Title

Not Everyone Has Access:� How Elementary Teachers’ Computer Science Goals and Strategies Relate to Equity

Author

Toohey, Melissa

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Not Everyone Has Access:
How Elementary Teachers' Computer Science Goals and Strategies Relate to Equity

A dissertation submitted in partial satisfaction of the

requirements for the degree Doctor of Education

by

Melissa Toohey

2022

ABSTRACT OF THE DISSERTATION

Not Everyone Has Access:

How Elementary Teachers' Computer Science Goals and Strategies Relate to Equity

by

Melissa Toohey

Doctor of Education

University of California, Los Angeles, 2022

Professor William Sandoval, Chair

The purpose of this study was to explore how elementary school teachers think about equity in relation to their goals in their computer science instruction, and to explore the strategies they articulated to make their practice more equitable. States are mandating computer science standards and instruction and requiring schools to teach the content (2020 State of CS). Parents want students to be taught the content, and students in affluent and predominantly white communities have greater access to computer science opportunities than their low-SES and minority counterparts (Margolis, 2020). In addition, teachers are not formally trained or prepared to teach computer science (2020 State of CS). This study adds to the limited body of research that focuses on computer science instruction at the elementary school level.

The current study used documents and interviews of 20 respondents. These respondents self-identified as elementary school teachers that teach computer science in some capacity. This sample of 20 teachers included teachers across the United States, as they self-selected to participate in the study through snowball sampling. The focus of this study was to explore computer science education at the elementary school level. The study explored the two questions:

1. How do elementary CS teachers think about equity in relation to their goals for their computer science instruction?

2. What strategies do elementary computer science teachers articulate to make their practice equitable?

This study found that elementary computer science teachers think about equity in two ways: opportunity and differentiation, and listed their goals as: providing access to opportunity, promoting positive identification with computer science, empowering students to use computer science for social change, as well as increasing academic achievement, and provided strategies for making their practices more equitable. This study added to the growing body of research on the topic of elementary computer science education, and supported and extended extant research. From these conclusions, recommendations for current implementation and practice and future research were shared. These recommendations included providing teachers training in integrating ideas around who holds the power in technology, expanding teachers' views on what constitutes equity in computer science education, and supporting teachers in leveraging computer science instruction to promote academic achievement. Teachers in this study expressed their interest in supporting students to acquire computer science skills and create equitable computer science educational pathways.

The dissertation of Melissa Toohey is approved.

Jean Ryoo

Kimberly Gomez

Ananda Marin

Kristen Rohanna

William Sandoval, Committee Chair

University of California, Los Angeles

2022

DEDICATION PAGE

When I was in 8th grade, my English teacher told me I “did not have what it takes” to be in high school honors English. The department head confirmed this by giving me an exam designed to exclude students and amplify educational inequities. In my first teaching role, I was told, again, I did not have what it takes to be a lead teacher. As a young student, I did not realize that these formative experiences with oppressive structures within education would fuel me to purse educational and social justice as a career and personal passion. While I could focus on the negative aspects of my educational experience, instead, I choose to focus on the educators that got me to my proudest moment.

My freshman English teacher, Mrs. Neumeister advocated for me, and got me into that honors English class after I’d failed the exam twice. She knew my capabilities exceeded far beyond how I performed on an exam. Cassie Zimmer recognized my strengths and grew me into the teacher I am today, and made me Lead Teacher when others believed I never could be. Mika Jain, Veronica Hopkins, Liliana Funes and Danielle Johnson modeled how to be extraordinary teachers, and I strove to be more like them every time I stepped into the classroom. Julie Alber and Sarah Buchan showed me how to value all unique strengths of my students, and how to be a fun-loving and joyful primary school teacher. Sean Collins supported me in reflecting as a practitioner, and encouraged me to do what was best for kids. Caren Holtzman convinced me to stick with a teacher preparation program, and believed in me when I was an undergrad and overwhelmed with the thought of managing a classroom.

I entered the UCLA ELP program with imposter syndrome, but left stronger, smarter, and ready to take on anything due to my amazing cohort members, who never stopped supporting and believing in me. During one of my first doctoral classes, Dr. Pat McDonough pulled me aside and said, “Melissa, find your voice.” With the love and support of Cohort 27, ELP faculty and staff, and all the teachers that believed in me along the way, I have found my voice.

This work is dedicated to all the students who are told, “You can’t” and “You won’t”. Believe the teachers that believe in you. We stand with you in your fight to tell the world, “I can, and I will!”

TABLE OF CONTENTS

ABSTRACT OF THE DISSERTATION ii
DEDICATION PAGE v
TABLE OF CONTENTS vii
LIST OF TABLES x
ACKNOWLEDGEMENTS xi
VITA xii
CHAPTER 1: INTRODUCTION 1

Statement of the Problem 1
Background of the Problem 1
The Need for a Highly Skilled Computer Science Workforce and the Role of Education 4
My Experience Teaching Computer Science 5
Gender and Racial Stereotypes in Computing 6
Transformative Computer Science Education 9
Existing Gaps in Research 12

Statement of the Purpose 13

Research Questions 14

Overview of the Research Design 14

Study Significance 15
CHAPTER 2: LITERATURE REVIEW 16

Introduction 16

Impact of Sense of Belonging for Marginalized Groups 16

Impact of Authentic Computing Experiences on Computing Identity 18

Professionally Authentic vs. Personally Authentic Experiences 18

Using Authentic Computing Experiences to Address Structural and Cultural barriers in the Computing World 19

Value of Authentic Computing Experiences in Elementary Settings 20

Professionally Authentic Computing in Elementary School Settings 21

Personally Authentic Computing in Elementary Grades 23

Current Implementation of Computing in Elementary School Settings 25

Teacher Preparation for Teaching Computer Science 26

Challenges for Elementary School Teachers 27

Culturally Responsive Computing Pedagogy 30
Conceptual Framework..33
CHAPTER 3: RESEARCH DESIGN 35
Introduction...35
Research Questions...35
Research Design and Rationale 36
Methods 36
Site and Population Selection 36
Data Collection Methods 37
Semi-structured Interviews..39
Lesson Review During the Interview 40
Data Analysis Methods 41
Positionality 46
Ethical Issues 47
Credibility and Trustworthiness...47
Study Limitations...48
Conclusion 49
CHAPTER 4: FINDINGS 50
Introduction...50
Principal Findings..50
Equity as Opportunity: Equal Access, Encouraging Female Students, and Representation
52
Equity as Differentiation...53
Teachers' Goals for Computer Science Instruction 54
Providing Access to Opportunities 55
Positive Identification with Computer Science 56
Empowering Students to Use Computer Science for Social Change 60
Increasing Academic Achievement..63
Teachers' Strategies for Equitable Practices 63
Fostering an Inclusive Classroom Environment...64
Incorporating Rigorous Pedagogy 71
Incorporating Student Agency and Voice 74
Creating Cross-Curricular Connections..78
Addressing Racism in Computer Science...80
Modeling Professional Inclusiveness 83
Incorporating Relevant Pedagogy..84
CHAPTER 5: DISCUSSION 86
Interpretations of Equity 87

Acknowledge Racism in CS and Enact Anti-Racist Practices....90
Create Inclusive and Equitable Classroom Cultures 90
Student Voice, Agency, and Self-Determination are Prioritized in CS Classrooms ..98
Family and Community Assets are Incorporated into CS Classrooms....99
Diverse Professionals and Role Models Provide Exposure to a Range of CS/Tech Careers 100
Limitations....100
Recommendations for Practice 101
Recommendations for Future Research....102
Conclusion 103
Appendix A....105
Appendix B....108
Appendix C (Kapor Center, 2021)....116
References....131

LIST OF TABLES

Table 3.1: Respondent demographics summary ……………..…………………………………40

Table 3.2: Thematic codes derived from teachers' responses ……………..……………….……46

Table 4.1: Themes mentioned by participating teachers in the categories of equity, instructional goals, and strategies……………………………………………………………49

ACKNOWLEDGEMENTS

First, I would like to thank my committee members, Dr. Kimberly Gomez, Dr. Ananda Marin, Dr. Jean Ryoo, and Dr. Kristen Rohanna for their support during the process of conducting my research and completing this paper. I would also like to thank my committee chair, Dr. William Sandoval, for his support and guidance as I worked on this culminating project over the span of two years in the Educational Leadership Program at UCLA. I could not have completed this work without the support of these UCLA professors, and my fellow Cohort 27 members.

I'd like to thank my parents, Tina and Len, for preparing me for the grueling process of writing a dissertation. As a young student, I had no idea that by indenturing me to complete Kumon packets, my parents were teaching me the same skills required of a dissertation writer: grit, determination, and overcoming frequent tantrums.

I must thank my husband, Scott Toohey, who supported me unconditionally, especially through the aforementioned tantrums. Thank you for standing by my side, taking on extra duties around the house, and providing the grumble-cisms to exorcise the tantrums from my body during all the times I chose to write instead of spending time with you. I could not have accomplished this without the light and levity you, Brezel, and Schnitzel brought during the difficult times. (Thanks to Brezel and Schnitzel for all the sloppy kisses provided!)

Finally, I'd like to thank all the friends and family that supported me. Thank you for understanding why I fell off the face of the earth for three years, and finding ways to keep my spirits up when I reemerged occasionally. I would not have accomplished this without the endless support, ski trips, games, laughs, great meals and countless requests to edit my writing.

VITA

2010	B.A. Political Science University of California, San Diego San Diego, California
2011	Master of Education University of California, San Diego San Diego, California
2011	Multiple Subject Teaching Credential Long Beach Unified School District Long Beach, California
2011-2012	Technology Teacher Conejo Valley School District Westlake Hills, California
2012-2015	Kindergarten and First Grade Teacher Seven Arrows Elementary School Pacific Palisades, California
2015-2017	Founding Coding, Engineering and Design Thinking Teacher KIPP Ignite Academy South Los Angeles, California
2017	Farmers Insurance $100,000 Dream Big Teacher Challenge Award
2017-2019	EdTech Curriculum Specialist and Coach Los Angeles, California
2019-2022	Director Of Education Ozobot Los Angeles, California
2022.	Curriculum Development Director Seesaw Learning Los Angeles, California

PRESENTATIONS

Toohey, M. & Cope, C. (March 2017) Coding, Engineering, and Design Thinking in Kindergarten. Presentation at SXSW Edu Conference, Austin, Tx

CHAPTER 1: INTRODUCTION

Statement of the Problem

Background of the Problem

The United States does not have enough graduates to fill the projected number of current and future computing roles. According to the U.S. Bureau of Labor and Statistics, computing jobs are projected to grow by 12% between 2018-2028, far quicker than other occupational fields. Cloud computing, collection and storage of big data, and information security skills will be in the highest demand. The Bureau predicts that over 500,000 jobs will be added. In the U.S., the average median wage is $38,640, while jobs in computer and information technology command an average median wage of $86,320. Early access to computer science education is increasingly relevant for students as it opens opportunities to higher earnings in the workforce. 3.5 million computing related jobs are expected to open by 2026, but only 19% of these are projected to be filled by U.S. Bachelor degree recipients (Department of Labor Statistics, Employment Projections). The industry is male-dominated, and as a result, excludes half of the potential population that could fill these jobs. Additionally, other perspectives highlight the opportunities in the technology field as an opportunity to empower, transform, validate, comprehend, and validate underrepresented minority populations (Gay, 2010; Scott, Sheridan, and Clark, 2015)

The technology workforce is dominated by White and Asian[1] males, and the lack of diversity in the workforce and in computing programs that engage learners is well

[1] Not all Asian subgroups are represented in the technology workforce.

documented (NASEM, 2021a). The causes of underrepresentation of particular populations is complex (Charleston et al., 2014; NASEM, 2018; Scott, Sheridan, and Clark, 2015), but some factors include access to courses and to technology, cultural barriers linked to norms and practices in the industry, and stereotypes and implicit biases (NASEM, 2021a). The underrepresentation from more diverse groups hinders invention (Scott, Sheridan, and Clark, 2015). White male bias has been well documented in technology. More specifically, it is generally accepted that white male bias affects search algorithms, artificial intelligence, and other search tools (Buolamnwini, 2021).

While the tech industry and American government realize the need for a highly skilled workforce, the computer science industry lacks diversity. In 2018, women comprised 57% of the national workforce, yet they only held 26% of computing jobs. Among this percentage, women of color were represented at alarmingly low rates. Six percent of the computing workforce were Asian women, while 3% were Black women, and 2% were Latina (Department of Labor Bureau of Labor Statistics, 2018). The pipeline feeding women into the workforce shows similar rates of inequity. In 2017, 57% of Bachelor's degrees were awarded to women, but women earned only 19% of Computer and Information Science degrees. Moreover, this percentage represents a significant decline over the last three decades: in 1985, women earned 35% of bachelor degrees in computer science (NCES, 2017). The low rate of women’s participation in computer science programs has been problematic. One explanation for this decline can be traced to early childhood socialization (Margolis & Fisher, 2002). Women and girls are not pursuing computer science in secondary education at the same rates as their male counterparts. The rates of female participation in computing courses does not proportionally reflect the student population, or meet the ratio of females to males in advanced mathematics courses (College Board AP Program Summary Report, 2018). The data suggest women are systematically disadvantaged when pursuing computing interests or coursework (Margolis, 2008). Opening computing opportunities to more students provides an opportunity to fill the demand for candidates ready to fill computing jobs in the tech industry.

Diversifying the tech workforce will meet job growth demands and provide an opportunity for better paying jobs for women and people of color. The tech industry has a focused interest in promoting computer science education. With the projected job growth in the tech industry and lack of qualified candidates to fill jobs, private companies have partnered with non-profit organizations to bring computer science to a diverse population of students. Google, IBM, Amazon, and other major tech companies have partnered with organizations like CSforALL (Computer Science for All) and Code.org to promote computer science in schools, and some have even created their own programs like Amazon's Future Engineer Program. CSforALL and Code.org have created movements that promote computer science instruction for all students, regardless of background, socio-economic level, or gender. IBM's Pathways in Technology Early College High School (P-TECH) prepares high school students to be industry ready without the need for a college degree. Due to the industry's interest in gaining prepared workers, many companies have increased funding and research to promote the outcomes that serve their interests. The technology industry recognizes their impending employment gap and has engaged with both the government and education sectors to promote and implement computer science in education. These organizations have responded to the needs and demands of the industry by providing more students access to computing experiences. While this is one argument for the need for computer science education, an opposing camp argues that computer science education is needed to empower students to positively contribute to their world. This idea will be explored in Chapter 2, within the conceptual framework section.

The Need for a Highly Skilled Computer Science Workforce and the Role of Education

Diversifying the workforce requires professionally and personally authentic computing experiences in K-12. In the past decade, computer science education expansion has focused on professional authentic experiences, and has been broadly supported by public policy and a growing number of K-12 educational institutions (NASEM, 2021). By 2028, the United States is projected to have 19 million more Science, Technology, Engineering, and Mathematics (STEM) jobs than workers to fill them (Legewie and DiPrete, 2011), and policy makers, educators, and industry leaders recognize the need to generate a pipeline to fill these roles. According to the Computer Science Teachers Association (CSTA) and Code.org, computing jobs are the number one source of new wages. In addition, 9 out of 10 parents want their students enrolled in computer science courses, yet only 45% of High Schools in the United States offer computer science courses (Trends in the State of Computer Science Report, 2019). All 50 states in the U.S. have or are in the process of adopting policies to promote access to computer science education (Trends in the State of Computer Science Report, 2019). Recently, there have been increasing calls to provide access to computer science into elementary and middle schools, which has led to efforts in the United States to introduce computer science to young learners (Rich et al. 2020). These endeavors have attempted to call for the development of interdisciplinary approaches to the integration of computing within STEM teaching and learning; build capacity within the educational system to support CS education; and examine ways to broaden access and participation of learners who have historically been underrepresented (NASEM, 2021a).

My Experience Teaching Computer Science

While the industry, general public, and stakeholders understand the concerning need to fill job roles, as an educator, my experience proved how computing experiences empowered the marginalized. During my time as the Founding Coding, Engineering, and Design Thinking Teacher at a public charter school in South Los Angeles, I served Black and Latino/a elementary students. 97% of students qualified for free and reduced lunch. My role served dual purposes. In the mornings, I taught small-group guided reading. In the afternoons, I taught coding, engineering, and design thinking to all the students at my school. One of my guided reading students, Catalina, entered kindergarten with Individualized Education Plans (IEPs) for a developmental delay and speech. She was placed in the lowest reading and math group, yet she was my strongest coder. I witnessed how her mind worked differently than her peers. Catalina's strengths included being able to break down problems, and take step by step approaches to solve the problems. While her classmates in the highest reading and math groups struggled with computing assignments, Catalina shined.

In this role, I often asked myself, "What messages are being sent to Catalina about what she is and is not able to do?" As one of her teachers, I wanted to focus on her strengths.

Catalina became my unofficial coding helper, I would publicly highlight her strengths and ask her to help other students who were struggling and frustrated with coding. An unexpected, but positive, outcome was that Catalina proved how computing can reach students that may struggle with traditional subject areas, and build their confidence and love for school and learning. Traditional subject areas may not offer all students the opportunity to find their strengths, but Catalina represents what drives me as an educator and a practitioner - the will to help all students discover all of their strengths, even non-academic ones. Catalina proved to me that all brains work differently. As an educator, I believe all students deserve to access high-quality education that supports and includes their cognition level, strengths, weaknesses, gender, interests, community, and culture. My experience with Catalina pushed me to explore equity and computing on a deeper level.

Gender and Racial Stereotypes in Computing

Gender and racial stereotypes regarding computing ability appear in early childhood. Girls and women who could potentially fill computing roles are discouraged from pursuing computing careers (Margolis & Fisher, 2002). Women must be part of the computing and design teams that are reshaping the world if new technologies are meant to serve both men and women (Margolis & Fisher, 2002). It is important for women to be included in computing and technology conversations, use technology, but most importantly, design and create technology (Margolis & Fisher, 2002). The patterns of who are and who are not represented start early. Access to computing varies by grade, rates of poverty at a school, ethnic make-up of a school, and school size (Banilower et al., 2018), but early gender socialization impacts how females self-select out of computing experiences.

The gender issue in computing can be traced to socialization in childhood. White and Asian Boys get more access to computing earlier in their lives, which leads them to be "boy wonders" (Margolis & Fisher, 2002). Boys are encouraged to take risks and expected to be adventurous, while girls are encouraged to be cautious and careful (Margolis & Fisher, 2002). Early exploration of computing is an exception for girls, when compared to their male counterparts (Margolis & Fisher, 2002). Home environments are also deeply influential in development of computer science interests (Margolis & Fisher, 2002). While boys are encouraged to explore computing at a young age with hands-on experiences, girls receive encouragement or enthusiasm from parents much later in life. However, parents tend to encourage their daughters to take computer science in high school, or to pursue the major in college (Margolis & Fisher, 2002). Boys are often seen as “boy wonders” as their male traits, early hands-on exposure, and success in sciences make them appear more apt and predisposed to be stronger in their computing traits. Far fewer girls display these traits, and that results in a culture of discriminating by gender (Margolis & Fisher, 2002). Boys are seen to be naturally good at computing, while girls see themselves as “outsiders” and not valuable contributors to computer culture and curriculum (Margolis & Fisher, 2002). These early gender experiences affect the bias and challenges that females face when exploring computer science from an early age through their academic and professional careers. While the industry and educational sector have realized a need to provide access to computer science education to fill the job gap, there is ample opportunity for opening access to computer science so that more girls and students of color can participate. Yet, these initiatives face challenges.

Diversifying the workforce requires personally and professionally authentic computing experiences starting in elementary school. Studies have shown that providing these experiences may increase student engagement and sustained interest in computer science (Eglash et al. 2013). These opportunities, along with implementation of Culturally Responsive Pedagogy (CRP) and Culturally Responsive Computing (CRC) can potentially increase the number of women and underrepresented minorities that pursue technology as a career. Culturally responsive teaching leverages students' strengths to serve as an anchor in the process of transforming traditional teaching practices that have oppressive histories (Gay, 2010). This allows students' to participate in a learning process that is student-centered and based on the needs of the students (Gay, 2010). CRC leverages and builds on CRP. The goals of CRC are framed to address the persistent gap in historically marginalized groups' participation and access to technology to increase the focus on technology education and the workforce, and focuses on teaching computing in ways that allows creation to be transformative (Scott, Sheridan, and Clark, 2015).

Bringing personal and professionally relevant experiences into elementary school settings meets a number of challenges. By providing historically marginalized students with access and opportunities to computer science experiences in elementary school grades, students will ultimately have more access and opportunities in computer science as they progress in their education. This may lead to increased engagement in the subject matter. By providing teachers with tools and skills to deliver meaningful and authentic computing experiences, more students will have access and exposure to developing computing skills. Authentic computing experiences can be defined as professionally and personally authentic, and are explored more deeply in the literature review. Incorporating authentic computing experiences could potentially positively impact the population in the computing workforce by creating a more diverse and representative computing workforce population.

