Tofel-Grehl, C., & Jiang, M. (2023, April). *A critical curriculum design case: Indigenous schooling past, present, and future*. Presidential session at the 2023 American Educational Research Association (AERA) Annual Meeting, Chicago, IL.

Searle, K.S., Tofel-Grehl, C., **Rogowski, A**., & Jiang, M. (2023, April). *Situating computing in critical and culturally responsive context: Bringing together Indian education for all and computer science in Montana.* Poster session at the 2023 American Educational Research Association (AERA) Annual Meeting, Chicago, IL.

Chipps, J., Dangol, A., Fasy, B. T., Hancock, S., Jiang, M., **Rogowski, A.,** Searle, K. A., & Tofel-Grehl, C. (2022). *Culturally responsive storytelling across content areas using American Indian ledger art and physical computing*. Concurrent session at the American Society for Engineering Education Annual Conference, Minneapolis, MN.

Fensie, A. **Rogowski, A**., Bellnier, K, Stidham, S., Jones, M., Wiley, L., St. Pierre, T., Alicea, M. (2022, October). *The study of adult learners in distance education: A scoping review of the literature*. Concurrent session at the 2022 Association for Educational Communications and Technology (AECT) International Convention, Las Vegas, NV.

Rogowski, A. (2022, October). *Student-to-Student Online Course Interaction with FlipGrid* Impact practice session at the 2022 Association for Educational Communications and Technology (AECT) International Convention, Las Vegas, NV.

Rogowski, A. & Recker, M. (2022, October). *A STEM/CT integrated lesson plan rubric.* Poster presentation at the 2022 Association for Educational Communications and Technology (AECT) International Convention, Las Vegas, NV.

Rogowski, A. Tulane, S., Delgadillo, L., Steven, J., Romney, A., Mickelson, P., Sermersheim, L. (2022, August). *Reimagining the Online Classroom: Six Course Activities to Foster a Transformative Learning Community*. Concurrent session at the 2022 Empowering Teaching Excellence (ETE) Conference, Logan, UT.

Fensie, A., **Rogowski, A**., O'Mahony, A. (2022, July). *A scoping review of the literature on adult learning processes in distance education.* Poster presentation at the 2022 International Mind, Brain, and Education Society (IMBES) Conference, Montreal, QC, Canada.

Fensie, A., St. Pierre, T., Krieger, M., Jones, M., Alicea, M., Wher, K., **Rogowski, A**., Bellnier, K., Stidham, S., Gill, P., Wiley, L., O'Mahony, A., & Allan, E. (2022, April). *The study of adult learners in distance education: A scoping review of the literature.* Poster presentation at the UMaine Student Symposium, Orono, ME.

Rogowski, A. (2022, April). *Computational thinking and professional learning for* paraprofessionals. Presentation at the 2022 Student Research Symposium, Logan, UT.

Rogowski, A. (2022, April). *Computational thinking professional development for K-6 educators*. Presentation at the 2022 Annual Graduate Conference on Learning and Technology, Logan, UT.

Rogowski, A. & Recker, M. (2021, November). *Forgotten: Where is the research on paraprofessional learning?* Roundtable session at the 2021 Association for Educational Communications and Technology (AECT) International Convention, Chicago, IL.

Vincent, H., **Rogowski, A**., Lee, V.R., Recker, M. (2021, April). *Learning to facilitate computer programming activities in library settings through expansively-framed unplugged computing.* Roundtable session at the 2021 American Educational Research Association (AERA) Annual Meeting (virtual).

Lee, V. R., Recker, M., & **Rogowski, A.** (2021, April). *Design needs and solutions for supporting maker programs in rural libraries*. Symposium at the 2021 American Educational Research Association (AERA) Annual Meeting (virtual).

Rogowski, A., Recker, M. M., Lee, V. R. & Searle, K. A. (2020, April). *A framework characterizing how school librarians facilitate informal making activities.* Roundtable session at the 2020 American Educational Research Association (AERA) Annual Meeting, San Francisco, CA. http://tinyurl.com/r24895v (Conference canceled due to COVID-19 pandemic).