Transformative Computer Science Education

In addition to projected workforce needs, computer science has opened opportunities for marginalized populations. There is a persistent gap in historically marginalized groups' ability to participate in creating technology in relation to the increased focus on technology in education and the workforce (Scott, Sheridan, and Clark, 2015). Students from these historically marginalized groups have historically been excluded from computer science education, and as a result, are underrepresented in the workforce. In addition to workforce opportunities, providing culturally responsive computing opportunities allows students to become social justice innovators, provides opportunities to exceed expectations (e.g., high school dropouts reengaging in school), allows communities to rally behind their students, and allows youth marginalized by race, social class, and gender to advance thought and action to unprecedented heights (Scott, Sheridan, and Clark, 2015). Computer science education allows for one pathway towards a social justice end (Scott, Sheridan, and Clark, 2015). Through computer science education, historically marginalized students can be empowered to create technology to transform their communities and worlds. The Culturally Responsive Computing framework has outlined that:

1. All students are capable of digital innovation,
2. Learning contexts support transformational use of technology,
3. Learning about one's self along various intersecting sociocultural lines allows for technical innovation,
4. Technology should be a vehicle by which students reflect and demonstrate understanding of their intersectional identities
5. Barometers for technological success should consider who creates, for whom, and to what ends rather than who endures socially and culturally irrelevant curriculum (Scott, Sheridan, and Clark, 2015).

Intentional incorporation of students' identities and cultures into computer science education allows students to create relevant technologies in potentially transformative ways. These voices have been historically excluded from the technology ecosystem. Allowing their participation opens up opportunities for these populations to create technology for meaningful purposes that the dominant white and male culture currently does not allow.

Through computer science education, students have the opportunity to engage with and take a transformative stance towards technology, and students are empowered to design technologies that do what these students want them to do (Scott, Sheridan, and Clark, 2015). Historically, these groups have been excluded from participating in the design of technology, and as a result, the technology is less aligned with the experiences and interests of underrepresented populations (Scott, Sheridan, and Clark, 2015). For example, Eglash et al. (2013) cites a study that explores how the chemical industry routinely creates deadly pesticides, dry cleaning agents, industrial solvents, and other products, and these effects are disproportionately concentrated in low-income communities (Bullard, 198), and connects this idea to the impact of marginalized populations being left out of the creation of technology. The author described how this is one example of how detrimental the outputs of design and innovation can be if taken out of social and cultural contexts. Inclusion of marginalized populations would benefit all citizens, and provide better approaches to solve society's problems.

Past research has shown that empowering marginalized communities with STEM skills benefits their communities. Terry (2011) worked with Black students to calculate crime rates to invalidate public claims of government reports that supported increased tax rates due the government's claim regarding the cause of decreased crime rates. In this situation, students were empowered through STEM education to counter oppressive forces in their community. The students proved that the crime rates were not tied directly to reasons the government claimed, and their analysis supported their community in fighting the increased tax rates.

Studies have shown that girls from marginalized backgrounds leverage innovation to display their growing knowledge, and become motivated to create products that challenge perceived notions of themselves and communities (Scott, Sheridan, and Clark, 2015). Providing opportunities for all students to participate in computer science allows students the potential to pursue professional opportunities and empowers them with skill to create transformative technology to advance their community. Ignoring these gaps in technology would perpetuate inequities and disproportionately impact low-SES communities of color (Gorski, 2009).

In addition to providing opportunities, pedagogy and how computer science is introduced to students is critical. CRC is a valuable approach for thinking about how to do. Particularly, the Kapor Center's Framework for Culturally Responsive-Sustaining Computer Science Education lists six core components to promote inclusion:

1. Acknowledge Racism in CS and Enact Anti-Racist Practices
2. Create Inclusive and Equitable Classroom Cultures
3. Pedagogy and Curriculum are Rigorous, Relevant, and Encourage Sociopolitical Critiques
4. Student Voice, Agency, and Self-Determination are Prioritized in CS Classrooms
5. Family and Community Cultural Assets are Incorporated into CS Classrooms
6. Diverse Professionals and Role Models Provide Exposure to a Range of CS/ Tech Careers

Existing Gaps in Research

While there is a growing body of literature focused on elementary school computer science education, there is minimal research around teacher preparation, systems of support, professional development, training, and pedagogical content knowledge for elementary school teachers teaching computer science. In the literature review that follows this chapter, I will explore and build upon some of the literature that is currently available. Some of this research currently available includes how elementary school teachers integrate computational thinking into their math and science lessons (Rich et al. 2020). There are few descriptions of how computational thinking has been integrated into core subjects at the elementary school level (Rich et al. 2020). Computational thinking is closely connected to computer science in the areas of content and practice. However, there is a lack of research in this area as well as computer science implementation at the elementary school level despite the importance of such skills for young students. Many state governments have computer science education mandates for elementary school students, and have adopted Computer Science Standards (2020 State of CS Report). Some educators argue that providing students with computer science education earlier in their academic careers is one way to promote equity and inclusion in the computer science field. In the literature review, I will provide an overview of the growing body of research that discusses how elementary school students benefit from computer science education, and that they are able to understand, acquire, and master computer science concepts and skills. With such limited literature on the topic of equity and computer science education at the elementary school level, I hope that my dissertation will fill this gap.

Statement of the Purpose

The purpose of this study is to examine how elementary teachers think about equity in their computer science instruction, what goals they have for computer science instruction, and the strategies they articulate to make their practice more equitable. The focus of this study is to explore how teachers implement computer science in the elementary classroom and the ways in which they think about equitable practices in relation to their computer science instructional goals. This study seeks to understand how teachers implement and articulate strategies to make their practice equitable. The study seeks to understand the participants' views and experiences and focuses on direct experiences of the participants in order to answer how elementary school teachers think about and implement equitable computer science instruction.

Research Questions

1. How do elementary CS teachers think about equity in relation to their goals for their computer science instruction?

2. What strategies do elementary computer science teachers articulate to make their practice equitable?

Overview of the Research Design

For my study, I focused on a group of elementary school teachers who teach computer science. I included a population of teachers that self-identify as teachers of computer science. I used interviews and lesson plans in my study. Because computer science is a newly emerging content area for elementary school teachers, environments will differ greatly. The study focused on any self-identified teachers of computer science that work directly to teach elementary school-aged children. In this context, elementary school was defined as kindergarten through fifth or sixth grade. Some districts include sixth grade within elementary school, so the study will utilize the label of "elementary school" depending on how the school or district classifies the sixth grade group.

Study Significance

This research could help decision makers support teachers in implementing computer science. This research will help teachers by guiding their instructional practices. Very little is known about how elementary teachers think about computing and computer science in relation to issues of equity. Understanding how teachers think about equity, the goals they have for computer science instruction, and their strategies for implementing equitable computer science learning experiences will provide needed insights into teaching computer science to elementary school students. Understanding how to support elementary teachers in thinking about and enacting equitable computing experiences is a prerequisite for professional learning experiences and curriculum that can support more equitable computing experiences at the elementary level. This can eventually support diversification of the computing workforce, and empower students from historically underrepresented populations to create and transform their lives, communities, and world with technology.

CHAPTER 2: LITERATURE REVIEW

Introduction

In this literature review, I discuss research around sense of identity as it relates to computer science, and how authentic computing experiences may be the key to increasing girls' and underrepresented minority groups' participation in computing. Then, I examine successful efforts in teaching computing in elementary schools. Personally and professionally authentic computing experiences are defined, and examined in the context of the elementary school classroom. Then, I review how successful computer science implementation requires elementary school teachers to be interested in and capable of delivering computing lessons in their classrooms. After, I review what current research has shown about elementary teacher training and attitudes about implementing computer science. I end by discussing what the research community knows about obstacles in teaching computer science at the elementary school level, and how Culturally Responsive Computing Pedagogy could increase marginalized populations' interest in computer science. This review will highlight that, the majority of extant research aims at professional authenticity, and will explore the growing body of research focused on personally authentic computing experiences at the elementary school level.

Impact of Sense of Belonging for Marginalized Groups

By providing opportunities for teachers to include pedagogical approaches, materials, and resources that build on students' interests, identities and backgrounds, teachers could potentially provide opportunities for students from diverse backgrounds to see themselves in computing roles (NASEM, 2021a). Generally, three components of identity, which include a sense of belonging, achievement, and behaviors, are accepted as the elements that determine if a person sees themselves recognized by others as someone who understands and uses the practices of that community (Carlone and Johnson, 2007; Cheryan, Master, and Meltzoff, 2015; Erikson, 1968; Lave and Wenger, 2002). Patterns of representation and underrepresentation start early, and many students from historically underrepresented groups are excluded from computing opportunities.

Sense of belonging impacts how women and historically underrepresented groups view themselves in the realm of computer science. Studies have shown that the disproportionality of women and of Black, Latinx, and Indigenous individuals in STEM fields is closely linked to identity (NASEM, 2021a). Studies have also shown that women with a strong STEM identity are more likely to persist in the field (Jones, Ruff, and Paretti, 2013). In addition, one study showed that students at Historically Black Colleges and Universities with strong STEM identities were more likely to enter the science field upon graduation (Stets et al., 2017). Access to technology, economics, racism, sexism, stereotypes, and implicit bias impact computing identity (NASEM, 2021a). Sense of belonging and identity are closely linked. Providing authentic computing experiences is one way to develop historically underrepresented groups' computing identities.

Impact of Authentic Computing Experiences on Computing Identity

Authentic computing experiences impact student development of their computing identity (NASEM, 2021a). Authentic learning can be defined as providing opportunities for students to connect real-world experiences they care about to professional practice (NASEM, 2021a). This is one way to increase reach to a more diverse range of students (NASEM, 2021a). Students need multiple experiences to develop continued interest and skills, and formal and informal experiences contribute to whether young people show sustained interest in computing (NASEM, 2021a). In addition, Ryoo (2019) found that authentic learning experience promoted engagement in high school computer science classrooms. Even though this research was focused on high school classrooms, it opens the door to examine if the same can be said of elementary school classrooms. Motivation can be narrowed down to three factors: 1) personal interest in the topic; 2) their perceived value of the topic; and 3) their development of competence and skills while engaging in the content matter (Bathgate and Schunn, 2017). Links between interest, identity, and persistence are closely related to student interest and competencies in STEM fields (NASEM, 2021a). Authentic experiences potentially help students build their computing identities, and in turn, could impact how structural and cultural barriers are addressed in the computing world.

Professionally Authentic vs. Personally Authentic Experiences

A recent report distinguishes between two kinds of authentic computing experiences (NASEM, 2021a). *Professionally* authentic experiences are those that align with and reflect the practices of the discipline. Professionally authentic experiences are designed to be close approximations of what a computing professional would do. *Personally* authentic experiences are those that are reflective of students' backgrounds, cultures, interests, prior knowledge, and environments. In current computer science teaching practices, there is heavy emphasis on modeling instruction around the work of the professional and increasing experiences that connect to real world problems that students care about (NASEM, 2021a). Personally motivating computing experiences may not be professionally relevant, but content reflecting professional practice can be grounded in personal interest (NASEM, 2021a). Professional and personal authenticity can be intertwined to create experiences that students find motivating. Providing students with professionally and personally authentic experiences may be one mechanism in addressing barriers to access in the computing world (NASEM, 2021a). Students who see themselves and their culture in the content they are learning may become more interested in pursuing computing in their education and careers.

Using Authentic Computing Experiences to Address Structural and Cultural barriers in the Computing World

Gaps in access to rigorous computer science education have been well documented, and it has been widely accepted in the field that solely increasing access to opportunities is unlikely to reverse inequities (Gomez, Lee, & Berkhoudt Woodman, 2022). According to NASEM (2021a), providing learning experiences that are culturally relevant and incorporate student interests, identities, and backgrounds opens access to personally authentic learning opportunities. Personally authentic opportunities may engage more learners from historically underrepresented groups, as opposed to solely focusing on professionally relevant practices.

Preliminary research has shown that these personally relevant experiences have the potential to engage students in STEM and computing (Lim and Calabrese Barton, 2006; Migus, 2014). Providing personally authentic experiences can be a powerful tool to engage students and increase interest in computing, as long as the experiences are relevant and reflective of students' identities and cultures.

Students from underrepresented groups are less likely to experience personally, culturally, and professionally relevant opportunities (Cheryan et al., 2016; Rodriguez and Lehman, 2018), which potentially impacts their sustained interest and access to computing experiences. The technology field has been historically dominated by white men, and the field has been reflective of their culture (NASEM, 2021a). In addition, professional practices in computing have historically excluded women and other marginalized groups. One can infer that professionally authentic experiences are strongly correlated with the gender and cultural associations of white males. As a result, the feeling of authenticity in computing can be exclusionary to women, underrepresented minority groups, and those with perceived differences in ability. For groups that have been historically excluded, inequities, biases, and stereotypes discourage participation.

Value of Authentic Computing Experiences in Elementary Settings

Authentic computing experiences in elementary school in STEM would be valuable to build computing identities and provide opportunities for students to see their culture reflected in computing opportunities. As stated previously, one study found that gender stereotypes around computing begin to form in first grade (Master, Cheryan, Moscatelli &

Meltzoff, 2017), and girls participate in computing courses at significantly lower rates at the middle school, high school, and university levels (Margolis & Fisher, 2002). In order for teachers to effectively address the gender and racial gap in the tech field, providing *authentic* experiences is key.

Computer science instruction at the elementary school level is a recent development, and not universally required in the United States. Some states have begun initiatives to create computer science standards, (State of Computer Science Education, 2020), and some states have mandated computer science as a high school graduation requirement. Systematically, teachers are not prepared to deliver basic computer science instruction (NASEM, 2021a, State of Computer Science Education, 2020), and as a result, they may not be prepared to adjust their teaching practices to incorporate personally and professionally authentic computing experiences.

Professionally Authentic Computing in Elementary School Settings

An emerging body of research on computing instruction in elementary grades demonstrates that, with appropriate instruction and technologies, children are capable of engaging in a variety of computational thinking practices. Existing studies focus on skills and/or knowledge outcomes, which aligns to the professional focus of computer science education. Elementary age students who get access to age-appropriate instruction can actively engage in computer programming (Bers, 2010). Generally, elementary students increased their algorithmic thinking and problem solving skills after engaging with a computational thinking game (Gürbüz et al., 2017). Elementary school-aged girls grew their computing knowledge by defining computing vocabulary, using checkpoint activities with immediate and corrective feedback, and scaffolding of coding concepts with unplugged activities in an informal learning context. These approaches were necessary in implementing and integrating computational thinking with science, and engaged learners that expressed a low interest in science (Luo et al. 2020). Students are able to express ideas that demonstrate computational thinking, and deeper subject matter understanding, such as math and science (Waterman et al., 2020). The outcomes of the studies described above focus on professional skills and knowledge outcomes, in addition to showing what students can accomplish in computing. Furthermore, girls verbalized that programming could be part of their future plans (Kalelioğlu, 2015).

More specifically, studies have shown findings at particular grade levels within elementary school settings. Primary school students have developed positive attitudes toward computing and girls have been shown to be as successful as their male counterparts (Kalelioğlu, 2015). In an eight-week course, pre-kindergarten through second grade students were able to master basic robotics and programming concepts, and older students were able to master increasingly complex skills in the same amount of time (Sullivan and Bers, 2016). K-2 students were able to use computational thinking competencies to create solutions for engineering problems (Ehsan et al., 2020). K-2 students were able to utilize computational thinking practices, such as problem decomposition, algorithm generation, logical reasoning and system integration of multiple components. These skills helped students connect to the engineering design work in their class (Hynes et al. 2016). Third and fourth grade students were able to develop relevant programming and computational modeling skills (Sengupta &

Farris, 2012). After receiving computational thinking instruction, fourth graders were able to recognize the need for debugging, but struggled to provide specific instructions, and understood that small errors could change outcomes (Dwyer et. al., 2014). Fifth grade students were able to improve their computational thinking knowledge by programming and choreographing a virtual character's movements (Leonard et al., 2015). These studies show that researchers and teachers have prioritized professionally authentic experiences, and that there is a growing body of research focused on personally authentic computing experiences.

Personally Authentic Computing in Elementary Grades

One study incorporated personally authentic computing in an elementary school classroom. Searle & Kafai (2015) focused their study on 10 Native American boys and incorporated computing with the creation of electronic textiles. They hypothesized that e-textiles would be appealing to Native American boys due to the strong tradition of community-based craft traditions. Their findings included the importance of connecting computing to the larger community values of a population, and the importance of allowing students to make decisions within the context of the design challenge. Their work connects to the efforts to engage underrepresented populations in computing and emphasizes the importance of culturally relevant computing. In a study of fourth through sixth graders, researchers found that students interested in programming identified as having greater creative and programming self-efficacy. In this study, boys showed a greater interest in programming than girls did. In higher grade levels, students valued programming as less meaningful. In this study, the authors defined meaningfulness as more likely than others to feel empowered, more likely to start a task, exert more effort to complete the task, and succeed (Kong et al. 2018). Research on this topic in the elementary school settings is still growing, but there is research around personally authentic computing in a summer camp context. Culturally relevant themes were utilized for female Latinx students and increased engagement with computer science (Franklin et al., 2011). These important studies highlight the engagement of underrepresented populations in computing, and how computing curriculum and content can be meaningfully delivered to underrepresented students. Since minimal research has examined the idea of tying computer science instruction to personally authentic computing experiences, there is an opportunity to contribute to this field of knowledge. There is still not enough information regarding whether this approach to computer science education is effective in encouraging students to sustain interest and pursue computing careers at the elementary school level, but studies have shown that personally authentic experiences promote engagement in high school settings.

While this study is focused on elementary school computer science education, past research of computer science education in high school settings has explored the link between personally authentic computing experiences and engagement. Ryoo (2019) found that the key pedagogical practices that had greatest impact on youth's interest and engagement with CS included: (1) demystifying CS by showing its connections to everyday life; (2) addressing social issues impacting both CS and students' communities; and (3) valuing students' voices and perspectives. Three teachers and over 90 of their students were studied. The researcher provided examples of how each teacher highlighted and incorporated students' personally authentic interests through their pedagogical approaches, and this resulted in students and teachers expressing deeper engagement with computer science (Ryoo, 2019). One example provided was when a teacher asked a student to explain skateboarding by using algorithms, a foundational computer science concept. The student reported that they did not see a clear connection between their deep interest in skateboarding and computer science, yet the teacher provided a meaningful and tangible way for the student to engage in the computer science content through their interest in skateboarding (Ryoo, 2019). Pedagogical practices can be leveraged to incorporate personally authentic interests and experiences. This study hopes to provide more literature around the pedagogical practices that elementary teachers articulate they implement in order to make their practices more equitable, and seeks to fill the gap in literature on this topic in the elementary school setting.

In addition, past research has shown that high school students learn best when they solve problems that are inspired by their own questions of the world (Goode and Margolis, 2011). Though curriculum alone cannot solve the equity issues within computer science education, meaningful engagement with computer science can be accomplished through acknowledging community and cultural wealth and assets, and incorporating student experiences, identities, perspectives, and out of school skills into learning (Goode et al, 2012).

Current Implementation of Computing in Elementary School Settings

There is a developing body of research around computer science implementation in elementary schools. In particular, a few studies have examined the impact of computational thinking in elementary school settings. Studies have explored how primary school teachers connected computational thinking to other content topics and explored misconceptions teachers held around computational thinking and computer science (Duncan et al., 2017). Others explore how teachers' understanding of computational thinking developed, grew, and became more nuanced over the course of a year of professional development in integrating computational thinking into science inquiry (Yadav et al., 2018). Another studied how teachers provided explicit computer science instruction in center-based or whole-class groupings (Israel, 2015). Another study documented how specific pedagogical strategies provide opportunities for students to engage in computational thinking during math and science lessons (Rich et al., 2020). While this area of teaching, learning, and research is developing, it can be inferred that some teachers, schools, districts, and states recognize the need and importance of computing instruction. Because of teachers' lack of knowledge and preparedness in the content area, there is a need for supporting teachers in creating meaningful computing experiences.

Teacher Preparation for Teaching Computer Science

Teacher knowledge directly impacts their teaching (Toom, 2017), as their understanding impacts how they specify learning goals for each lesson (Hiebert et al., 2007). Teachers must understand how people learn in order to make informed decisions about how to proceed when students do not master concepts or skills (Darling-Hammond, 2006). If teachers are not able to combine their pedagogical knowledge and practice with their computer science content knowledge, they will face many challenges in implementing computer science in their classrooms. Teachers must understand how to utilize computer science content knowledge and translate it into classroom practice (Toom, 2017). Teacher training and professional development are essential in preparing teachers to deliver computer science at the elementary level. Professional development offers educators the opportunity to translate computer science knowledge into teaching practices (Rich et al. 2020), but, in general, teachers are not prepared or trained to design and implement rigorous and accessible computer science learning opportunities that reflect computational cognition or contexts relevant to the social concerns of students (Gomez, Lee, & Berkhoudt Woodman, 2022).

One study examined computer science professional development programs and evaluated programs based on duration of training, support for classroom implementation, explicit focus on active learning methods, focus on pedagogical content knowledge, student performance data, and collaboration with local school or district administration (Menekse, 2015). This study found that only one computer science professional development program met all of these criteria. Forty-three percent of reviewed programs included 50+ hours of professional development, 52% provided in-classroom support, 62% included active learning methods, 38% focused on pedagogical content knowledge, 33% collaborated with school and/or district administration, and only 14% provided students learning data. Policy standards have been set in many states, but enforcement and implementation prove to be a challenge for teachers.

Challenges for Elementary School Teachers

One of the main obstacles for successful CS instruction for elementary school teachers is that they are not formally prepared to teach computer science, robotics, or computational thinking skills. Few preservice elementary teachers have the ability to take computer science courses in their preparation programs (Margolis, Ryoo, and Goode, 2017). One study found that a majority of computer science professional development was designed for high school teachers. There were 12 studies that focused on high school teachers, four for both middle and high school teachers, two for only middle school teachers, and three for high school, middle school, and elementary teacher combined, and none found for only elementary school students (Menekse, 2015). Gomez, Lee, & Berkhoudt Woodman (2022) discussed how, “As a field we essentially ignore the teachers' existing professional assets, including their pedagogical beliefs, goals, and practices that could be usefully leveraged in CS education. We need to better understand how practicing teachers learn about CS teaching and learning, and how existing teacher pedagogy can be leveraged as they build their CS educational practices.” Teachers are equipped with best pedagogical practices for teaching, but further research around incorporating best practices into computer science instruction is worthy of examination.

One study found that professional ecosystems strengthened individual communities’ effectiveness by providing participant access and support and in the combination that best supports their professional learning. These ecosystems include computer science communities, organizations, and educational institutions working together (Falkner et al., 2018). If teachers can be properly trained and have continual access to support, more could be inclined to include computer science instruction in their classrooms. As more students have access, students from diverse backgrounds may garner interest in the subject matter. Access is elemental, as many students currently do not have either. The earlier access to personally and professionally authentic computer science experiences is provided, the sooner and easier it will become for girls and students from underrepresented populations to engage and gain confidence in computer science.

Studies have focused on integrating computing into core subject areas. Teachers who were enthusiastic about integrating computer science into their teaching could do so with more integrity over time with professional development (Ketelhut et al., 2019). In addition, teachers hold diverse views on the concept of computational thinking and vary in their instructional resources (Garvin et al., 2019). Israel & Lash (2019) found that when teachers integrated computational thinking with mathematics lessons, concepts increased in complexity across grades, mathematics instruction was the focus, while the computational thinking aspect was secondary, and three types of lessons were taught: no integration, partial integration, and full integration. Math content remained the priority for instruction, while computational thinking was a secondary priority. Teachers who integrated computational thinking with core content areas in elementary schools showed interest in continued integration, and expressed exploring other opportunities to integrate computational thinking into their future teaching (Waterman et al., 2020). Teachers who are given training and support are willing to incorporate computing into their teaching. While teachers may be willing to learn, inequities still persist.

Inequities in computing instruction occur at the elementary school level. One study showed that some teachers delivered computational thinking instruction that did not always provide equal access to all students. In this study, researchers found that without explicit support, teachers implemented computer science instruction in ways that perpetuate current trends of inequity, and found that teachers would benefit from explicit training to support underrepresented students. The main example of inequity included teachers offering computing experiences as lesson extensions to above-average students (Coenraad et al., 2020). Educators would benefit from training and exposure to Culturally Responsive Computing practices.

Culturally Responsive Computing Pedagogy

Teachers are slowly becoming required to provide computer science instruction at elementary level, yet many are unprepared. If the goal of providing computer science instruction at younger ages is to garner and sustain interest to fill the pipeline with more candidates, and to empower historically marginalized students with transformative computer science skills, providing meaningful and relevant computer science opportunities may be one way to increase interest and participation of underrepresented students. According to Scott, Sheridan, and Clark, (2015), in a review of over 50 computing programs, the vast majority focus on technical skills, and none mention issues of diversity, community, culture, or identity. Culturally Responsive Pedagogy (CRP) and Culturally Responsive Computing (CRC) provide frameworks for educators to guide their teaching, and their students' learning.

Culturally responsive teaching leverages students' strengths to serve as an anchor in the process of transforming traditional teaching practices that have oppressive histories (Gay, 2010). This allows students' to participate in a learning process that is student-centered and based on the needs of the students (Gay, 2010). Past studies have shown that CRP can be incorporated into computer science education (McLoughlin, 1999) and how it can impact learning outcomes. Eglash et al. (2013)'s study showed the positive results of utilizing CRP with computing education by focusing on: "(1) self-identity as neither static nor predetermined; (2) motivating and improving science, technology, engineering, and math that connects directly to students' cultural knowledge and identities; and (3) supporting youth in being producers with technology." CRC is closely related to CRP, as it draws upon the extensive work of CRP (Scott, Sheridan, and Clark, 2015). Culturally responsive pedagogical strategies can be applied to make technology education accessible to underrepresented populations by using asset-based approaches (Scott, Sheridan, and Clark, 2015). CRP is a pedagogical strategy that engages diverse youth and stands in stark contrast to traditional deficit models of thinking that fault students' personhood, communities, backgrounds, and families (Scott, Sheridan, and Clark, 2015). CRP focuses on these factors as assets, and teachers generally employ CRP by using asset building approaches that include opportunities for reflection and connection (Scott, Sheridan, and Clark, 2015).

According to Scott, Sheridan, and Clark, (2015), culturally competent educators, "develop and openly demonstrate their own cultural competency about students' identities, use this knowledge as the foundation on which to build lessons, develop meaningful and sustainable relationships with students predicated on the notion that they will succeed, and maintain a heightened sensitivity to the school's sociopolitical context as a place that can emancipate or oppress". Culturally responsive educators connect to their students in non-traditional ways (Scott, Sheridan, and Clark, 2015). Studies have suggested that CRP can improve Black students' self-image and self concept that are diminished in dominant culture (Ladson-Billings, 1994). CRP, and it's computing-related framework, CRC, offer an approach to provide inclusive and meaningful computer science opportunities for underrepresented students.

The goals of CRC include the tenets of CRP but apply them to technology education. The framework examines how technology education can reflect and encourage reflection on students' complex and intersectional culture (Scott, Sheridan, and Clark, 2015). The focus of CRC includes:

1. "Motivate and improve science, technology, engineering, and math (STEM) learning experiences;
2. Provide a deeper understanding of heritage and vernacular culture, empowerment for social critique, and appreciation for cultural diversity;
3. Bring 1 and 2 together: to diminish the separation between the worlds of culture and STEM" (Scott, Sheridan, and Clark, 2015)
4. And, "...This technology must not only respond to these identity issues, but also satisfy pedagogical demands of the curriculum" (Eglash et al. 2013).

The goals of CRC are framed to address the persistent gap in historically marginalized groups' participation and access to technology to increase the focus on technology education and the workforce, and focuses on teaching computing in ways that allows creation to be transformative. According to Scott, Sheridan, and Clark (2015), "...technological success should consider who creates, for whom, and to what ends rather than who endures socially and culturally irrelevant curriculum." Participation of underrepresented minorities in computer science provides opportunities for these groups to create in transformative ways, and participate in technology creation that includes these groups' backgrounds, histories, contexts, and needs. CRC teaching strategies can include peer-teaching, participatory tasks, clearly stated outcomes, and discourse involving problem-based activities and include four design principles: prior knowledge, cultural ways of knowing, engagement and motivation, and civic and social empowerment (Scott, Sheridan, and Clark, 2015). In addition, CRC emphasizes educators' needs to reflect on their own identities, cultural backgrounds, and motivations (Scott, Sheridan, and Clark, 2015). By offering opportunities for educators to integrate CRP and CRC into their teaching practices, students may have more long term sustainment and interest in computer science opportunities.

Conceptual Framework

The Kapor Center's Framework for Culturally Responsive-Sustaining Computer Science Education served as the conceptual framework for this study. Appendix C shows the framework, and this framework was used because of the intentional focus on culturally relevant-sustaining pedagogy. The framework states:

> Culturally responsive-sustaining computer science pedagogy ensures that students' interests, identities, and cultures are embraced and validated, students develop knowledge of computing content and its utility in the world, strong CS identities are developed, and students engage in larger socio-political critiques about technology's purpose, potential, and impact.

Due to the study's focus on equity and strategies to promote equity, this framework served as the conceptual framework. The framework provided the foundation for the interview protocol and thematic coding in the data analysis. For the interview protocol, the core components of the framework were utilized to initiate discussions around teaching strategies utilized to increase equity. Then, the framework was used to create the a priori codes for the strategies that teachers articulated. This framework served as the primary conceptual framework for the study.

In summary, the United States is facing a crisis in which the number of projected computing roles quickly outpaces the number of graduates to fill them. This economic and educational concern could be addressed by providing more students, specifically girls and underrepresented minorities, access to computing experiences in elementary school settings. Studies have shown that girls and underrepresented students are not represented in STEM career fields, or in the educational pipeline. One way to address this is to provide professionally and personally authentic experiences so that these students persist in their interest and participation in computing. By incorporating Culturally Relevant Teaching and Culturally Relevant Computing practices, teachers provide girls and underrepresented students of color the opportunity to positively identify with computer science. This literature review has highlighted the growing body of research around personally authentic computing experiences for students in elementary school settings, with the majority of research focused on professional skills and knowledge outcomes.

CHAPTER 3: RESEARCH DESIGN

Introduction

The goal of my study was to understand how teachers view equity in relation to their goals for computer science instruction, and what strategies they articulate and implement to make their practice equitable. The need for this study stems from our current limited computing pipeline and the need to empower marginalized populations with transformative computing skills. Furthermore, the computing field is not diverse, and lacks representation by women and people from historically underrepresented populations. Computer Science is becoming more and more relevant in high school and middle school settings. If students are exposed to computing opportunities in elementary school, there might be more opportunities for girls and students of color to pursue computing careers, and allow them to utilize technology to uplift their own lives, communities, and world.

Research Questions

1. How do elementary CS teachers think about equity in relation to their goals for their computer science instruction?

2. What strategies do elementary computer science teachers articulate to make their practice equitable?

Research Design and Rationale

This study was a qualitative study using an in-depth interview design from the perspective of teachers implementing computer science in elementary classrooms. Respondents participated in two interviews. Prior to being interviewed, the respondents provided a lesson plan document, which they walked through and reflected on with the interviewer. According to Creswell and Creswell (2018), a qualitative approach is used to seek meaning people give to a problem. The focus of this study is to explore how teachers implement computer science in the elementary classroom, and the ways in which they think about equitable practices in relation to their computer science instructional goals. This study seeks to understand how teachers implement and articulate strategies to make their practice equitable. The study seeks to understand the participants' views and experiences and focuses on direct experiences of the participants in order to answer how elementary school teachers think about and implement equitable computer science instruction. Though it is possible to survey teachers, that would not provide in-depth data regarding how and why teachers made certain decisions around the equitable implementation of computer science instruction.

Methods

Site and Population Selection

My study included any elementary school teachers actively teaching and implementing computer science in the United States. In the United States, elementary schools can vary in ranges from kindergarten - sixth grade. Teachers had to be teaching in an elementary school setting, and actively implementing computer science instruction. Since computer science is not a mandated subject, and there are no comprehensive training programs for teaching computer science, any teacher that self-identifies qualified.

Data Collection Methods

This study utilized interviews and documents to explore computer science implementation at the elementary school level. This study used an opportunity sample. I began by interviewing participants in my existing professional and personal network and utilized snowball sampling. As part of my recruitment strategy, I offered the opportunity to participate in my study by posting in various online communities, such as the community boards for a national professional network of computer science teachers. I ended the interview by asking participating teachers if they knew of other elementary teachers that teach computer science who might be interested in participating in the study. I continued to ask for snowball referrals until I reached the 20 participant sample size for the study. Once 20 teachers completed the interviews, I stopped recruiting for the study.

I interviewed a variety of teachers based on their experience and comfort level with computer science instruction. This way, the study explored how any elementary teacher that teaches computer science considers and implements equitable practices in their computer science teaching practices. I interviewed 20 teachers, and once 20 teachers completed interviews, I stopped recruiting and accepting respondents for the study .

Table 3.1 Respondent demographics summary

	Job Title	US Region	School Type	Gende r
R1	3rd Grade Teacher	South	Public	F
R2	1st-5th Grade Enrichment Teacher	Northeast	Public	F
R3	Instructional Technology Coach	West	Public Charter	F
R4	Technology Teacher	Northeast	Private	M
R5	STEAM Director and Teacher	Northeast	Private	F
R6	K-5 STEM Teacher	Midwest	Public	F
R7	STEM Teacher	Midwest	Public	F
R8	Technology Teacher	Northeast	Private	M
R9	2nd Grade Teacher	West	Public	F
R10	K-7 CS Teacher	Northeast	Public Charter	F
R11	K-8 CS Teacher	West	Private	M
R12	K-4 Technology Facilitator	Northeast	Public	F
R13	3rd Grade Teacher	West	Public	F
R14	Technology Teacher	Midwest	Public	M
R15	K-8 STEAM Teacher	South	Private	M
R16	First Grade Teacher	West	Public	F
R17	6th grade Science Teacher	South	Public	M
R18	K-6 CS Teacher	Northeast	Public	M
R19	Makerspacer/STEAM coordinator	West	Private	M
R20	4th Grade Teacher	West	Public	F

Semi-structured Interviews

Respondent teachers engaged in two parts of the study: two 45-60 minute long interviews. Thirteen respondents gave concise responses, and answered all interview protocol questions in the first interview session, such that a second interview was not needed. The first part of the study consisted of an interview and requested teachers to provide a lesson plan. They then reflected on their lesson plan and teaching practices. The lesson plan was used as a guide and reference for the interview questions, which focused on respondents' goals for computer science instruction.

In the second interview, core components of the Kapor Center's Reimagining Equitable Computer Science Framework were provided to participants to reflect and comment on in relation to their lesson plan and teaching practices. Each teacher interviewed was asked, "How do you think about equity in your computer science teaching? What does equity mean to you?, and then they were asked if or how they might incorporate each of the six core components from the Kapor Center's Framework for Culturally Responsive-Sustaining Computer Science Education. Each core component was displayed on an individual slide, and was read aloud in the interview. Respondents were asked to reflect on the core components and asked, "What resonates with you?" before explicitly being asked if and how they incorporate each core component into their computer science teaching practices. For the first three interviews, the first core component, anti-racist practices, was shown first. Respondents showed guarded and timid responses, since the topic was heavy and difficult to discuss. After these interviews, I moved the anti-racist practices to the last core component I asked about, to build rapport and comfort during the interview before diving into a challenging topic. This allowed respondents to share more in the beginning of the interview, rather than provide guarded responses.

All of the interviews were semi-structured in an effort to create a dialogue and exploratory discussion, as opposed to a rigid question and answer format. The questions asked pertained to the process of implementing computer science, goals for computer science instruction, strategies they implement in the classroom and their thoughts about equity. Interviews lasted approximately an hour, and were hosted and recorded on Zoom. I was the sole interviewer and took notes on the responses of my participants. Both interviews occurred within a two-week period of one another.

Lesson Review During the Interview

Before the initial interview, each teacher was asked to provide one computer science lesson plan that they felt was successful in their classrooms. Because the study focuses on equity, a lesson that a teacher deemed successful would speak to the interests and engagement of students in computer science. During the first interview, they discussed their lesson plan and provided thoughts and reflections on their instructional goals and how these goals relate to equity. Respondents began by answering questions about their location, students, and classrooms. Then, respondents were asked questions about their goals in relation to their lesson plan. The goal of the first interview was to understand what teachers were trying to accomplish in their computer science lessons, and what part equity played in their instruction.

In the second interview, respondents revisited the same lesson plan, and were asked to read the six core components of the Kapor Center's Reimagining Computer Science Education Framework. Participants were told that this framework has been put together by a computer science organization that has promoted computer science education for many years, but this study purposefully examined their thoughts of the framework and how they did or did not apply elements to their computer science teaching practices, and if the framework pushed them to think differently about how they teach computer science. They were prompted to share their thoughts as to how each core component could be applied to change and enhance equity in their existing lesson plan. As the participant read a core component of the framework, they were asked if the main idea of the core component could be addressed in the lesson plan. This allowed participants to respond with adjustments and changes to their lesson plan, or express that they would not incorporate the idea into their teaching practices.

Data Analysis Methods

The audio and video recordings of the zoom interviews were reviewed before transcription. Then, I transcribed the interviews using otter.ai. While reviewing the transcripts, notes were taken and organized by theme in a spreadsheet to start tracking potential categories and repeating themes. A priori coding was utilized to help finalize the categories. Table 3.1 displays potential responses from participants and how responses were coded. Transcripts and recordings were reviewed again to be coded again using MaxQDA. Codes were broken into three categories: teachers' views on equity, goals of computer science instruction, and strategies teachers articulated.

The NASEM report was used to code for teachers' ideas around equity (NASEM, 2021b). It became apparent that while teachers spoke about access as opportunity, there was one theme that emerged that was not referenced in the NASEM report: equity as differentiation (NASEM, 2021b). The report defines four buckets of equity in computer science instruction:

1. Equity as access and opportunity,
2. Equity as achievement and/or positive identification with computer science as a discipline,
3. Equity as expanding what constitutes computer science in the form of integrating alternative cultural perspectives, and
4. Equity as including computer science as part of justice movements (NASEM, 2021b)

The language I looked for in transcripts led me to code one of the variations included describing allowing all students to have opportunities to learn computer science, how teachers wanted kids to have fun and engage in computer science, and how teachers described ensuring they differentiated instruction so that all students could participate in computer science opportunities.

Since this was an emergent theme based on responses, I created a separate theme to reflect the data in the responses. The themes for goals were derived from the answers to the question, "What do you hope your students get out of computer science this year?" and "What are your goals for your computer science teaching?" These themes were coded based on teachers responses and using language to describe how they wanted all students to have the opportunity to engage in computer science, how they wanted students to enjoy computer science opportunities, how they wanted to empower students with skills and ability, and how they wanted to increase academic achievement. The interviews and lesson plans were reviewed and coded for references to equity. Specifically, responses and annotations were examined for how teachers view equity as defined in their teaching practices. These views can include: Equity as access and opportunity, equity as achievement and/or positive identification with computer science as a discipline, equity as expanding what constitutes computer science in the form of integrating alternative cultural perspectives, and equity as including computer science as part of justice movements (NASEM, 2021). In addition, the Kapor Center (2001) identifies specific courses of actions that reflect each core element of equity. Participants' responses were coded to fall into these six categories of equity:

1. Anti-racist teaching practices,
2. Identity as tools for inclusion and equity,
3. Rigorous and authentic pedagogy and curriculum,
4. Inclusion of student agency, voice, and self determination in computer science instruction,
5. Inclusion of families, communities, assets, and culture in curriculum, classrooms and learning opportunities,
6. Incorporation and identification of a diverse variety of experts as role models

After conducting the interviews, it became clear that teachers in the sample fell into 2 categories:

1. Equity as Opportunity
2. Equity as Differentiation

Since the responses fell into these two codes, the remaining categories were removed from the data analysis. Table 3.1 includes how responses were coded to reflect the two themes in equity. I captured the variability in how teachers think about equity by coding and categorizing participant's responses, lesson/unit plans, and annotations.

The themes for the strategies articulated were pulled from the Kapor Center Framework for Culturally Responsive-Sustaining Computer Science Education. These themes were pulled from the core components of the framework: acknowledging racism in CS and enacting anti-racist practices, creating inclusive and equitable classroom environments, pedagogy and curriculum are rigorous, relevant, and encourage sociopolitical critiques, student voice, agency, and self-determination are prioritized in CS classrooms, family and community cultural assets are incorporated into CS classrooms, and diverse professionals and role models provide exposure to a range of CS/tech careers. Appendix C displays this framework. Responses were cross-referenced with the core components and course of action in the framework, and these themes were created.

Table 3.2 Thematic codes derived from teachers' responses

Label	Definition
Equity as Opportunity	Any instance when a respondent discussed opportunity or access as equity.
Equity as Differentiation	Any instance when a respondent discussed meeting different needs of learners as equity.
Providing Access to Opportunity	Any instance when a respondent discussed exposure to CS, awareness of CS in students' everyday life, and/or providing an opportunity to learn the basic CS concepts and skills to build foundational knowledge.

Promoting Positive ID with CS	Any instance when a respondent discussed student enjoyment, accomplishment, engagement, excitement, confidence, or a sense of belonging with CS activities.
Empowering Students to Use CS for Social Change	Any instance a respondent connected their CS instruction to underrepresented populations, and any instances when respondents describe instruction as solving problems or improving the lives of marginalized groups.
Increasing Academic Achievement	Any instance when a respondent discussed academic achievement as a goal not related to understanding computer science per say.
Fostering an Inclusive Classroom Environment	Any instance when a respondent discussed anything they do to include all students in the classroom community. This could include collaboration, scaffolding, family life, cultural backgrounds, efforts to include students based on traits, gender, and academic ability.
Incorporating Rigorous Pedagogy	Any instance when a respondent discussed having high expectations, promoting and supporting persistence, thinking critically to find computational solutions, aligning to standards or state-mandated assessments.
Incorporating Student Agency & Voice	Any instance when a respondent discussed student voice, self-determination, creativity, choice, personalization, reflection, planning, freedom to explore, options, independence, and facilitating learning rather than teaching.
Creating Cross-Curricular Connections	Any instance where a respondent discussed connecting computer science instruction to other academic content areas.
Addressing Racism in CS	Any instance when a respondent discussed actively teaching to combat racism and actively working against racism.
Highlighting Professional Relevance	Any instance when a respondent discussed exposing students to real life careers, application, and professionals in CS.
Modeling Professional Inclusiveness	Any instance when a respondent spoke about exposing students underrepresented populations in the professional field.

Incorporating Relevant Pedagogy	Any instance when a respondent asserted they make CS more relevant for their students.

Positionality

In my professional role, I have made connections with district technology leaders across the country. I have connected and formed relationships with Ozobot Certified Educators, an educator ambassador program designed to support computer science teachers in implementation. There are over 200 teachers enrolled in the program, but fewer than 30 are active participants. I also reached out to my greater network to recruit other computer science teachers that use other platforms for implementation. I built relationships with teaching teams as some of these participants did not know me, or could have confused my role as a researcher with my role working for a for-profit company.