Rogowski, A., Recker, M., Lee, V. R., & Searle, K. (2020, April). *Librarians as informal learning designers: A framework for facilitating STEM-rich making in the school library.* Poster presentation at the Student Research Symposium, Logan, UT. https://digitalcommons.usu.edu/researchweek/ResearchWeek2020/All2020/43/

Lee, V. R., Recker, M., & **Rogowski, A.** (2019, July). *The library as a venue for making and learning.* Paper presentation at the American Association of Physics Teachers, Provo, UT.

Rogowski, A., Lee, V. R., & Recker, M. (2019, May). *Meaningful making: Bridging STEM & teen programming.* Presentation at the 2019 Utah Library Association Annual Conference, Sandy, UT.

Rogowski, A. (2019, March). *Design capacity for STEM-oriented making in libraries.* Poster presentation at the 2019 Learning, Instruction, and Technology (LIT) Conference, Salt Lake City, UT.

Rogowski, A., Lee, V., & Recker, M. (2019, March). *STEM in the library: Tips & tricks to get started.* Conference session at the 2019 Utah Educational Library Media Association (UELMA) Annual Conference, West Valley City, UT.

Rogowski, A., Phillips, A., Recker, M., Lee, V. R. (2019, April). *Design capacity for informal learning within schools: An analysis of school librarians and maker activities.* Paper presentation at the 2019 American Educational Research Association (AERA) Annual Meeting, Toronto, ON, Canada.

Rogowski, A., Phillips, A., Recker, M., & Lee, V. R. (2019, April). *Design capacity for informal learning within schools: An analysis of school librarians and maker activities.* Poster presentation at the 2019 Utah Coalition for Educational Technology Conference, Provo, UT.

Lee, V. R., **Rogowski, A**., Phillips, A. Recker, M. (2018, October). *Using a "light touch" to support middle school librarians with implementing STEM-oriented Maker activities.* Paper presentation at the 2018 Association for Educational Communications and Technology (AECT) International Convention, Kansas City, MO.

Rogowski, A., Phillips, A., Lee, V., & Recker, M. (2018, May*). STEM-geared Making for teens in the library*. Paper presentation at the 2018 Utah Library Association (ULA) Annual Conference, Provo, UT.

Phillips, A., **Rogowski, A**., Lee, V.R., & Recker, M. (2018, March). *STEM-geared Making for the busy librarian.* Presentation at the 2018 Utah Educational Library Media Association (UELMA) Annual Conference. West Valley City, UT.

Lee, V., Phillips, A., & Recker, M., & **Rogowski, A.** (2018, August). *Educative maker* *activity materials for small town librarians to support connected learning.* Paper presentation at the 2018 Connected Learning Summit, MIT Media Lab, Cambridge, MA.

Lee, V. R., Recker, M., Phillips A., & **Rogowski, A**. (2018, April). *An asset-based framework for youth maker development in libraries*. Structured poster session at the 2018 American Education Research Association (AERA) Annual Meeting, New York, NY.

Mendenhall, K., **Guyant, A.** (2013). *The effectiveness of Imagine Learning literacy program with English language learners.* Oral presentation at the 2013 Southern Utah Education Conference (SUECON) Annual Conference. St. George, UT.

Invited Guest Speaker

Rogowski, A., & Tulane, S. (2022, May). *Orienting students toward success*. eLearnX 2022. Logan, UT.

Lee, V. R., & **Rogowski, A.** (2019, February). *STEM-rich making in the library*. Webinar for Info2Go! Idaho Commission for Libraries.

Professional Experience

Design Lab Manager | USU, Instructional Technology & Learning Sciences
March 2018- May 2019
Maintain makerspace, seek user feedback, improvements to makerspace, new technologies, and programming for students to participate in the space

Evaluator | USU, School of Teacher Education & Leadership
December 2017- May 2018
Evaluate course rubric, determine inter-rater reliability of rubric, research and suggest rubric changes and improvements.

UX Research Intern | DigiCert
Fall 2018
Conduct interviews, qualitative data analysis, develop, program, and distribute surveys, conduct focus groups, quantitative data analysis, and provide consultations for academic research proposal assessments.