Since I had existing relationships with the teachers, schools, or principals, I positioned myself as a UCLA graduate student. They may or may not have known me personally, or professionally outside the context of this study. I kept my study voluntary and confidential to ensure that no participant felt pressured to participate and to ease any concerns of sharing negative feedback or comments with district or school administrators. For those that did not know me, I hoped that being an unfamiliar party would help increase participants' candidness. If participants did recognize me as an employee of Ozobot, I reassured them that my focus was not on our product, but as an educator looking to support teachers in computer science implementation.

Ethical Issues

The only ethical concern could be if the teachers I interview were aware of my role at Ozobot. If they were familiar with the product or used it regularly, they may have felt inclined to not be as transparent about how they utilize the product in their teaching. This was not an issue at all, as being familiar with Ozobot is not a requirement to participate in the study. I did not mention or refer to Ozobot unless the participant brought it up. If the respondents talked about my connection to Ozobot, I politely reminded them that this study is outside of my role and responsibilities at Ozobot. If the teachers do use Ozobot, they may recognize me from the instructional video content that we produce, as I provide on-camera instruction within the videos, but I will not be offering this information.

Participants could have been hesitant to share any negative information or opinions as they may not have wanted to be identified by their administrators, so I reminded participants that participation was completely anonymous and voluntary. Participants verbally consented to participating in the study and no identifying information was shared. All school sites were anonymous and I used pseudonyms for all participants and kept all data on a personal device that was password protected. Upon completion of my study, I provided all participants with a copy of the study and deleted all transcripts.

Credibility and Trustworthiness

If teachers recognized me and viewed me as a representative of Ozobot, they may have skewed their responses in a more positive and favorable manner in regards to the

Ozobot product. If they brought this up, I presented myself as a researcher first and explained my purpose. Explaining my study garnered trust, generated buy-in, and focused the conversation around computer science teaching in general, and not specifically teaching with Ozobot. If they did focus heavily on Ozobot, I adjusted my line of questioning to incorporate other areas and themes of computer science instruction.

Another potential threat to my study was my own personal bias. My identification as an Asian-American woman and knowing that I belong to an underrepresented minority group may have impacted my interactions with teachers. Most participants knew that I was a former computer science teacher and worked in the computer science education field. As someone who believes in the power and importance of computer science instruction, I could have potentially displayed that bias in a line of questioning or part of the conversation. This may have skewed my results and interview responses. It was essential for me to remain neutral as a researcher, and not allow my personal feelings or opinions to impact the interview or analysis of my data. Before beginning the interview, I shared the purpose of my study with all participants in order to alleviate this bias. In order to mitigate bias, I collected rich data and transcripts using direct quotes from interviewees. I posed neutral questions and used standardized protocols and coding procedures to ensure systematic data analysis.

Study Limitations

Since computer science is not a mandated subject for elementary school students, the findings from the small sample size of the study may not be generalizable to the general population of elementary school teachers. Since my study relied on a snowball sample, the data may be skewed by those that are enthusiastic about computer science education, rather than reflect the general population of teachers.

Conclusion

This qualitative study explored how teachers' goals for computer science instruction relate to equity and the strategies they articulate to make their practice equitable by utilizing semi-structured interviews with an examination of lesson plans. Teachers that self-identified as teachers of computer science qualified for the study, and 20 respondents were recruited. Respondents were asked to provide one lesson plan that they felt was most successful and they reflected on this lesson plan in two interviews. Interview responses and documents were coded and analyzed to form the findings of the study.

CHAPTER 4: FINDINGS

Introduction

This chapter reports the findings of a qualitative research study that focuses on 20 elementary school teachers who teach computer science in their classrooms. This chapter will review the principal findings, and examine teacher's interpretations of equity, goals for computer science instruction, and the strategies they articulated they implemented to make their practice more equitable. I sought to address the following questions:

1. How do elementary CS teachers think about equity in relation to their goals for their computer science instruction?

2. What strategies do elementary computer science teachers articulate to make their practice equitable?

Principal Findings

The qualitative analysis conducted in my study revealed the relationships between equity and teacher's goals for computer science instruction, and strategies that teachers implemented to promote equity in their computer science teaching. Analysis of the data produced principal findings in response to my original research questions. First, respondents interpreted equity to mean opportunity and/or differentiation. Second, respondents' goals for computer science teaching included four themes. Third, in order for respondents to reach these goals, they articulated various strategies they implemented in their classrooms. The themes expressed are summarized in Table 4.1, with each check mark representing the respondent expressing each theme at least once in both interviews. A dash indicates that the theme was not mentioned whatsoever in either interview. Thematic definitions are summarized in Table 3.1. In the following sections, I will discuss my findings in greater depth. In what follows, individual respondents will be identified by the identifier presented in the table (e.g., R1, R2, etc). When respondents are quoted, commas indicate pauses, dashes indicate self-interruptions, and elision represents omissions of short pieces of transcript that do not alter the meaning of presented quotes. Occasional words [in brackets] have been inserted to clarify respondents' talk.

Table 4.1 Themes mentioned by participating teachers in the categories of equity, instructional goals, and strategies. Themes in each category are sorted in descending order of the number of teachers expressing each theme.

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16	R17	R18	R19	R20	SUM
Teachers' Views on Equity																					
Equity as Opportunity	✓	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	17
Equity as Differentiation	-	✓	-	-	✓	-	-	✓	✓	-	✓	✓	-	-	✓	✓	-	-	-	-	8
Teachers' Goals for Comptuer Science Instruction																					
Providing Access to Opportunity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	20
Promoting Positive ID with CS	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	20
Empowering Students to Use CS for Social Change	-	-	-	✓	-	-	-	✓	-	✓	✓	✓	-	✓	-	-	-	-	✓	✓	8
Increasing Academic Achievement	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	1
Strategies Articulated by Teachers																					
Fostering an Inclusive Classroom Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	20
Incorporating Rigorous Pedagogy	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	20
Incorporating Student Agency & Voice	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	20
Creating Cross-Curricular Connections	✓	✓	-	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	17
Addressing Racism in CS	✓	✓	-	✓	✓	✓	-	-	✓	✓	✓	✓	-	✓	✓	-	-	✓	✓	✓	14
Highlighting Professional Relevance	✓	✓	✓	✓	✓	✓	✓	-	✓	-	✓	✓	-	✓	-	-	✓	✓	-	✓	14
Modeling Professional Inclusiveness	✓	-	-	✓	-	✓	✓	✓	-	✓	-	✓	✓	-	✓	✓	-	✓	✓	✓	13
Incorporating Relevant Pedagogy	-	✓	-	-	-	-	-	-	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12

Teachers' Interpretations of Equity

All respondents spoke about equity in two ways: access to opportunities and differentiating their teaching to reach all students. All respondents spoke about equity in one of two ways. 17 of the 20 respondents described equity as access to opportunities in computer science. Eight respondents described equity in terms of differentiating instruction for their students. Five respondents spoke about equity in both of these ways.

Equity as Opportunity: Equal Access, Encouraging Female Students, and Representation

The most common interpretation of equity was thinking of equity as opportunity in computer science. Of the 17 respondents that discussed their thoughts about equity and linking them to the idea of providing opportunity, their thoughts fell into three categories: providing equal access to all students (11 of 17), encouraging female students (3), and equity as representation (3).

Respondents who defined equity as providing equal access for all students communicated that they recognized that computer science education is not ubiquitous and opportunities for all students to engage was limited. R3 stated, “I think everyone should be exposed to computer science. It's like everywhere, right? But not everybody has equal access to it.” R11 supported this paradigm. They also acknowledged their belief that not all students have access and stated, “[Equity is] giving every student what they need to have opportunities in life that they otherwise might not have had, if we didn't give them those things.” Respondents recognized that exposure and access were vital for students, yet not all students have access. These teachers identified that computer science is relevant for their students, and expressed belief that all students should have opportunities to engage with the content.

A small group of respondents expressed opportunity equity as encouraging female students. These teachers identified that girls are underrepresented in the professional field, and acknowledged the relationship between providing opportunity for all students and encouraging female students to engage with the content. R13 spoke to their belief about the importance of identifying the gender gap by stating, "...we are doing a better job at opening these opportunities up for girls, and being more aware of that."

Of the respondents that defined equity as opportunity, a subset of three respondents spoke about representation in computer science. These teachers found value in highlighting diversity in the field so that students could see themselves represented in what they were learning. R5 stated,

> [I] try to have more cultural examples when we talk about people …Not only just the white man, not even just the white women, but diverse groups from different parts of the country, I mean, different parts of the world. Because many cultures have roots in computational thinking and computer science, it's just that certain places moved ahead a little faster. But there are many examples of computational thinking across the globe … that is equitable, in many ways, and gives everyone a place to identify and see themselves in it.

This respondent recognized the importance of exposing students to different cultures and perspectives as part of their computer science instruction.

Equity as Differentiation

Eight respondents spoke about equity in terms of differentiating instruction and their descriptions sounded similar. These respondents identified that there were different needs from different learners in their classroom, and verbalized equity as meeting the various needs of learners through their teaching. For example, R12 stated, "It means that everyone's getting what they need to get across the board in order to make them successful. …that definitely looks different for different people.” R8 explained this in terms of diversity, “I think it's about individual needs for diverse students. We really just boil it down to, ‘What does differentiation look like in my classroom?’” R5 pushed this idea further by discussing how differentiation can apply to student interest and strengths. They stated:

> I do try to seek opportunities to address different types of learners that we have. We have some who love math and love numbers, and we have some that love arts. I feel like the design should not be only for those who are good in science and math, because you do need diverse thinkers across the board to develop a good product. And so focusing on it from one angle, either the mathematical part of it, or the science part of it, you tend to lose a few who might not be able to keep up or who find that's not interesting. So you find applications that they may see in say, arts or performing arts or visual arts, so finding different avenues for them to see application, and practice any skill that they've learned. That's how I see equity. (R5)

For this teacher in particular, student strengths and interests tied directly into differentiation of learning. For other teachers, equity was viewed as providing opportunities for students to be successful with the content. Overall, teachers viewed equity as the opportunity to engage in the content and as differentiation to reach all students.These views were closely linked to their goals for computer science instruction.

Teachers’ Goals for Computer Science Instruction

All 20 respondents stated that providing access to opportunity and promoting positive identification with computer science were goals for their instruction. Eight respondents discussed empowering students to use computer science for social change as a goal, and one teacher discussed increasing academic achievement as a goal for computer science instruction.

Providing Access to Opportunities

All respondents stated that their goals were to ensure that students had access to engage with and learn about computer science. Respondents discussed the importance of giving students opportunities to acquire future-ready skills, connecting computer science to students' everyday lives, and providing exposure to basic computational skills. Preparing students for their futures and connecting instruction to students' everyday lives was a recurring theme of many respondents. R14 stated:

> I want kids prepared for the future. I want them to be aware of what opportunities are out there for them that they could easily grasp and make a great life for themselves, no matter where they live or where they're from. So I try to expose them to as much as possible career wise and try to make it very practical, like, what things are actually used for in the real world? That way they can understand why we're learning what we are. (R14)

In addition to access to opportunities, teaching foundational skills was articulated as a main goal and priority of respondents. R7 spoke about the importance of building a foundation for student understanding of computer science, and how students' skills could grow throughout their elementary school careers. The respondent stated:

> It's nice that I work with students kindergarten through fifth grade, because I can provide and know what foundation they've been given. So I can build on that each year. And now that I'm in my sixth year, so my fifth graders, I was with them as kindergarteners. So it's really cool to see what they've learned with me in kindergarten, and how they started off using simple block based coding, to train, build and write in JavaScript, Python, and really expand and grow. (R7)

R7 saw growth as a direct outcome of opportunity. R12 further supported this viewpoint by stating, "I think the biggest thing that I want to do with my students is to expose them to different components of computer science." Respondents articulated goals of setting foundational knowledge of computer science and how it tied directly to their thoughts about preparing their students for the future. Preparing students for the future was a goal because these respondents talked about foundational skills as an important outcome of opportunity. R9 spoke about the relationship between access to opportunity and positive identification with computer science. They stated, “ and I think hav[ing] students being able to see that it's not only accessible, but it's something that they might enjoy doing. And I think this is a core component in that end goal.” Access to opportunity directly tied to all respondents’ views of positive identification as goals for their computer science teaching.

Positive Identification with Computer Science

All respondents stated that positive identification with computer science was a goal for their computer science instruction, yet they offered a variety of means or targets of such identification.

Enjoyment and Excitement

18 of 20 respondents stated that enjoyment of and excitement around computer science was a goal for their computer science instruction. 15 out of 20 respondents stated that engagement was a goal of computer science instruction. Five respondents stated that increasing student interest was a goal. Five respondents mentioned building student confidence as a goal. Three respondents listed increasing representation of underrepresented populations as a goal, and two respondents stated empowerment of students as a goal. Most respondents spoke about enjoyment and excitement around computer science as a goal for their students.

Respondents gave various examples of how enjoyment and excitement around computer science was a goal. R9 stated, “I hope that they just enjoy it. I think…they can say that [what] they learned in second grade was how to code a robot or how to create their own video game on Scratch or something like that.” The goals of providing students access to basic computer science skills linked closely to enjoyment of the content. The same respondent gave an example of how enjoyment also tied directly to engagement with computer science content. R9 said, “I mean, they enjoyed it… [and] some students really ran with it.” Others expressed student enjoyment and excitement around sharing what they learned and created. R7 said:

> Oh, they love it. I mean, they love when they're finished and, and sharing their app with a friend and their friend gets to play their app and how they take that ownership and [say,] “How cool is that I created this?” They enjoy the feedback from other kids [saying things] like, “I liked how you use this or how you did this.” And some of these kids surprise me, and they teach me things too. So they enjoy it. (R7)

Others articulated how focusing on fun deepened student engagement and investment in their own learning. R3 said:

> I think they just thoroughly enjoyed the lesson because it had music, but they had fun doing it and working together and finding [that] all their hard work throughout the year paid off because they had something to show for it and a final way. And by far, I think this was like the lesson they love the most. (R3)

Engagement with computer science was closely tied to enjoyment and excitement, and respondents spoke to this idea.

Engagement

In this study, all respondents stated that one of their goals was to promote positive identification with computer science. Another way respondents spoke about this was in relation to engagement. R2 stated, “My students are really interested in Minecraft so it's like an engaging topic for them. So, I feel like if you're having a good time and learning, that's like really setting you up to really bring that knowledge into your brain.” The relationship between positive identification, enjoyment, and engagement are deeply intertwined. Respondents spoke about leveraging these goals to promote positive identification for their students. R5 shared:

> In every class, I try to get at least 70 to 80% engagement. When I mean, engagement is like, they'll all do it. But I want them to dig deeper. And so that's what I'm going in for. So when they are engaged and asking those questions and trying to make it on their own, or trying to do that on their own, that's a win. (R5)

This respondent found value in having students engaged in the material, but also articulated a goal for students to engage deeply in their learning.

Confidence

Five respondents spoke about how confidence building was a component of promoting positive identification with computer science. R18 stated they wanted to give students “...the opportunity to build confidence within that domain to know that they can do it.” R1 elaborated on this idea by saying:

> ...confidence building is when those kids that might not necessarily excel in other academic areas, but they figured out how to make this [and] they figured out how to maneuver this. …It just gives them a different type of opportunity

> to find a strength, you know, to find something that they are good at, or that they are the ones that [can] contribute in this setting. (R1)

These educators stated the value of confidence building in their students, and explicitly stated it as a goal to promote positive identification with computer science. Some respondents tied confidence building to the idea of promoting student interest as a goal.

Student Interest

Five respondents spoke about promoting student interest in computer science as a goal. R14 said:

> At least one topic that we're working on… they really liked. And they went all out with it, and they're really proud of their work. And so, you know, it kind of gives me hope that they could find that niche and.... So that's how I just tried to expose them to as much as possible, to give them ideas…so I tell them, You know what, no matter what your, your parents do, or your neighbors do, if you know that's something that you like, by all means go for it. But if you're not interested, you find what you are [interested in], and you go for it. (R14)

This respondent articulated promoting student interest to expose students to potential careers. R19 also articulated that their goal was to provide students with the opportunity to gain interest in the content. They stated, “I want some of the kids who might normally be like, ‘No, that's not for me’ to be like, ‘Okay, maybe there is a spot where I could be interested in that.’”

Representation

Three respondents spoke about promoting representation as a goal. This might be along racial/ethnic lines, as discussed by R10, who teaches primarily low-income Black and Latinx students,

> ...But I think when we provide every single one of our kids (a computer science education), we are telling every single one of our kids that they can be a computer scientist, that they can think computationally. And I think that's something I tried to reinforce to my kids, every single one of them is capable of doing this, and that they are needed in this work. (R10)

R19 saw representation as a gender issue, saying they wanted their female students to “ see themselves in computer science. I'm very passionate about closing the gender gap in computing careers.” R2 supported this thought and stated, “...[my students] are computer scientists. That's how I refer to it. If you are using algorithms and computers to make programs for people, you are computer scientists. So that's what I tell them.” These respondents valued promoting the idea that all their students were considered computer scientists if they were engaging with the materials and learning the content.

Empowering Students to Use Computer Science for Social Change

Eight teachers mentioned a goal of empowering students to use what they learned in computer science to address social issues of their generation. Their ideas fell into four different categories: social activism, improving the lives of underrepresented populations, environmental activism, and student empowerment.

Social Activism

Four respondents mentioned social activism as a goal for their computer science instruction and gave details about how they focused on this in their instruction. These respondents focused on various issues, such as linking learning about the Civil Rights Movement and comparing it to the modern Black Lives Matter Movement. R8 spoke about exposing students to issues like the digital divide. They stated:

> [We have] conversations about digital divide, who has access to technology and why? So they talk about who can afford it. [We] also talk about rural communities simply just don't have it period. Or if they do, it's extremely expensive. …So… that's touching on: why are we doing what we're doing? (R8)

This respondent tied the idea of computer science education to why their students were learning the content.

Improving the Lives of Underrepresented Populations

Three respondents spoke about the importance of applying computer science learning to improving the lives of underrepresented populations. One prime example of this was R11 describing their coding and robotics unit focusing on designing for disabled populations. They described:

> [This lesson is] using the Lego spike essential set to have a little mini figure character who's in a wheelchair, so they have to build a taxi. The story is that this character is going to visit a landmark in the fictional city that Lego invented, and he needs to get from one end of the city to another. He's in a wheelchair. So the kids have to invent a little model robotic car, and then figure out a way, after building it, to code it in a coding app, and to get it to move. They test drive their little taxi that they built. Then, they were challenged to improve either the design of the taxi or the code. I had them create obstacle courses and [they] tried to code their car to go through the obstacle course. (R11)

The respondent continued to describe how students intentionally designed the taxi to meet the needs of the disabled character. This respondent spoke about how they rooted computer science instruction in real world applications, but also pushed students to incorporate design for underrepresented populations.

Environmental Activism

One respondent explicitly spoke about incorporating environmental activism into their computer science teaching, with the goal of increasing student awareness of that social issue. R14 said:

> I wanted them… to see the video, "Meet Your Carbon Footprint". [It] shows regular life in an animated form. And it was a good one to show for that, because I wanted to tie in global warming, and I want them to be thinking about their environmental impact. And I always try to squeeze those kinds of things into our computer science stuff, things that are happening in the real world that they need to hear about, in everything that they do, you know, just to be globally aware. (R14)

This respondent articulated promoting awareness of environmental issues as a goal in their computer science instruction, and articulated encouraging students to use computer science to address environmental issues.

Student Empowerment

One respondent specifically spoke about empowering students to use computer science for social change by using technology to make the world a better place. R4 stated, "...we are showing our girls that there are a lot of women out there that are making, and… using technology for good, it wasn't like, you know, consumerism as much as trying to make the world a better place." While some see the importance of computer science instruction as purely economical, this respondent recognized the importance of providing students with the tools and skills to improve the world around them.

Increasing Academic Achievement

Only one respondent listed increasing academic achievement as a goal for their computer science instruction. R12 described how, "...we're always a little bit lower scoring in math at my school. So I try to do as much secret math as I can, in my robotics units" in order to give students more exposure and real-life applications to the math skills they are learning. They described further:

> We're trying to tie it to something that they need a little bit of work on, which is geometry because that's consistently one of our lower areas. It's geometry, and then measurement, our two lowest scoring areas every time. That was in the back of my mind whenever I designed some of these different challenges, like how can I tie that in and okay, we're talking about what makes up a square. Why is that important? What makes up a triangle? What's the measure of the angles? So doing all of that. (R12)

This respondent identified that student performance in math was low, and articulated their use of computer science to incorporate more opportunities to practice the math concepts and apply them to real life situations. One goal of their computer science instruction was to increase math achievement.

Teachers' Strategies for Equitable Practices

Respondents discussed their beliefs around equity and how they tied to their goals for computer science instruction. All respondents articulated examples of strategies they implemented in their classrooms to make their practices more equitable. This section will focus on the strategies that 50% or more of respondents articulated.

Fostering an Inclusive Classroom Environment

All respondents spoke about fostering an inclusive classroom environment in order to make their practice more equitable and inclusive. In general, respondents described a need to ensure belonging and safety for all students in their classroom. Others provided detailed examples of fostering an inclusive classroom environment through the inclusion of families and student communities, scaffolding to meet the needs of all learners, and incorporating student identity and cultural backgrounds into computer science instruction. Respondents had various ways of including these strategies. Incorporating families and communities into computer science learning was the most common strategy discussed by respondents.

19 respondents gave specific examples of how they included families and community in computer science instruction. Some were as simple as informing families of what students were learning in computer science. For example, R2 said, “I do send out information about what we do each day on our Remind app.” Other examples articulated included deeper engagement with families and communities through parental involvement and family nights. R6 stated:

> We have a family night where kids come in one night, every few months. And you know, we do bring families into our classrooms, we do kind of show them the computer science thing. I try to share everything we're doing in [our classroom] to include family, so they feel like they know what we're doing. (R6)

Both R2 and R6 mentioned the importance of informing parents and families about what their children were learning so parents and families were aware. R7 provided more expansive ideas of how they claimed to engage parents and families. They stated:

> I think families, community, and community partners are so important in education. They should be included in these activities that we're doing. And I have invited many parents in [to the classroom] who are in these fields, and bringing them in and helping out and doing things. And you know, we celebrate that here. We invite families to come in with their students during their special time here during STEM, and we invite them to come in and let their kids teach them how to code. They think it is so cool that their kids can do this. And they have no idea how to do it or that their students understand this. Parents being involved in the learning and sharing their knowledge and things like that. I think that is extremely important. (R7)

This respondent articulated leveraging families as a tool to have students share their learning and knowledge. Notions of family engagement were varied in terms of how much and what types of engagement teachers mentioned.

All respondents verbalized the importance of fostering an inclusive classroom environment and 18 respondents spoke specifically about inclusion and sense of belonging in the classroom for students. R1 stated, “I just want everyone to dive in, I want everyone to be involved. And I try to create a climate where it's safe for everyone to try, and it's safe for everyone to fail and try again.” The respondents voiced a need for students to feel safe and take risks. The respondents articulated that they were able to build this community of safety and risk taking by providing an opportunity for students to explore various strengths. Multiple respondents discussed inclusion through providing students opportunities to find strengths. R6 stated:

> I've seen a lot of students who excel in here aren't students [that are at] the top of their class. I think that's because in my classroom, kids are up and moving, kids are talking to each other, kids are hands-on, and they're getting to tinker, and be creative and build where they're not doing all of those things all the time in their classrooms. So I guess why they're doing well here is, you know, I have the alternative seating, I have options for students to get up and move around and work with different people. So I think because it's hands-on, I think because, you know, I'm not making them sit there still, I think they're

> just getting a different perspective and trying different things that they might be better at. (R6)

This respondent spoke about how their specialty computer science class allowed for students to engage in their computer science learning much differently than their traditional classroom setting. As a result, students that did not excel in traditional subject areas had the opportunity to excel with computer science. The respondent verbalized successful inclusion of students that may otherwise have been excluded because of their perceived ability. Collaboration was described by most respondents as a strategy to increase a sense of belonging and inclusion. With computer science teaching, collaboration was discussed as a strategy to promote equitable teaching practices. 18 respondents described how they incorporated collaboration in their classrooms. R3 stated, “I think coding in a group also has benefits too, because there's like teamwork and collaboration.” Respondents discussed the value of student collaboration, and gave examples of how student collaboration provided access and equity. R5 described, “We do have events like code night where we mentor the kids to be presenters. And also [have] students share what they code with their peers. And then we have a little hackathon kind of thing.” This example of student discussion, interaction and collaboration with computer science allowed students to engage more deeply with the content. R1 described the value of collaboration by stating:

> Next time I do it, [I’ll do] groups of three. Just because I think in groups of three, there's more opportunity for everyone. Sometimes in groups of four, it's easy for one to get lost. By the end, everybody was involved, but it would force a bit with a group of three. (R1)

This respondent stated that they leveraged collaboration and group work to increase engagement with computer science learning. Others discussed leveraging student interest to promote an equitable and inclusive classroom environment.

Respondents voiced the value of aligning student interests with instruction to promote student engagement and an inclusive environment. R5 stated:

> I do try to seek opportunities to address different types of learners that we have. We have some who love math and love numbers, and then we have someone [that loves] the arts. And I feel like the design should not be only for those who are good in science and math, because you do need diverse thinkers across the board to develop a good product. So focusing on it from one angle, either the mathematical part of it, or the science part of it, you tend to lose a few who might not be able to keep up or who find that's not interesting. So you find applications that they may see in say, arts or performing arts or visual arts, so finding different avenues for them to see application, and practice any skill that they've learned. (R5)

This respondent expressed how they leveraged student interest to create an environment where strengths and interests were incorporated into computer science instruction. Other respondents articulated utilizing scaffolding as a way to ensure engagement and understanding of all students in their class.

Student Identity

15 respondents described incorporating student identity as a strategy to create an inclusive classroom. R2 gave examples of providing deeper and more meaningful engagement by stating:

> I would love for students to show off more of who they are, and have more voice and share more from their background. In a previous lesson, …students were coding projects in Scratch. I was just telling them, you can change it to any language, I pointed that out. And like there's also the translate block. And so one of the kids had one kid whose family spoke Russian and one family

> who spoke French. And then we're also in a dual language school. Then, I got all these like, multilingual projects, and it was really quite awesome. It was really great. And the students are really interested to hear and then our platform allows the parents to respond. So we're getting responses from parents and other languages, other languages and you can hit the Translate [button] and see what they're saying. So the kids were like, just delighted to bring that in. (R2)

This respondent showed the link between incorporating families and student identity. They articulated leveraging both strategies to promote computer science learning and increased student interest and investment in learning. Other respondents gave more examples of how they included family, student identity, and culture in their computer science teaching practices. One example was the use of a family and cultural survey that was utilized to create instructional content. R4 stated:

> We send a family cultural survey out. And we've gathered information about what traditions [you have in] your families, things you celebrate, and things that are important. You have values that you all uphold and things you talk about. We're going to create a digital book that celebrates each person's traditions. So it could be as simple as we have pizza every Friday night and a movie night or that we celebrate Diwali. Or, something along those lines of we have an extended family in our house and we have a big family celebration every Friday with all our cousins and aunts and uncles. I mean, it could be whatever the case might be. So yeah, [I'm] taking those elements and bringing in infusing into the classroom. I think that's what we should always be doing, whether it's computer science or not. (R4)

This respondent articulated how they intentionally sought out information about students' families, culture, and traditions, and articulated the importance of including these elements in all teaching and learning. Other respondents spoke about providing examples of student identity in their computer science teaching. 15 respondents described how representation of student identities and cultures was utilized as an inclusive classroom practice. R6 stated:

> One thing I started with, I like to incorporate picture books a lot with my STEM and my computer science lessons. And when I was doing this, I realized that every holiday I was doing was an American holiday, [and] every picture book was a white person. My students [who] were Chinese made me think about a lesson celebrating the Chinese New Year. I was like, wow, I need to figure out how to include holidays that other kids are celebrating. So that's one of them I integrated in. But then I was thinking about the picture books. And it was like, every single picture is every white student. I have students of other ethnicities and races that I need to include. So where I started is I wanted to create a diverse STEM classroom library. So I incorporated a ton of books that included all different races and genders. (R6)

This respondent articulated how they prioritized representation of various cultural backgrounds and ethnicities in their classroom to foster an inclusive classroom environment. R6 continued to describe the importance by explaining:

> I think student identity is really important. I think creating those meaningful learning experiences is so important. And that's one way that you're going to really have students remember this lesson and care more about this lesson, when you are creating something meaningful for them. If you're just doing it to do it, I mean, they're not going to care, I think it has to be meaningful. When I was incorporating the backgrounds of students, their race, different things like that, their holidays, their beliefs, I feel like they felt more connected to it, because they knew I was trying to build that relationship they knew I was trying to like, pinpoint something that they were good at, or something that meant a lot to them. And I feel like when you do those things, you're just building better relationships with your students. (R6)

This respondent articulated how they leveraged student identity and culture to build connections for the students, build positive relationships, and to provide meaningful instruction. Other respondents articulated how they provided students an opportunity to explore other cultural backgrounds through computer science. Some respondents focused on cultural elements by having students explore cultures outside of their own in their computer science learning. R9 explained:

> We used computer science to do a research project on a different culture that wasn't their own. So they did. They did their research on it, they read books about it, they wrote a report about it. And then they did a Scratch project that made it look like a presentation where they had to show the flag and they had to draw it to draw out. They made a presentation with Scratch. (R9)

Respondents articulated how they explicitly incorporated cultural awareness in their computer science instruction, and some focused on cultural awareness. Others articulated how they focused on student collaboration as a way to foster an inclusive classroom environment.

Scaffolding

15 respondents spoke about scaffolding as a strategy to promote an inclusive classroom environment. Respondents spoke about providing avenues for student success in computer science as part of their strategy for making their computer science instruction equitable. Some spoke about scaffolding as a way to ensure success for students. For example, R2 stated:

> I also preview some of the new blocks like previewing vocabulary, so that when they see these things come up in the lesson, it's not like a shock to them. You've seen this, [and] this is coming up from the lesson. And then, one of the things I want to do because of this, the concept is tricky, I have an unplugged activity, so that we can take it outside the computer, so we could just focus on the concepts. (R2)

This respondent was able to give an example of their strategies to promote equity by stating that they scaffold content so all learners had access. Some respondents connected the idea of providing rigorous challenges with the strategy of incorporating student agency and voice in instruction. Respondents discussed providing rigorous challenges and some included ideas of how rigor and student choice, voice, and agency were closely tied strategies to promote equity in their classrooms. R2 stated:

> I have mild, medium and spicy challenges. …And I'm moving away from having them copy my code step by step, and telling them okay, now you need this. And now you need this, because I used to do a lot…. then, when they're doing that activity, they weren't copying the code. For me, we're just reading and talking about it. And then when they went to the challenges, they were able to modify this code. And a lot of kids in this lesson did get to the spicy challenge. Even though … they weren't able to complete it in the time we had. But they were wrestling with it. They were talking about why it wasn't working. I was surprised that they got to the spicy challenge on this one. (R2)

This respondent discussed giving students mild (easy), medium, and spicy (hard) challenges as a way to scaffold learning and allowing students to choose what challenge best fit their needs and interests. This respondent articulated how they explicitly utilized rigorous pedagogical strategies in combination with student voice and agency to make their practice more equitable. All respondents also described incorporating rigorous pedagogy as a way to increase equitable practices.

Incorporating Rigorous Pedagogy

All respondents discussed incorporating rigorous pedagogy as a strategy to promote equity in their computer science teaching. When speaking about rigor as a strategy to increase equitable practices, respondents had varying responses around the idea of rigor. These responses included thoughts about high expectations, navigating frustration tolerance and promoting perseverance, providing opportunities for authentic problem solving, and offering rigorous challenges. 14 respondents spoke to the idea of having high expectations for all students. R1 spoke about having high expectations of all students when they stated:

> I think rigor is huge for me, because I truly believe that the kids are capable of so much. If we don't, we don't ever challenge them, then they'll never, they'll never have the opportunity to meet those goals if it's not presented to them … I always tell my kids I have high expectations in here, and you're going to meet them, because I'm going to help you. (R1)

This participant tied their strategy of having high expectations of all students to meet their goal of providing access to computer science opportunities. In addition to providing high expectations, R4 discussed the relationship between high expectations and scaffolding in their classroom. They stated:

> ...We always have high expectations for our students. …I think more importantly, is that we create a low floor that all [kids can] hit and then we can have personalized expectations along the way. I'm not saying the floor should be so low that like, you know, anyone can hit it. But it should be low enough that someone's not gonna fail, [but] they're gonna reach that and feel good about that, and then might push them to understand the skills that we're trying to accomplish. … Obviously, we want to have high expectations for our students, but we also want to be successful. And how do you balance that? (R4)

Both respondents realized the power of having high expectations for students, but also understood that scaffolding and support were essential in their pedagogical practices in order to provide equitable opportunities for all students. Other respondents found that pushing students' frustration tolerance to provide opportunities for perseverance was another strategy for providing rigor in computer science instruction.

Respondents in the sample spoke about providing students the opportunity to work through frustration and giving them opportunities to persevere. Six respondents spoke specifically about this. R3 stated:

> Even though they're struggling, I feel like they have such good perseverance to try to figure it out and willing to help each other figure it out. I think it's gonna help them in the long run with endurance, trial and error. It's okay to

> fail and make mistakes. That's how you learn. And that's like, one of the big things, it's okay if you make mistakes, because that's how we learn. And I know, that's what we try to instill in them, that it's okay. (R3)

Respondents discussed how computer science instruction provided authentic opportunities to work through frustration and how they supported students in persevering through challenges. The idea of providing opportunities to show persistence closely links to the idea of having students problem solve.

13 respondents discussed providing opportunities for problem solving as a strategy to provide rigor in their computer science instruction. R14 stated:

> [Students have] that natural curiosity…towards problem solving. We talked about that [at the] beginning [of our] our coding unit today. That, you know, while looking at a sequence and trying to find the bugs and different things in their code that they have to go through, and they have to come step by step. We talked about the amount, the lines of code that are in their iPad, and [are in] the apps. …you know, one little thing goes wrong, you have to go back and you have to try to identify where that is, what can be done to solve the problem, [and] what are the possible solutions? And then try it. But, you know, also having that persistence to know, 'Nope, that didn't work.' And telling them that it's okay that it didn't work, [and] that's how we learn … we don't learn by getting it right. We learn when things go wrong, and make corrections and changes. (R14)

Respondents articulated how they saw computer science as an opportunity to provide authentic problem solving opportunities, and having students apply their critical thinking skills. Respondents saw the value in having students work through the problem solving process and thinking critically about how they could address the problems posed. Other respondents voiced the value of incorporating student agency and voice into their teaching practices.

Incorporating Student Agency and Voice

All respondents described incorporating student agency and voice into their computer science teaching practices. In this section, I will describe the variations of how respondents leveraged student choice, voice, interest, and ownership of learning.

Student Choice

18 respondents spoke about incorporating student voice. Respondents gave many examples of how they provided student choice in their computer science instruction. One example was when R11 discussed allowing students to choose their partners for a coding project. They stated, " [When] we started Legos, I told them to start talking to each other and figuring out who they want to work with, so that they came in on Monday already knowing who their partner was." Others incorporated student choice into the structure of their computer science activities. R1 described allowing students to choose how they wanted to show their learning through various options. They stated:

> They connected with it so much that whenever we did social studies projects and science projects [I] would always leave that option. [I] would give them a menu of options. And then there would always be that one extra option that was to create your own project. Submit your idea. I had at least three of my 22 [students] that said, "Hey, can I create a scratch assignment for this? Can I do this project on Scratch?" And so for them, that was their voice, that was how they wanted to express themselves. It was amazing. They did it on their own. (R1)

This respondent described how they allowed choice to be built into their teaching practices, and how students opted-in to incorporate their computer science knowledge into other content areas. Other respondents articulated how they were more explicit in providing choice for students within the context of their computer science lessons. R8 described how they allowed students to customize and personalize their computer science projects. They stated:

> [Students would ask], "Could I change this?" Then as we're doing [the project], [students would ask], "What kind of backgrounds? How many points should it take away? How much time should we have for all those things?" [I'm] including them in the process of building it. So we're collectively [building] but it's not me telling you how it's done. (R8)

This respondent allowed student choice in their teaching in order for students to personalize their learning.

Student Voice

18 respondents described seeking out opportunities to promote student voice in order to facilitate learning. R16 stated, "Well, student ideas and input, that's always important. They're always encouraged to share what they have or what they've learned with each other and with me, and I like to share their creations with the class." This respondent articulated how they leveraged student voice to have their students co-construct knowledge. Other respondents discussed how student voice enhanced learning in the classroom. R16 stated:

> I had students with different projects come up with different ideas that maybe I haven't thought of. They explored on their own and figured out how to do certain things. So when they do something that I didn't know about, or that they found was really cool, and where if I feel like it'll be beneficial to all, whether or not is beneficial to all, I think it's important to give students the opportunity to share what they learn, … and they're able to share it with the class. (R16)

This respondent articulated how they focused on student collaboration in combination with student voice to enhance their computer science instruction. R13 corroborated this position by describing how they utilized student voice to enhance instruction. R13 stated:

> Sometimes along the way kids have really great ideas. They're like, “What if we do this?” I'm very much about, “Yeah, let's think about that. Maybe we need to tweak this lesson a little bit. Because what you're saying is, “Yes, I absolutely want you to show me that in the lesson.” I'm not set in stone when I have a lesson, I have a roadmap of where I want to go. I'm absolutely open. Because I think if kids, students start to suggest things, that means that they're more engaged in their learning, and they will probably keep that information a lot longer than if I just tell them how we're going to do it. (R13)

This respondent connected the idea of allowing opportunities for student choice to student investment.

Student Interest

12 respondents spoke about promoting student interests in their computer science instruction. One example of inclusion of student interests and voice in computer science education was when R10 described teaching the academic concept of arrays, but including student interest in the content. R10 stated:

> You can store anything in an array. And so one time, we made an array called BlackPink, and we stored all the members of the music group in the array, so you could use it later on [even though] you don't really know it's gonna tell you… but keeping those moments to make it more engaging and related to what the kids actually care about, is really important. So that's what has happened in the array lesson. (R10)

This respondent acknowledged that the Korean pop group, BlackPink, had little relevance to computer science concepts being taught, yet still included BlackPink in instruction to promote student interest and opened opportunities for students to incorporate their voice.

Overall, respondents gave examples of how student voice and agency were utilized in their classrooms to increase equity. These responses covered ideas ranging from giving students choice in partners, to the design of their learning artifacts, to incorporating student interests that were not relevant to the computer science content.

Student Ownership

8 respondents discussed student ownership as a strategy. One respondent described an app challenge unit that they taught as part of their Computer Science Instruction. They described how student choice and voice allowed for deeper ownership in student learning and student learning artifacts. R20 stated:

> I have a group this year, that with all of our social emotional learning lessons, and our talk about being stressed and how it's difficult coming back, and I'd say there's like they're doing meditation, like an app for meditation and exercise, and helping helping students come be able to calm themselves down by themselves, not having someone else do it for them, being able to meditate, helping them do the breathing exercises. That group of students not only thought of the idea themselves, but created music to meditate with. That's how, you know, how determined they were. They were creating music, they were creating sound bites of how you if you've ever listened to the Calm app, and then that voice and the melodic voice, that kids were creating sound bites for the different exercises and stretches and the poses that you are going to have. I mean, come on, you cannot ask for more buy in when it's something that comes from them. You know, I didn't tell them what kind of app to create. They came up with it on their own. Same with their games that they came up with. I didn't tell them what game [to make]. They're thinking, "What does my buddy need? My buddy is having a hard time with their sight words, [so] I need to create a game that's going to want them to play and want to practice and want them to learn so they can get better at it." That elevates the determination to be successful. It also makes them think about what is going to help their audience buy into it, especially with the little bits one. It's not just what's going to make them entertained by it or them wanting to do it, but what can I do to make it so my buddy is more engaged in the learning process as well. I can't even begin to tell you the elevation of agency with that. (R20)

In their description of the app projects, they describe how allowing students to choose what type of app to make and the audience that it serves increased student ownership and agency with computer science.

Creating Cross-Curricular Connections

Respondents gave many examples of how they tied computer science instruction to other subject areas and created cross-curricular connections. 17 respondents gave specific examples and discussed how cross-curricular connections provided concrete examples for learning. These respondents voiced how cross-curricular connections with computer science allowed students to reinforce learning from other subject areas.

Some respondents gave examples of how they provided opportunities to create cross-curricular connections with computer science. R16 gave an example of a cross-curricular computer science project their students worked on. They stated, “I wanted to get into Scratch Jr. to teach about seasons so that the kids can show me the different changes in seasons through their work.” They saw value in having students show their content knowledge and incorporate their computer science learning. R18 expanded on this idea and stated:

> One of the things that I'm trying to do is to bridge the gap between my classroom and their regular classroom. And to say, “How can we apply this concept when it comes to learning math?” Or, when it comes to writing, when it comes to something in science, that they have opportunities to use, you know, use their learning in their own way to tell a story or make a game. (R18)

Others made a connection between computer science and practical applications in language arts. R20 described, “There's so much overlap with language arts and cause and effect. [With] social science and science, with the whole cause and effect, that is what coding is. Cause and effect, with our particular project, [is what] we were working on.” Others verbalized the importance of providing students with concrete computer science experiences to construct knowledge in other subject areas.

One respondent described students not understanding basic measurement skills, and connecting computer science to reinforce the concept of measurement. R12 stated:

> [I try to] find ways to tie in math vocabulary and application, because I find so often, by third grade, you think that they're like, completely capable of measuring things until you hand them a yardstick. And they're like, “Wait, like, what do I do with this?” They can do it on a worksheet fine. They can use a ruler on a worksheet, but you hand them a yardstick and tell them to measure a distance of 24 inches and like, like their heads explode. So I think that's a really big part of this, too, is just finding ways to tie in all of that curricular content, but also, the application like working through the problem solving and like, oh, yeah, you have to measure like, you have to know how to use use these tools that you've learned about in a more abstract way. Like, here it is concrete[ly]. (R12)

Providing students with opportunities to practice cross-curricular skills in combination with computer science skills proved to be beneficial for this respondent. Other respondents voiced similar thoughts and gave examples of how they used computer science to reinforce student learning in other subjects.