Teaching Experience

Graduate Teaching Fellow | USU, Instructional Technology & Learning Sciences
For Dr. Mimi Recker

August 2023- December 2023

Graduate Course

ITLS 6760- Grant Writing
(Format: online) (Semester: F2023)
Topics Include: Finding RFPs, creating a compliance matrix, proposal writing

Graduate Instructor | USU, Instructional Technology & Learning Sciences
November 2019- August 2022

Graduate Courses

ITLS 6390- Transformative Uses of Technology for Learning
(Format: online) (Semester: Su2020, Su2022)
Topics Include: educational reform, contemporary technologies, constructionism, digital fabrication and making, role of technology in transforming education

ITLS 6520- Instructional Design I
(Format: online) (Semester: Sp2020)
Topics Include: instructional design processes, instructional design models, needs analysis, design solutions, evaluation

Undergraduate Courses

ITLS 5500- Technology Integration & Innovation in Education
(Format: online) (Semesters: F2020, Sp2020, F2021, Sp2021, F2022, Sp2022)
Topics Include: 21st century education, ISTE standards, innovation in education, digital citizenship, technology integration models, lesson planning, student data and privacy laws, and various technology skills (screencasts, podcasts, multimedia PDFs, cloud-based tools, websites, curation tools, LMS)

Elementary Teacher | Eaglecrest Elementary, Alpine School District
5th grade- 2014-2015, 2015- 2016
6th grade- 2016-2017

Student Teacher | Eaglecrest Elementary, Alpine School District
January 2014- May 2014
5th grade

Student Teacher | Lehi Elementary, Alpine School District
January 2014- May 2014
K-2 mild/moderate special education
3-6 mild/moderate special education

Honors & Awards

Legacy of Utah State Award, 2023
Department of Instructional Technology and Learning Sciences at Utah State University

Travel Award, 2022
National Science Foundation ($2000)

Division C Graduate Student Seminar, 2022
American Education Research Association, San Diego, CA ($275)

Graduate Teacher of the Year, 2022
Utah State University

Graduate Teacher of the Year, 2022
Emma Eccles Jones College of Education and Human Services at Utah State University ($300)

Graduate Teacher of the Year, 2022
Department of Instructional Technology and Learning Sciences at Utah State University

Graduate Teacher of the Year, 2021
Department of Instructional Technology and Learning Sciences at Utah State University

Research & Development Scholarship, 2021
Department of Instructional Technology and Learning Sciences at Utah State University, ($845)

CEHS Byron R. & Shirley Burnham Scholarship, 2018
Department of Instructional Technology and Learning Sciences at Utah State University, ($1000)

Outstanding Graduate Student, 2018
Department of Instructional Technology and Learning Sciences at Utah State University

Elaine C. Southwick Award*, 2014
Southern Utah University
* Most prestigious and distinguished award a female can be awarded at SUU)

Service

Sub-Committee Member, 2022-2023
eLearnX/Seminars/Learning Circles Subcommittee
Empowering Teaching Excellence, Utah State University

Vice President- Online, 2022-2023
Instructional Technology Student Association (ITSA)

Learning Circle Facilitator, 2022
Empowering Teaching Excellence, Utah State University

Reviewer
Journal of Computer Science Integration
Journal on Empowering Teaching Excellence: Open Book Series (2021)
Journal on Empowering Teaching Excellence

Conference Proposal Reviewer
AECT 2022, AECT 2023

Student Council Advisor, 2016-1017
Eaglecrest Elementary

Professional Memberships

Association for Educational Communications and Technology (AECT) 2019-present
American Educational Research Association (AERA) 2019- present
International Society for the Learning Sciences (ISLS) 2022

Public School Outreach

Wellsville Elementary STEM Fest | February 2019
Birch Creek Elementary STEM Fest | March 2019
Nibley Elementary STEM Fest | April 2019

Skills

Data Collection
Qualtrics, Survey Monkey, Google Forms
Data Analysis
MAXQDA, R, Excel
Programming
Scratch, basic HTML
Learning Management Systems
Canvas, Google Classroom

Content Management Systems

WordPress, Weebly, Google Sites, Confluence

Online Collaboration

Zoom, Trello, Slack, Jira, Google Meet, JamBoard, Box, Dropbox

Computer Applications for Teaching & Learning

Google Suite, Microsoft Suite, Photoshop, iMovie, Storyboard, Audacity, Ocenaudio, LucidChart, LucidPress, Loom, Canva, FlipGrid, Padlet, Powtoon

Other

Spanish