Multiple respondents described how combining computer science with other curricular areas allowed opportunities to reinforce skills. R19 connected computer science to reading instruction. They stated:

> So coding with kids before they can read will help them become better readers. I come at this whole thing from [a] literacy [standpoint]. From that standpoint, the more I learned about coding the more I loved it, the more I'm like this, this is an activity that we should be doing with kids because it builds the brains and the directions. (R19)

They stated how the computational thinking component of computer science helped reinforce literacy skills. R6 described how computer science instruction reinforced student learning in science. R6 said:

> My students in their classroom, I knew they were learning about the solar system, I knew they were ordering the planets. So for me, I had them do the retelling, [and] they had this show how the solar system, how the planets were lined up, how they rotated around the sun. I had my students add in robotics. So now they're having to add the computer science aspect, they have to figure out, "How do I write the code to make it travel in the circle? How do I make it go to the next space and travel that same path." I was doing that science aspect, but then I'm making them do the coding to where they're adding the blockly codes, [and] they have to figure out the math how to make it work. They're doing trial and error. So if they would have just been in their classroom, [they would be] drawing a thing of the planet. Now they're adding all these other pieces and adding in computer science and math, coding, and, you know, a lot more problem solving and trial and error than they would have if they would have just drawn the solar system. (R6)

The respondent described how they created the cross-curricular lesson, and then explained how the cross-curricular connection provided opportunities for reinforcement of learning. They stated:

> So I'm thinking of the planet activity, the meaningful learning experience, I tie that activity in because I knew it was one that students were learning in their classroom. So for me, I was doing that because it was a review. It was something that they had already known. And it was something that they could kind of show that I knew this and if students didn't know at all, I also played a brainpop video to make sure they could check their answers before they moved on. (R6)

The respondent articulated how they utilized cross-curricular implementation of computer science and how it allowed for students to more deeply explore what they learned in other content areas, and reinforced student learning and knowledge.

Addressing Racism in Computer Science

Fourteen respondents spoke about how addressing racism in CS was a strategy they used to promote equity in their computer science teaching. R6 stated, "I try to focus on the anti racist practices.” R4 expanded on this idea and incorporated this pedagogical approach to their computer science teaching specifically by stating:

> Most people in technology are old white dudes, like, you know, like me. So like, how do we change that? How do we bring other voices, but it's not…just hint[ing] in my class. They're hearing this in other parts of the class, and I am thankful that we're in school that it talks about that kind of stuff, and we don't, you know, we're not hiding from [the topic]. You know, I think we have to think about developmentally appropriate ways of teaching that I think that's some areas of growth. (R6)

This educator articulated how they prioritized student awareness of racism in computer science, and focused on ways to highlight the lack of diversity in the field. Their school environment allowed for discussions around equity and race, and this teacher included this pedagogy in their computer science teaching practices.

R18 recognized the importance of addressing racism in their computer science teaching. They stated:

> So, yes, it definitely is something that I think about and [I'm] trying to figure out how to present it in a way that's going to be relatable to really young kids. They're all white. How do I broach that? How do I? How do I show where bias is happening, where computer science is causing harm, right, that's another thing that I'm trying to do is, bringing examples of where things are not great all the time. There's harm that is being caused. I've definitely tried to do a lot of that work with teachers as well. [I'm] getting them to sort of understand around where this bias is happening, where we really need to be critical about how we're applying these different applications and these algorithms and those kinds of things. I'm trying to find my place in how I can lead. (R18)

These respondents articulated how they recognized their role in systemic racism, and understood that racism plays a major role in the professional field.

Highlighting Professional Relevance

Highlighting professional relevance was discussed by 14 respondents. These respondents spoke about highlighting jobs and careers that computer science could potentially lead into, and others discussed opportunities to bring professionals from the field to their students. R1 describes how exposing students to computer science careers was a focus of their computer science instruction. They stated:

> We talked about how there are so many jobs out there. CS related jobs. That's something that we mentioned all the time that this isn't just something we do now. [We highlight] that it's actually a skill. We're helping build in these [skills] for their futures, and [highlight] that there's all these opportunities out there. As a whole school, we talk about it a lot. We talked about [how] there's lots of jobs, and this is a skill that you could use to do so many things in the future. So there is that kind of conversation. In terms of exposure to those professionals, I think that's still pretty limited. (R1)

This respondent articulated how they recognized the power of exposing students to various career options, but noted the limitation of their strategy. While discussion of careers was a priority in their classroom, exposing students to career professionals was limited. Other respondents provided examples of how they think about incorporating professionals in the computer science field into their classroom teaching. R20 described:

> I use the Stanford thought process for their design thinking school. [I'm] wondering if I can connect with Stanford itself, and see if they've got people that are in the program that be willing to Skype or zoom with my class to give them some ideas on design thinking and challenges or anything like that. Especially if we've already started and the kids have issues or problems, it'd be great to bring them in to help with the process and to interview and ask, "What was your purpose for going into this school and doing this? What are you hoping to do? Job wise?" (R20)

Others gave specific examples of how they incorporated professionals in their classroom teaching. R7 described their process for exposing their students to professionals in the field. They stated:

> I'm always trying to network and get people in here from all types of areas and all jobs, so that my students can see men or women, there's not a barrier between race or education levels. And I asked them to share those things with me, you know, their hurdles are with the students,their setbacks, and how they got to where they are. I think that's important for those students to see that and for their goals, in the end of what they want to do, and to see a diverse job or careers out there for computer science. (R7)

70% of the sample discussed how highlighting professional relevance was a strategy for equity in their computer science classrooms.

Modeling Professional Inclusiveness

13 out of 20 respondents discussed how they modeled professional inclusiveness in their computer science classrooms. R10 gave multiple examples of how modeling professional inclusiveness can be incorporated into computer science teaching. They stated:

> November is a Native American heritage month. This was trickier to find, as role models from that have Native American identity as part of their experience, but also working in computer science. There's one article that I found online that has 18 rock stars in the Native American community working in computer science, but it was clearly written for an adult. So I took that article, took those pictures and basically made these [into] little, bite sized, child friendly reading level [articles]. I called [them] biography paragraphs. Those will be hung up outside my room. [I wanted] kids to see that there are people who share their identity, that are doing really cool things in computer science. I do make sure to balance. For example, on my wall, I have Kimberly from Black Girls Code. I want to make sure that kids know that their identity can be a part of the work that they do in computer science.…Those are the opportunities, I use the videos and the visuals that I provide. We have read alouds too [focused on topics] like Katherine Johnson, and another book called Social Code. This is a little Black girl that I'm reading [about]. Having those visuals around, [and] just letting [students] know that people that look like them go into this career. It's possible. I [also] identify, especially in the older grades, that it might be tricky there, there will be obstacles, and it will be tough, but just like other things have been tough in their life, they're more than capable of handling it and we'll be working to dismantle that. (R10)

This respondent articulated how they saw the value in modeling professional inclusiveness, tied it to specific ways of promoting professional inclusiveness in the classroom, and also acknowledged that highlighting the challenges of inclusion was a priority in their teaching.

Incorporating Relevant Pedagogy

12 of the 20 participants spoke about incorporating relevant pedagogy into their computer science teaching. These educators identified the power of incorporating student interests outside of school and incorporating them into computer science teaching. R2 described how they incorporated authentic learning into their teaching practices overall by stating:

> There's always a hook question where I've tried to relate things to their lives that are not necessarily from school. So just talking about lives at home, or things they've seen on TV, in the questions to kind of draw them into what we're what we're talking about, and see how it relates to their everyday life. (R2)

R20 expanded on this idea by describing the App Challenge that they have their students participate in. They described how they incorporated relevant pedagogy in the context of this challenge:

> I had a group of four girls that were working together. I said go home, and talk to your parents about what apps you use. Ask, "What's important to you? What do you need? Look around, talk to neighbors, talk to friends. One gal had gone home and talked to her mom specifically about this. She had a 16 month old younger brother. Mom had mentioned that the baby monitors…she wasn't happy with what was on the market. The girls started to go, " Baby monitors, a baby app, what can we do?" I don't think the other three had younger siblings. They said, "Have a picture of your little brother, and show them you have a purpose here." It's incredible. Then they go back to ask more questions. [They asked Mom,] "What else would you want with this app?"

> Their app grew based on the fact that it was something that was important to them. (R20)

This educator articulated how they empowered students to create and utilize their computer science learning in authentic and meaningful ways for students. Relevant pedagogy was a theme mentioned by over half of the participants and many gave examples of how they incorporated authentic learning into their computer science teaching practices.

Overall, this study examined teachers' views on equity in relation to their goals for computer science instruction. Teachers articulated various strategies to make their computer science teaching practices more equitable. Respondents interpreted equity as opportunity and differentiation. Their goals included providing access to opportunity, promoting positive identification with computer science, empowering students to use computer science for social change, and increasing academic achievement. They described using strategies like fostering an inclusive classroom environment, incorporating rigorous pedagogy, incorporating student agency and voice, creating cross-curricular connections, addressing racism in computer science, highlighting professional relevance, modeling professional inclusiveness, and incorporating relevant pedagogy to increase equity in their computer science classrooms.

CHAPTER 5: DISCUSSION

The purpose of this study was to explore how elementary school teachers think about equity in relation to their goals in their computer science instruction, and to explore the strategies they articulated to make their practice more equitable. Answering these research questions could add to research that may eventually lead to recommendations of computer science instructional best practices and provide teachers with effective strategies. States are mandating computer science standards and instruction and requiring schools to teach the content (2020 State of CS). Parents want students to be taught the content, and students in affluent and predominantly white communities have greater access to computer science opportunities than their low-SES and minority counterparts (Margolis, 2020). In addition, teachers are not formally trained or prepared to teach computer science (2020 State of CS). This study adds to the limited body of research that focuses on computer science instruction at the elementary school level.

This last chapter will focus on discussing key findings through the NASEM report (NASEM, 2021b) and the Kapor Center's Framework for Culturally Relevant-Sustaining Computer Science Education (Appendix C) in addition to the extant literature that informs computer science education at the elementary school level. Finally, the chapter will discuss implications for practice, present limitations of the study, and propose recommendations for future research.

Discussion of the Findings

Interpretations of Equity

All teachers in the study described equity as access to opportunity to computer science instruction and frame goals around that idea. According to the findings, all teachers defined equity as opportunity or differentiation in order to provide equitable access to computer science instruction. In addition, teachers discussed how positive identification with computer science was a main goal of instruction, but none of the participants listed this as a form of equity. According to NASEM (2021b), there are four categories of equity within computer science education:

5. Equity as access and opportunity,
6. Equity as achievement and/or positive identification with computer science as a discipline,
7. Equity as expanding what constitutes computer science in the form of integrating alternative cultural perspectives, and
8. Equity as including computer science as part of justice movements (NASEM, 2021b)

The findings of this research support some elements of this framework, but notably, teachers did not mention equity as “expanding what constitutes computer science in the form of integrating alternative cultural perspectives” or “ including computer science as part of justice movements” explicitly. Teachers in the study were committed to making their pedagogy equitable, but did not express their views on equity in these two ways. However, their goals for computer science instruction directly tied to their idea that equity in computer science education meant allowing all students access and opportunities to participate.

Past research has examined the relationship between teacher's beliefs around equity and its connection to computer science education. Other studies have looked to "...make visible teachers' beliefs about educational equity, and the role of CS within these beliefs about equity" (Gomez, Lee, & Berkhoudt Woodman, 2022). This study further extends on that research, and supports findings in the field around teachers' perceptions of equity in the context of computer science education. By and large, respondents viewed equity as an opportunity to engage with computer science content. This confirms existing literature. Teachers expressed that not all students have the opportunity to engage with computer science. These teachers recognized students in their classroom and in the country, had varied access to computer science education, and acknowledged that providing access was an equity issue. These views supported and expanded field discussions that "...gaps in access to rigorous CS education between students from minoritized backgrounds and their more privileged peers" (Gomez, Lee, & Berkhoudt Woodman, 2022) exist. Teachers widely accepted that opportunities were limited for some students, and this would directly impact student experiences and outcomes. In addition, teachers were able to articulate the urgency and importance of computer science education. Teachers were aware of the need for students to be prepared with skills for their future, and how computer science skills were becoming increasingly relevant in their students' lives. These views further support research and further confirm that, "teachers sought to pursue varied learning outcomes for students they believed CS lessons were well-positioned to facilitate. These beliefs and goals were informed by their understanding of the role of CS in broader social contexts, (e.g., the labor market), the affordances of CS education, and how these intersect with the needs of their students"

(Gomez, Lee, & Berkhoudt Woodman, 2022). In these teachers' views, providing opportunities for students to engage with computer science allowed students to better prepare for their futures. Yet, teachers did not recognize that, "access to more rigorous CS education alone is unlikely to reverse these inequities" (Gomez, Lee, & Berkhoudt Woodman, 2022). Teachers prioritized providing opportunities, but did not deeply explore that opportunity alone was unlikely to change outcomes of students. Yet, teachers believed in the power of access, and expressed their intent to ensure all students had the opportunity to engage in computer science learning.

Teachers expressed a common view about equity as providing opportunities for students and they framed instructional goals around that view. They described instructional strategies in terms of accomplishing those goals. This study extends on prior research by Gomez, Lee, & Berkhoudt Woodman's (2022) that focused on teachers' experiences co-designing computer science education. They stated, "...we know little of what, and how teachers learn about CS education when teachers work collaboratively in professional communities to codesign the core fabric (goals, pedagogy, and content) of CS lessons and pedagogy." This study extends this research by explicitly examining the goals that teachers express for their computer science teaching. Teachers' personal interest, perceived value of the topic, and development of competence and skills impact motivation to teach the content (Bathgate and Schunn, 2017), and their goals directly align to their perceived value of computer science education.

The following section will explore how the strategies articulated by teachers in the study support the Kapor Center Framework's Core Components: Acknowledge Racism in CS and Enact Anti-Racist Practices, Create Inclusive and Equitable Classroom Cultures, Pedagogy and Curriculum are Rigorous, Relevant, and Encourage Sociopolitical Critiques, Student Voice, Agency, and Self-Determination are Prioritized in CS Classrooms, Family and Community Assets are Incorporated into CS Classrooms, and Diverse Professionals and Role Models Provide Exposure to a Range of CS/Tech Careers.

Acknowledge Racism in CS and Enact Anti-Racist Practices

While teachers did not explicitly speak to the impacts of racism in the computer science field, they did speak about modeling professional inclusiveness. While teachers did not explicitly explore the ideas of white culture dominating technology and innovation and implicitly designing and creating for their own needs, teachers did see value in highlighting professionals from underrepresented backgrounds. While not explicit in addressing the impacts of racism in the technology world, teachers recognized that the field is not diverse, and included examples of diversity in their teaching. This serves as a stepping stone for teachers to address the deeper concerns of lack of diversity, which I will discuss in my recommendations section.

Create Inclusive and Equitable Classroom Cultures

Providing Access to Opportunity

All respondents spoke about how their goals for computer science instruction included providing access to opportunities. As we saw in the last chapter, respondents expressed that they viewed equity as access to opportunity and as the execution of differentiated learning to provide access to all learners. Respondents also described specific strategies that aligned to this instructional goal. Teachers interviewed in the study recognized that their classrooms consisted of different types of learners with different needs and discussed that equitable teaching included providing ways for all students to access the content through differentiation of instruction. Their ideas around differentiation and equity were aligned, with respondents commenting on how they support all the students in their classroom based on ability, interests, strengths, and weaknesses. These views support the CRC framework that recommends, "... teaching strategies can include peer-teaching, participatory tasks, clearly stated outcomes, and discourse involving problem-based activities and include four design principles: prior knowledge, cultural ways of knowing, engagement and motivation, and civic and social empowerment" (Scott, Sheridan, and Clark, 2015).

Teachers discussed how they used different ways to include all learners, and provided examples of how they utilized some of these strategies listed in the Kapor Center framework. These views on equity informed teachers' goals for their computer science instruction. Teachers described that they utilized strategies of: fostering an inclusive classroom environment, incorporating rigorous pedagogy, incorporating student agency and voice, and creating cross-curricular connections in order to increase student access and opportunity to computer science instruction. These strategies that respondents spoke about aligned to best practices that teachers are well versed in, and utilize across other subject areas. Teachers seemed to apply what they know works best for their students in other content areas and applied them to their computer science instruction in order to make their teaching more equitable.

Teachers recognized that computer science education is not ubiquitous and verbalized a need to provide computer science instruction to all of their students. Previous research has shown that computer science instruction can perpetuate inequities, especially if only certain student populations are given opportunities to engage. For example, at the elementary school level, teachers unknowingly perpetuate these inequities by providing computer science instruction to high achievers or early finishers (Coenraad et al., 2020). Respondents were intentional about their goals to bring computer science opportunities to all of their students, with many recognizing inequitable structures within their own schools and organizations when it came to computer science education. Teachers discussed their investment in teaching the subject and the importance of providing opportunities for all students. They expressed deep investment in the content, and supported past findings that teachers of computer science want "...to have students 'transfer skills and mindsets, such as problem- solving strategies and collaboration skills, to other domains' and to have students 'understand that the challenge that they will complete is similar to careers that involve coding"(Gomez, Lee, & Berkhoudt Woodman, 2022). Exposure was an essential goal, and teachers in the study "...articulated a desire to build exposure, knowledge, and skills for students to participate in an emerging labor market" (Gomez, Lee, & Berkhoudt Woodman, 2022). Providing students with the opportunity and understanding of the significance of computer science in their lives was directly linked to teachers' computer science teaching goals.

These views on equity directly aligned to their goals for providing opportunity, and reinforced the idea from the CRC framework that all students are capable of digital innovation. Teachers articulated that they viewed equity as providing students opportunities to engage with computer science and that intentional differentiation based on students' backgrounds, knowledge, and achievement ensured all students' access to digital innovation.

Promoting Positive ID with CS

All teachers in the study discussed promoting positive identification with computer science as a goal for their instruction. Teachers articulated that they utilized multiple strategies to achieve this goal: fostering an inclusive classroom environment, incorporating student agency and voice, highlighting professional relevance, modeling professional inclusiveness, and incorporating relevant pedagogy. Teachers in the study were aware of the disparities within the tech population, and acknowledged that females and underrepresented minorities were not proportionally represented in the field. They articulated strategies to actively combat the disparities and to increase participation in computer science within their classrooms.

Teachers discussed their explicit goals of having students enjoy learning computer science, empowering students with skills, incorporating student interest, increasing engagement and confidence, and increasing representation. Teachers understood the power of including students' interest, identities and backgrounds to provide students with the opportunities to see themselves in computing roles (NASEM, 2021) and leveraged these strategies to make their practice more equitable. Teachers understood that having strong STEM identities was important and internalized that "...women with a strong STEM identity are more likely to persist in the field" (Jones, Ruff, and Paretti, 2013). In addition, teachers were very aware of the gender gap in the computer science field. Studies have shown that primary school students have developed positive attitudes toward computing and girls have been shown to be as successful as their male counterparts (Kalelioğlu, 2015) and teachers in the study were intentional about developing positive attitudes towards computing.

Pedagogy and Curriculum are Rigorous, Relevant, and Encourage Sociopolitical Critiques

Relevant Pedagogy

Some respondents spoke about various ways they incorporated relevant pedagogy in their computer science instruction. By allowing students to incorporate their personal interests, backgrounds, and identities in computer science, the teachers provided a way for students to apply their identities to their computer science learning. These educators sought to, "... provide learning experiences that are culturally relevant and incorporate student interests, identities, and backgrounds [which] opens access to personally authentic learning opportunities" (NASEM 2021a). Student interests were highlighted in their instruction and they collaborated in the construction of student learning and allowed for exploration and understanding of their identities. Their experiences incorporating relevant pedagogy supported the claim that personally relevant experiences have the potential to engage students in STEM and computing (Lim and Calabrese Barton, 2006; Migus, 2014). These teachers verified that authentic learning was tied to student engagement in computer science.

Increasing Academic Achievement

One teacher in the study explicitly named increasing academic achievement as a goal for their computer science instruction. While only one teacher named this as a goal, all teachers discussed incorporating rigorous pedagogy and seventeen teachers discussed creating cross-curricular connections as strategies for their computer science instruction. Providing rigor and connecting computer science learning to other curricular topics were strategies listed by almost all participants, but only one connected this to explicitly teaching and integrating skills in their computer science instruction to reinforce math skills that students consistently scored low on in standardized tests.

Teachers in the study showed how they were able to include the core CRC tenet of "motivate and improve science, technology, engineering, and math learning experiences." Ensuring that students positively identified with computer science allowed for natural motivation and interest in the subject for students. By creating cross-curricular connections, teachers improved STEM learning experiences by building on students' knowledge.

Teachers in the study were able to explicitly connect computer science concepts and skills to other subject areas. Past studies have explored how primary school teachers connected computational thinking to other content topics (Duncan et al., 2017) and how teachers' understanding of computational thinking developed, grew, and became more nuanced over the course of a year of professional development in integrating computational thinking into science inquiry (Yadav et al., 2018). Teachers saw the value of incorporating other content areas into computer science teaching, and developed opportunities for their students to incorporate computer science into other subject areas. Their views supported the notion that teachers who integrated computer science with core content areas showed continued interest in integration (Waterman et al., 2020). Teachers gave examples of how computer science integrated with other subject areas that were state-mandated subject areas.

A small subset of teachers in the study discussed increasing academic achievement as a goal for their computer science instruction. Specifically, teachers gave examples of how they integrated math concepts into their teaching. They specifically spoke about focusing on math concepts and skills that their students consistently score lower in. Past studies have shown that, " …when teachers integrated computational thinking with mathematics lessons, concepts increased in complexity across grades, mathematics instruction was the focus, while the computational thinking aspect was secondary, and three types of lessons were taught: no integration, partial integration, and full integration. Math content remained the priority for instruction, while computational thinking was a secondary priority" (Yadav et al., 2018). These teachers' experiences contradict these findings, as these teachers focused on robotics and coding as their primary aspect of instruction, but included mathematical concepts as a secondary priority.

In the study, all respondents included examples of how they provided rigorous computer science learning opportunities. Their responses supported the claim that, "Understandings of what constitutes a rigorous CS education have also been in flux" (Gomez, Lee, & Berkhoudt Woodman, 2022) as teachers described rigor in a variety of ways. Teachers discussed having high expectations of their students, helping students navigate frustration tolerance to promote resilience, providing opportunities for authentic problem solving, and offering challenges as "rigor". Deeper discussions in literature have discussed how, "offer[ing] rigorous, CT-focused, CS instruction to minoritized students is a critical part of addressing the broader equity challenges within CS education (Gomez, Lee, & Berkhoudt Woodman, 2022). Respondents in this study supported Gomez, Lee, & Berkhoudt Woodman's (2022) claims and discussed how providing rigorous opportunities for computer science learning tied directly to their belief around increasing equity

Empowering Students to Use CS for Social Change

Less than half of the teachers discussed empowering students to use CS for social change as a goal for their computer science instruction. Almost two thirds of respondents did speak about addressing racism in computer science as a way to have students understand the context of the computing field and the world around them. Many teachers agreed and identified that there were structures of racism in computer science, and reflected on their role within the context of racism in computer science. Teachers in the study understood that barriers in the computer science field were directly tied to issues of race. Teachers understood that, "the technology field has been historically dominated by white men, and the field has been reflective of their culture" (NASEM, 2021) and expressed interest in combating the historical marginalization of underrepresented populations. While teachers understood and acknowledged that racism manifested in computer science, they did not provide concrete strategies to explicitly address and teach about this with their students.

One of the main ideas of CRC is that learning contexts support transformational use of technology. Teachers articulated strategies that supported the CRC tenet of learning contexts support transformational use of technology. Particularly, they spoke about strategies that empowered students to use computer science for social change. Some teachers were explicitly applying their instruction to empower students to better their world and communities. Other teachers focused on highlighting how computer science education provides innovative career opportunities.

A small subset of teacher respondents verbalized a goal of empowering students to use computer science for social change. Their thoughts supported the CRC framework of using computing to empower for social critique and diminish the separation between the worlds of culture and STEM (Scott, Sheridan, and Clark, 2015). These educators saw the practical application of computer science instruction and how students could be empowered to better their communities and world.

Student Voice, Agency, and Self-Determination are Prioritized in CS Classrooms

Student identities were leveraged and brought into computer science learning, and strengthened learning outcomes in computer science. Teachers in the study recognized the importance of incorporating student voice and agency into their teaching practices. They supported the notion that leveraging elements of CRC would make their practice more equitable. Teachers gave examples how they, "...provide a deeper understanding of heritage and vernacular culture, empowerment for social critique, and appreciation for cultural diversity; and Bring 1 and 2 together: to diminish the separation between the worlds of culture and STEM;" (Scott, Sheridan, and Clark, 2015). The teachers interviewed expertly intertwined student identity, culture, and agency to promote investment and engagement in computer science learning. These elementary school teachers were able to merge student culture and computer science in meaningful and relevant ways for their students.

Family and Community Assets are Incorporated into CS Classrooms

All teachers in the study highlighted ways that they fostered an inclusive classroom environment, and specifically highlighted incorporating family and community assets. By incorporating this into their computer science teaching, teachers were able to allow students the opportunity to understand heritage and culture, and diminished the separation between student culture and STEM. All teachers in the study described strategies to make their classroom inclusive. Their examples provided supported past research that highlight the importance of inclusive practices in computer science education. Teachers expressed how they used pedagogical strategies to build on students' interest, identities and backgrounds to allow students to see themselves in computing (NASEM, 2021a) and leveraged this to create inclusive classroom environments. They discussed the three components of identity: sense of belonging, achievement, and behaviors (Carlone and Johnson, 2007; Cheryan, Master, and Meltzoff, 2015; Erikson, 1968; Lave and Wenger, 2002) and provided opportunities for students to explore their identities as computer scientists. Teachers recognized the gender gap in computing, and worked to actively combat the inequities caused by the gender gap. Teachers understood that, "... girls participate in computing courses at significantly lower rates at the middle school, high school, and university levels" (Margolis & Fisher, 2002) and actively combatted gender stereotypes around computing that past studies have found to begin to form in first grade (Master, Cheryan, Moscatelli & Meltzoff, 2017). In addition, teachers also sought to promote inclusion in the classroom by providing opportunities to include diversity, community, culture, and identity. Past studies have found that these inclusive elements were missing from over 50 computing programs (Scott, Sheridan, and

Clark, 2015). Teachers were intentional about providing an inclusive environment so that all their students could succeed.

Diverse Professionals and Role Models Provide Exposure to a Range of CS/Tech Careers

Teachers in the study spoke about the relevance of the computer science content they were teaching, and spoke about potential careers that students could pursue with computer science knowledge. Their responses supported the idea that "Professional and personal authenticity can be intertwined to create experiences that students find motivating. Providing students with professionally and personally authentic experiences may be one mechanism in addressing barriers to access in the computing world" (NASEM, 2021). Teacher respondents gave examples about how they highlighted jobs in the computer science field, sought out professionals to come into their classrooms, and connected these to students' lives and world.

Limitations

This study had limitations including using Zoom as an interview platform, having a limited size and self-selected snowball sample, potential bias around responses, and the limitations of conducting research during the COVID-19 pandemic. First of all, I used Zoom to conduct interviews and collect data. This method of data collection proved to be a limitation as a handful of interviews experienced poor internet connection on the interview and interviewee's end. As a result, some transcript responses could not be fully transcribed as segments of the interviews were not clearly captured and recorded due to poor internet quality. This occurred during certain responses, and did not affect the overall data collection. Second, the sample size consisted of 20 self-selected teachers that opted-in the study through snowball sampling. This sample is reflective of teachers that were interested in sharing their views and perspectives of computer science instruction, and may not be representative of the general population. In addition, this study focused on how teachers articulated implementing equitable strategies, but there was no accountability mechanism to ensure that teachers truly used these practices. To strengthen the findings and results of the strategies that teachers use in the classroom, observations could have been utilized. Due to the COVID-19 pandemic and participants located throughout the United States, observations were not possible.

Recommendations for Practice

Drawing on conclusions from past research in combination with the findings of this study, I provide three recommendations. First, teachers highlighted professional inclusiveness, but did not incorporate discussions around who holds the power in technology and the impact that has on underrepresented and marginalized populations. While this strategy partially addressed one of the core principles of CRC, there is opportunity for teachers to include this knowledge into their teaching practice. Teachers would benefit from training around the racial and social justice issues within computer science, and they would benefit from training around best practices for teaching this content to their students.

Second, all teachers expressed their ideas around equity as providing students opportunities to engage with computer science experiences. NASEM outlines four ways in which teachers can implement equity within computer science instruction (NASEM, 2021b), but none of the teachers discussed two main elements: “expanding what constitutes computer science in the form of integrating alternative cultural perspectives” and “ including computer science as part of justice movements”. I recommend that teachers be provided training around what these two elements entail, and how they can incorporate these elements into their teaching practices.

Third, teachers would benefit from understanding how computer science instruction can support and increase students’ academic achievement. While teachers spoke about providing rigorous pedagogy and making cross-curricular connections in their computer science teaching, only one teacher connected this strategy back to increasing students’ academic achievement. In the current educational climate, high-stakes testing majorly influences teachers’ practice. The one teacher in the study discussed how she knew the trends in the shortcomings of student math scores, and would intentionally incorporate those concepts within her computer science lessons. Research has shown that computer science instruction is becoming mandated in states, yet teachers do not have more time in their day to teach more content. Teachers would benefit from training around how to leverage computer science instruction to support student academic achievement.

Recommendations for Future Research

Combining the findings of this study with prior literature provides the opportunity to recommend future research. This study was limited in that the research was conducted during a global pandemic. To extend on this research, I recommend studies focused on classroom observations of computer science teaching, which would provide deeper insight into how teachers' goals align with equitable practices. This would allow researchers to understand if the strategies and goals that teachers articulately truly align with their implementation and practice. In addition, studying students and their families' views on equitable computer science instruction would provide exploratory examination as to how teachers' perceptions align with student and family goals for student learning. In addition, not many respondents spoke about using computer science education for social change. Deeper study around this topic could potentially benefit the teaching field, as teachers would benefit from best practices around how to implement this component in their teaching.

Conclusion

Overall, this study found that elementary computer science teachers think about equity in two ways: opportunity and differentiation, and listed their goals as: providing access to opportunity, promoting positive identification with computer science, empowering students to use computer science for social change, as well as increasing academic achievement, and provided strategies for making their practices more equitable. This study added to the growing body of research on the topic of elementary computer science education, and supported and extended extant research. From these conclusions, recommendations for current implementation and practice and future research were shared. These recommendations included providing teachers training in integrating ideas around who holds the power in technology, expanding teachers' views on what constitutes equity in computer science education, and supporting teachers in leveraging computer science instruction to promote academic achievement. Teachers in this study expressed their interest in supporting students to acquire computer science skills and create equitable computer science educational pathways.

Appendix A

Interview #1 Protocol

Hello, thank you for your willingness to participate in this interview. My name is Melissa Toohey. I'm currently a doctoral student at UCLA.Your participation in interviews is voluntary. I will do their best to make sure that your private information is kept confidential. Information about you will be handled as confidentially as possible, but participating in research may involve a loss of privacy and the potential for a breach in confidentiality. Study data will be physically and electronically secured. As with any use of electronic means to store data, there is a risk of breach of data security.

You can choose whether or not you want to be in this study, and you may withdraw your consent and discontinue participation at any time. Whatever decision you make, there will be no penalty to you, and no loss of benefits to which you were otherwise entitled. You may refuse to answer any questions that you do not want to answer and still remain in the study. No data collected during this study, including de-identified data will be shared for future research.
If you have questions about your rights as a research subject, or you have concerns or suggestions and you want to talk to someone other than the researchers, you may contact the UCLA OHRPP by phone: (310) 206-2040; by email: participants@research.ucla.edu or by mail: Box 951406, Los Angeles, CA 90095-1406. If you have questions or concerns about the study, you may contact my faculty sponsor, Prof. Sandoval, at sandoval@geis.ucla.edu

The purpose of the research interviews is to add to what we know about elementary school computer science implementation and to explore teachers' perspectives on computer science instruction.

Everything you share with me will be strictly confidential, but I may use your insights, without revealing your identity, to draw conclusions for my study. This interview will take approximately 30-60 minutes on Zoom. Do you consent to participating in the study and being recorded?

1. Tell me about your school.

2. Tell me about your students.
    a. Tell me about the gender and racial demographics of your students.
    b. Probe for number of students
    c. Probe for grade levels, if more than one
3. Tell me about your role at your school.
    a. Tell me whatever you think is important about where you teach.
    b. Probe for details about school: rural, urban, suburban. High or low SES? Title I?
4. How did you get into teaching computer science?
    a. How often?
    b. Integrated or stand alone?
5. What do you hope your students get out of computer science this year?
6. What are your goals for your computer science teaching?

Thank you! I'd like to have a conversation about how you interpret this lesson based on your goals for teaching and your students. The remaining questions will focus on the lesson plan you submitted prior to this interview.

7. What are your goals for this lesson?
8. Earlier you said one of your teaching goals is _____, tell me about how this lesson accomplishes that.

9. Earlier you said one of your student goals is _____, tell me about how this shows up in this lesson.

10. How do your students tend to respond to this lesson?

    a. What do you notice about the kids that seem more engaged than others?

    b. What do you notice about the kids that seem less engaged than others?

    c. Why do you think that is?

11. On a scale of 1-5, (one being not at all, and 5 being very well) how well do you think this lesson accomplishes the goals we've talked about so far? I'm interested in what you think works well and what you think might work better.

12. Is there anything else about this lesson that we haven't discussed that you'd like to share?

13. I'm finding teachers through word of mouth. Do you know of anyone that teaches CS at the elementary level that might be willing to participate in this study?

Appendix B

Interview #2 Protocol

Hi [Name], it's great to be talking to you again! Before we jump in, I just want to remind you that:

- This session will be recorded.
- Your identity will be protected and your responses will be confidential.
- You can choose not to answer any questions and remain in the study.
- You can stop the interview at any time without any consequences.

For our conversation today, I'd like to talk to you about how you think about equity in computer science. I'm sure you know that the workforce is white and male dominated, and there are people in the CS education world that are working to get more girls and students of color into CS. I want to share core components of a framework with you, and look at how and if you incorporate them into your own teaching. This comes from the Kapor Center, and they've been promoting CS education for a long time. Have you heard of them?

Before we start, how do you think about equity in CS teaching? What does that mean to you? Why do you think that?

Let's go through the framework. I'm most interested in understanding your response to it.

1. The first component is acknowledging the racism and white supremacy in computer science and inluding anti-racist practices and the de-centering of whiteness. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I'd like to understand how you think about it. What do you think about this core component?
    a. What parts resonate with you?
    b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?
    c. Can you give me an example so that I know what you mean?
    d. What would it look like in the lesson we talked about? Thank you. Let's go to the next one.
2. The second component is incorporating families, communities, cultures, and assets into the design of the curriculum, classrooms, and learning opportunities. Families are sought out and included in the construction of computer science activities. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I'd like to understand how you think about it. What do you think about this core component?
    a. What parts resonate with you?
    b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?
    c. Can you give me an example so that I know what you mean?

d. What would it look like in the lesson we talked about? Thank you. Let's go to the next one.

3. The third component is about pedagogy and curriculum that is rigorous, standards-aligned curriculum, and are authentic to students' experiences and cultures. High expectations are set for all students. Current and historical socio/political contexts are examined. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I'd like to understand how you think about it. What do you think about this core component?
    a. What parts resonate with you?
    b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?
    c. Can you give me an example so that I know what you mean?
    d. What would it look like in the lesson we talked about? Thank you. Let's go to the next one.

4. The fourth component is about student voice, agency and self-determination being valued, encouraged, incorporated into the learning process. Students' ideas and input are utilized to co-create classroom instruction. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I'd like to understand how you think about it. What do you think about this core component?
    a. What parts resonate with you?

b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?
c. Can you give me an example so that I know what you mean?
d. What would it look like in the lesson we talked about? Thank you. Let’s go to the next one.

5. The fifth component is about student identity and incorporating meaningful learning to ensure belonging to students of all backgrounds. Individual and collective identity exploration opportunities are utilized to create inclusive classrooms. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I’d like to understand how you think about it. What do you think about this core component?
    a. What parts resonate with you?
    b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?
    c. Can you give me an example so that I know what you mean?
    d. What would it look like in the lesson we talked about? Thank you. Let’s go to the next one.
6. The final component is about intentionally incorporating a diverse variety of CS experts into classroom learning and instruction. Specific efforts are made to identify role models from diverse identities, backgrounds, careers, and trajectories. Just to be clear, I'm not advocating for this, but this organization is advocating for it, and I’d like to understand how you think about it. What do you think about this core component?

a. What parts resonate with you?

b. Is this something you already think about in your CS teaching? If so, how? If no, how would you try to include this in your teaching?

c. Can you give me an example so that I know what you mean?

d. What would it look like in the lesson we talked about? Thank you. Let's go to the next one.

Anti-Racist Practices

- Racism and white supremacy in computer science is acknowledged.
- Anti-racist practices and de-centering of whiteness are enacted.

Inclusion of Families, Communities,Cultures, and Assets

- These elements are incorporated into the design of curriculum, classrooms, and learning opportunities.
- Family and community members are sought out and included in the construction of computer science learning and activities.

Pedagogy and Curriculum

- These are rigorous, aligned to standards, and are authentic to student experiences, experiences, and cultures.
- High expectations are set for all students.
- Current and historical socio-political contexts are examined.

Student Voice, Agency, and Self-determination

- These elements are valued, encouraged and incorporated in the learning process.
- Student ideas and input are actively solicited to co-create classroom instruction.

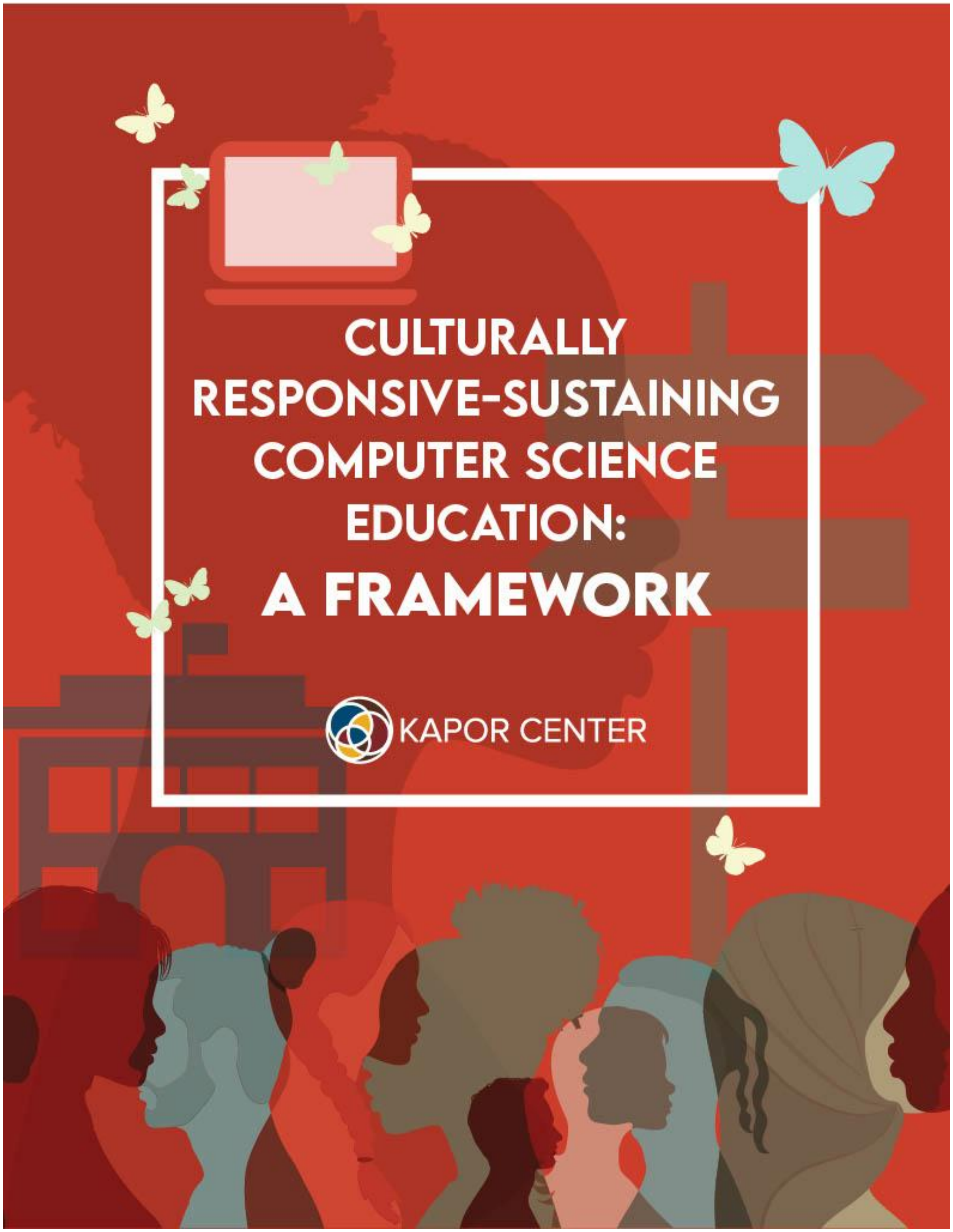
Student Identity

- Meaningful learning experiences and ensuring belonging are provided to students of all backgrounds.
- Individual and collective identity exploration opportunities are utilized to create inclusive classrooms.

Incorporation of Diverse CS experts into Classroom Learning

- Intentional exposure to diverse professionals and careers is incorporated into instruction.
- Specific efforts are taken to identify role models from diverse identities, backgrounds, careers, and trajectories.

Appendix C (Kapor Center, 2021)


CULTURALLY
RESPONSIVE-SUSTAINING
COMPUTER SCIENCE
EDUCATION:
A FRAMEWORK
KAPOR CENTER

June 24, 2021

Culturally Responsive-Sustaining CS Education Framework Team: Kalisha Davis (Project Director, Kapor Center), Shana V. White (Senior Associate, Kapor Center), Dinah Becton-Consuegra (Chief of Staff and Learning, Kapor Center), Allison Scott (CEO, Kapor Center)

National Advisory Board: Dr. Aman Yadav (Michigan State), Dr. Yolanda Sealy-Ruiz (Teachers College, Columbia), Dr. Bryan Brown (Stanford), Dr. Patricia Garcia (University of Michigan), Ruth Blackhawk Cameron (American Indian Science and Engineering Society), Bryan Twarek (Computer Science Teachers Association), Lien Diaz (Constellations Center, Georgia Tech), Olatunde Sobomehin (StreetCode Academy), Shana V. White (Kapor Center)

National Student Advisory Board: All'ayah Jarmon, Emilee Coxsey, Emily Jordan, Eyiara Oladipo, Ezra Feleke, Myles Mackie, Olamiposi Ajao, Olatayo Sobomehin, Zuriel Johnson

Framework Reviewers: Dr. Nicol Howard, Dr. Rosa Perez-Isiah, Dr. Kimberly Scott, Dr. Richard Ladner, Monise Seward, shea martin, Michelle G. Lee, Rebecca Luebker, Cindy Wong, Dr. Abigail Joseph, Shanti Coaston, Dulce-Marie Flecha, Owen Peery, Christopher Barrious, 2020-2021 CSTA Equity Fellows

Acknowledgements: The authors of the framework would like to extend sincere thanks, gratitude, and appreciation for the decades of work by leading scholars on frameworks of culturally relevant, responsive, and sustaining pedagogy as a strategy to improve outcomes for students from underrepresented and marginalized backgrounds. We appreciate the time and dedication of the national advisory board members, the student advisory board members, and the framework reviewers, who generously contributed their expertise to this project. This project is supported by Code With Google and Google.org through the Equitable Computer Science Curriculum Initiative and the Kapor Foundation.

KAPOR CENTER
kaporcenter.org

BACKGROUND

Computer Science as a Foundational Literacy

As technological innovation and advancement continues to impact all industries and sectors of the economy, influencing every aspect of our lives including how we work, learn, and communicate, computational literacy is critical for all students. Equitable access to computer science education is a critical national priority to address racial and economic disparities in the tech sector, to drive creativity, innovation and problem-solving, to and prepare a diverse tech workforce to meet the future needs of our economy, and to ensure students develop core computing literacies, while understanding social, cultural, and political impacts of technology.

Inequality in Computer Science Education

Despite the increased significance of computer science education, computational thinking, and computing literacy across all fields and occupations, access to computer science education is unequally distributed by race, gender, socioeconomic status, and geography. Just 47% of high schools in the U.S. offer computer science courses. Black, Latinx, and Native American students and students from low-income communities are significantly less likely to have access to CS courses in their schools. Moreover, while Black, Latinx, and Native American students comprise 43% of U.S. K-12 enrollment, they represent just 23% of students taking AP CS courses. Girls also face similar disparities, with only 29% of girls taking AP CS courses despite representing a greater percentage of the K-12 population. Black, Latinx, Native American and Pacific Islander students, low-income students, students with disabilities, and girls face additional barriers that impede their opportunity to participate in CS coursework. These barriers range from stereotypes about computing, lack of understanding of the discipline of computer science, to lower expectations for success, lack of inclusion and belonging in CS classroom environments, and lack of engaging and culturally relevant curriculum. Inequality in computer science education is situated within a context of significant historical inequality across societal institutions broadly, and in education specifically. Disparities in school funding and access to resources, such as broadband and technology devices, qualified and certified teachers, and rigorous STEM and computer science courses, limit students' opportunities for educational success. More broadly, a history of systemic racism and inequality negatively impacts outcomes in education, employment, housing, income, wealth, health, and life expectancies of communities of color.

Toward an Equitable Future for Computer Science Education

While much of the focus of broadening participation in computer science efforts has centered around increasing access to courses, access alone is insufficient and significant disparities remain. We believe that a multi-pronged approach centering racial justice is required to ensure meaningful participation, success, and matriculation in computer science education for students from all demographic backgrounds to close racial, gender, and socioeconomic equity gaps. This approach must address foundational educational disparities, create equitable policies at the federal, state and local levels, and invest deeply in the development of teachers, pedagogical practices, and curriculum that reflect and sustain students' cultures, experiences, and interests. Yet, we know from our survey of 3,700 CS teachers that less than 60% of CS teachers felt equipped to utilize culturally relevant pedagogical practices, believed existing curricular resources were culturally relevant, or felt confident incorporating critical discussions of computing's role in society and driving inequality. Toward that end, this empirically and theoretically-driven framework articulates an approach for designing and implementing equitable and culturally responsive pedagogical practices within computer science classrooms.

A FRAMEWORK FOR CULTURALLY RESPONSIVE-SUSTAINING CS EDUCATION:

Recognizing the need for a research-driven framework to design and implement equitable and culturally responsive-sustaining pedagogy within computer science classrooms, we have developed the **Culturally Responsive-Sustaining CS Framework** for K-12 computer science education. The **Culturally Responsive-Sustaining CS Framework** builds upon decades of theory and research on culturally relevant and responsive pedagogy across disciplines and was developed in partnership with researchers, practitioners, teachers, students, and other education advocates. The framework is intended to guide teacher preparation and professional development, curriculum development, and policies for developing a robust pipeline of CS teachers that will ultimately ensure greater adoption of culturally sustaining practices within computer science classrooms, close equity gaps in computer science, and improve the outcomes of marginalized students in computer science education. It is our intention that this framework will help to move the needle on equity in computer science education. We anticipate that the framework and its utilization will continue to evolve and be refined over time.

The process for developing, refining, and reviewing the framework was completed over a period of several months during 2020 and 2021. The project team reviewed relevant literature and co-created a draft framework with members of the National Advisory Board (which consisted of nine dynamic thought leaders with expertise in racial equity, social justice, community building, and CS education). The framework draft was then reviewed and edited by the National Advisory Board, the National Student Leadership Team (comprised of nine high school students representing nine communities in seven states); CSTA Equity Fellows, K12 computer science teachers, national experts in equity, inclusive and culturally responsive K-12 teaching; and other leaders and scholars within the national K-12 computer science education community.

There are two elements to this framework:

01

A **shared definition** of culturally responsive-sustaining computer science classroom pedagogy

02

Articulated **core components** for implementing culturally responsive-sustaining computer science pedagogy

Definition: Culturally Responsive-Sustaining CS Pedagogy

Culturally responsive-sustaining computer science pedagogy is situated within a context of racial, socioeconomic, and gender inequality in K-12 CS education. It articulates a strategy to move beyond increasing access to computer science courses and ensure all students have the opportunity to be inspired and engaged in computing education, develop critical computational skills, and have equitable opportunities to pursue computing careers and contribute to technological innovation. **Culturally responsive-sustaining computer science pedagogy ensures that students' interests, identities, and cultures are embraced and validated, students develop knowledge of computing content and its utility in the world, strong CS identities are developed, and students engage in larger socio-political critiques about technology's purpose, potential, and impact.** Culturally responsive-sustaining computer science pedagogy includes: the teacher's instructional practice, the curriculum, resources, and activities used in the classroom, as well as the instructional design practices utilized by the teacher. Culturally responsive-sustaining computer science pedagogy is necessary but not sufficient to achieve equity in computer science education. It must be implemented alongside broader solutions to dismantle racism and inequity in education, employment, health, and the environment, all of which disproportionately negatively impact marginalized communities.

Six Core Components:

The Six Core Components of the framework serve as guidance for any educators seeking to create culturally sustaining, equitable, and inclusive K-12 computer science classrooms. The core components are complemented by additional details about ways that these components can be enacted. It is our goal that a range of K-12 educators, including teachers, instructional coaches, administrators, and curriculum providers utilize these six core components to directly inform their instructional practices, curriculum development, and approaches to implementing equitable K-12 computer science education.

CORE COMPONENT	COURSE OF ACTION
01 The role of racism and white supremacy and its manifestation in computer science education is understood and acknowledged. Anti-racist practices and the decentering of whiteness are enacted within computer science courses and classrooms.	Educators explore their own identities (racial, gender, cultural, ethnic, linguistic, religious, socioeconomic, etc.) and their positions of privilege and power/oppression
	Educators demonstrate awareness of white supremacy and racism in education, computing, and CS classrooms as well as commit to ongoing learning to understand systemic racism as a part of their commitment to anti-racist and trauma informed pedagogy
	Educators actively use language to call out racism and decenter whiteness in CS courses
	Educators explicitly teach and engage in anti-racist/anti-bias and trauma-informed practices in CS
02 Inclusive and equitable classroom cultures are co-created to cultivate meaningful learning experiences for students and ensure belonging for students from all backgrounds. Individual and collective identity exploration are utilized as tools to create inclusive classrooms, ensure belonging for all students, and ensure equity.	Educators actively and intentionally confront and dispel stereotypes and biases about the abilities and skills of students from groups marginalized in CS
	Educators actively explore, understand, and reflect upon their own identities, positionality, power, and privilege, and how these constructs reside/operate within society and computer science
	Educators honor and affirm students' intersecting identities within curriculum, instructional practices, and classroom culture and support students' navigation of CS and society at large
	Educators intentionally recruit students with disabilities; Black, Native American, and Latinx students; girls; and non-binary students into CS courses
	Educators help students explore their identities to develop CS projects that reflect their passions and interests
	Educators deliberately establish an accessible classroom community that recognizes, respects, and includes the voices, ideas, needs, and perspectives of all students by engagement in consistent class check-ins, advisory, and student feedback sessions

CORE COMPONENT	COURSE OF ACTION
03 Pedagogy and curriculum is rigorous — aligned to K-12 CS standards and high expectations are set for all students; relevant — authentic to students' experiences, interests, and cultures; and examines current and historical socio-political contexts within which CS is situated.	Educators ensure curriculum is high-quality, rigorous, challenging, and aligned to state and national standards
	Educators support students in learning about the history of their respective communities, honor their ethnicities and cultures, and incorporate their cultures, interests, and passions into the learning process
	Educators actively seek out vetted resources and regular opportunities to learn about the current and historical cultures of their students
	Educators utilize pedagogy and curriculum which equips students to critically examine technology and interrogate its role in society as well as its ethical, political, and societal implications
04 Student voice, student agency, and self-determination are valued, encouraged, and incorporated during the CS learning process. Student ideas and input are actively solicited from students to co-create classroom instruction.	Educators incorporate student voices and perspectives throughout the curriculum and classroom experience, engaging them as cultural experts
	Educators engage students as emerging experts to lead activities, support peer-to-peer teaching and learning, and encourage ongoing feedback
	Educators honor and respect the diverse ways that students process and learn information, striving to be mindful and inclusive in their engagement

CORE COMPONENT	COURSE OF ACTION
05 Families and communities — and their cultures and assets — are incorporated into the design of CS curriculum, classrooms, and learning opportunities. Families and community members are intentionally sought out and included in the construction of CS classroom learning and activities.	Educators value and consult with families and community members and incorporate their perspectives into the CS classroom
	Educators partner with community-based organizations to build interest in CS among students and families and encourage the learning of CS inside and outside of the classroom
	Educators encourage and invite families and communities to engage in learning CS for their own knowledge and growth, as well as to support student learning
	Educators align CS content and instruction with in-school and out-of-school experiences, cultures, and perspectives
06 A diverse variety of experts are incorporated into the classroom (including researchers, community members, entrepreneurs, and tech leaders) to intentionally expose students to a variety of computing professionals and careers. Specific efforts are taken to identify role models from diverse identities, backgrounds, careers, and trajectories.	Educators expose students to a range of computing and technology-related careers, programs, and opportunities that are aligned to student interests
	Educators actively build relationships with members of the local and national tech community who can lend their knowledge and expertise to the classroom experience
	Educators actively seek out and recruit diverse guest speakers and experts representing underrepresented or marginalized groups in computing
	Educators leverage a variety of tech tools to introduce students to industry professionals and career pathways within their classroom, especially when in-person opportunities are a challenge

ABOUT US

The Equitable CS Initiative is a multi-year endeavor designed to create equitable computer science education experience for children and youth in grades K-12. We will achieve this through intentional and collaborative efforts that dismantle racism and bias; center CS learning on student assets, culture, and experience; and elevate the value that families and communities bring to the learning process. There are six different elements in this project: (1) A ***shared definition*** of culturally sustaining computer science classroom pedagogy, (2) An articulated set of ***core components*** for implementing culturally sustaining computer science pedagogy, (3) An articulated set of ***competencies*** for students in the computer science classroom, (4) A series of ***rubrics/scorecards*** to examine whether culturally sustaining instruction is being implemented at the classroom, school, and system level, (5) A set of ***modules*** providing a comprehensive examination of each core practice of the framework, and (6) A virtual ***professional learning community*** for teachers to share, learn, and grow while implementing this framework. This initiative is guided by an Advisory Board of CS education experts, a student leadership team, and is informed by and builds upon empirical and theoretical research literature. Ultimately, we aim to ensure CS educators are equipped with resources and strategies needed to create and develop engagement, identity, and persistence in computing pathways for Black, Latinx, Native American, gender non-binary students, and girls to ensure they have equitable opportunities to pursue computing pathways in K-12, college, and career.

KAPOR CENTER

The Kapor Center aims to enhance diversity and inclusion in the technology and entrepreneurship ecosystem through increasing access to CS and STEM education, advancing diversity and inclusion in tech companies, and investing in community organizations, diverse entrepreneurs, and gap-closing social ventures. The Equitable CS Initiative is one of our key initiatives and part of a broader focus on equity in computer science education.

APPENDIX 1. KEY TERMS

Culturally Relevant Pedagogy: Culturally relevant pedagogy includes three core principles: (1) academic achievement, (2) cultural competence, and (3) critical consciousness. Culturally relevant teaching must develop students academically, demonstrate a willingness to nurture and support cultural competence, and support the development of a sociopolitical or critical consciousness (Ladson-Billings, 1995).

Culturally Responsive Pedagogy: Culturally responsive pedagogy is defined as a framework for teaching that uses cultural knowledge, prior experiences, frames of reference, and performance styles of ethnically diverse students to make learning encounters more relevant and effective for them (Gay, 2010). Culturally responsive pedagogy is enacted in instructional techniques, instructional materials, student-teacher relationships, classroom climate, and self-awareness to improve learning for students.

Culturally Sustaining Pedagogy: Culturally sustaining pedagogy builds upon culturally relevant and responsive pedagogy. It seeks to perpetuate and foster linguistic, literate, and cultural pluralism (i.e., preserves the unique cultural identities and histories of all students) as part of the democratic project of schooling and sustains the cultural practices of communities of color (Paris and Alim, 2017).

Culturally Responsive Computing: Culturally responsive computing, whose roots are based in culturally responsive teaching is defined as, "a strategy constructed to engage culturally and linguistically diverse youth, concerned with empowerment, transformation, validation, comprehension, multidimensionality and emancipation." CRC proposes five tenets: (1) All students are capable of digital innovation, (2) The learning context supports transformational use of technology, (3) Learning about one's self along various intersecting sociocultural lines allows for technical innovation, (4) Technology should be a vehicle by which students reflect and demonstrate understanding of their intersectional identities, (5) Barometers for technological success should consider who creates, for whom, and to what ends rather than who endures socially and culturally irrelevant curriculum (Scott, Sheridan, and Clark, 2015).

Culturally Responsive-Sustaining Computer Science Pedagogy: Culturally responsive-sustaining CS pedagogy builds upon culturally relevant, responsive, and sustaining pedagogies and culturally responsive computing to articulate a clear framework for equitable and inclusive computer science teaching and learning. Culturally responsive-sustaining computer science pedagogy ensures that students' interests, identities, and cultures are embraced and affirmed, students develop knowledge of computing content and its utility in the world, students develop strong CS identities, student, family, and community voices and experiences are validated, and that students engage in the larger socio-political critiques about technology's purpose, potential, and impact. Culturally responsive-sustaining computer science pedagogy offers six core practices: (1) understanding the role and impact of racism and inequality in CS, (2) developing classroom culture must be inclusive and equitable for all students, (3) usage of high quality CS curriculum that is relevant to the experiences and interest of students, (4) student voice is validated, affirmed, and amplified within the CS classroom, (5) intentional inclusion of students' families and the school community within the CS classroom (6) diverse variety of experts and computing role models purposefully incorporated into the classroom learning experience.

Computer Science: Computer science is defined by the Association for Computing Machinery as the "study of computers and algorithmic processes, including their principles, their hardware and software designs, their implementation, and their impact on society." Computing is a broad term defined by the Association for Computing Machinery as "any goal-oriented activity requiring, benefiting from, or creating computers... including five sub-disciplines of computer science, computer engineering, information systems, information technology and software engineering." The terms computing, computer science, and the abbreviation "CS" are used interchangeably throughout this framework.

APPENDIX 2. REFERENCES

Computer Science Education Frameworks:

- K-12 Computer Science Framework
- CSTA Standards for CS Teaching
- CS for CA: CS Equity Guide
- Code.org Nine Policy Ideas to Make CS Fundamental in K-12
- The Teacher Accessibility, Equity & Content (TEC) Rubric for Evaluating Computing Curricula
- CS Visions: Diverse Visions of CS Education in Practice

Culturally Relevant and Responsive Frameworks and Rubrics:

- Culturally Responsive-Sustaining Education Framework (New York State Department of Education)
- Culturally Responsive Curriculum Scorecard (NYU/Steinhardt)

Culturally Relevant/Responsive/Sustaining Research and Theory

- Toward a Theory of Culturally Relevant Pedagogy (Ladson-Billings, 1995)
- Culturally Responsive Teaching: Theory, Research, and Practice (Gay, 2000)
- Culturally Responsive Teaching and the Brain (Hammond, 2014)
- Culturally Sustaining Pedagogy : A Needed Change in Stance, Terminology, and Practice (Paris, 2012)
- Culturally Sustaining Pedagogy: A Critical Framework for Centering Communities (Alim, Paris & Wong, 2020)
- Cultivating Genius: An Equity Framework for Culturally and Historically Responsive Literacy (Muhammad, 2020)
- Guide to Racial Justice & Abolitionist Social and Emotional Learning (Abolitionist Teaching Network)
- We Want To Do More Than Survive: Abolitionist Teaching and the Pursuit of Educational Freedom (Love, 2020)
- Cultural Revitalizing Pedagogies: Critical Culturally Sustaining/Revitalizing Pedagogy and Indigenous Education Sovereignty, (McCarty & Lee, 2014)
- But That's Just Good Teaching! The Case for Culturally Relevant Pedagogy (Ladson-Billings, 2009)
- Culturally Responsive Teaching as an Ethics and Care Based Approach to Urban Education (Shevalier & McKenzie, 2012)
- Educating All Students: Creating Culturally Responsive Teachers, Classrooms, and Schools (Brown, 2014)
- Operationalizing Culturally Relevant Pedagogy: A Synthesis of Classroom-Based Research (Morrison et al., 2008)

Culturally Relevant/Responsive/Sustaining CS/STEM Education

- Culturally Responsive Computing: A Theory Revisited (Scott, et al., 2015)
- Culturally Relevant CS Pedagogy: From Theory to Practice (Madkins et al., 2020)
- Culturally Responsive Computing As A Brokerage: Toward Asset Building With Education-Based Social Movement (Lachney, 2016)
- Toward Culturally Responsive Computing Education (Eglash et al., 2013)
- Animal tlatoque: Attracting Middle School students to Computing Through Culturally-Relevant Themes (Franklin et al., 2011)
- Attracting Native Americans to Computing (Varma, 2009)
- Bridging the Diversity Gap in Computer Science with a Course on Open Source Software (Weng & Murphy, 2018)
- Assuming Brilliance: A decriminalizing approach to education African American and Latino boys in elementary school STEM settings (Basile & Lopez, 2018)
- Creating Access and Opportunity: Preparing African-American male students for STEM trajectories PreK-12 (Wright et al.,2016)
- Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms (Yadav, 2016)
- Cultural stereotypes as gatekeepers: increasing girls' interest in computer science and engineering by diversifying stereotypes (Cheryan et al., 2015)
- Democratizing computer science knowledge: transforming the face of computer science through public high school education (Ryoo et al., 2012)
- Increasing Diversity in K-12 Computer Science: Strategies from the Field (Goode, 2008)
- This is What Diversity Looks Like: Making CS Curriculum Culturally Relevant for Spanish-speaking Communities (Miranda et al., 2019)

Equity, Inclusivity and Racism in Schools and CS/STEM

- Ambient Belonging: How Stereotypical Cues Impact Gender Participation in Computer Science (Cheryan et al., 2009)
- Avoiding Racial Equity Detours (Gorski, 2019)
- An Ecological Model of STEM Education: Operationalizing STEM for all (Basham et al., 2010)
- Guide to Inclusive Computer Science Education (Microsoft, 2019)
- Eliminating Racism in the Classroom (Morgan, 2021)
- Embedding and Sustaining Inclusive Practices in STEM (McPherson et al., 2019)
- Exploring Intersectionality in Education: The Intersection of Gender, Race, Disability, and Class (Petersen, 2006)
- Good Teaching? An Examination of Culturally Relevant Pedagogy as an Equity Practice (Schmeichel, 2012)
- Humanizing Pedagogy: Beliefs and Practices on the Teaching of Latino Children (Huerta, 2011)
- Inclusive STEM High School Design: 10 Critical Components (Peters-Burton et al., 2014)
- Influencing Middle School Girls to Study Computer Science Through Educational Computer Games (Stewart-Gardiner, 2013)
- Schools as Racial Spaces: Understanding and Resisting Structural Racism (Blaisdell, 2016)
- Take Space, Make Space: How Students Use Computer Science to Disrupt and Resist Marginalization in Schools (Ryoo et al, 2020)
- The Computer Science Teacher Landscape: Results of a Nationwide Teacher Survey (Koshy et al, 2021)
- Understanding STEM Education and Supporting Students Through Universal Design for Learning (Basham & Marino, 2013)
- Weaving Cultural Relevance and Achievement Motivation into Inclusive Classroom Cultures (Kumar et al., 2018)

Ethical and Sociopolitical Considerations in CS

- Ethics, Identity, and Political Vision: Toward a Justice-Centered Approach to Equity in Computer Science Education (Vakil, 2018)
- Embedded EthiCS: Integrating Ethics Broadly Across Computer Science Education (Groscz et al., 2018)
- It's About Power: A Call to Rethink and Equity in Computing Education (Vakil & Higgs, 2019)
- Culturally Situated Design Tools: Ethnocomputing from Field Site to Classroom (Eglash et al., 2006)
- Exploring Politicized Trust in a Racially Diverse Computer Science Classroom (Vakil & Mckinney de Royston, 2019)
- Let's Teach Computer Science Majors to be Good Citizens. The Whole World Depends on It. (Núñez et al., 2021)
- Race After Technology: Abolitionist Tools for the New Jim Code (Benjamin, 2020)
- Algorithms of Oppression: How Search Engines Reinforce Racism(Noble, 2018)

Developing Student Voice, Agency, and Identity in the Classroom and Computing:

- Connecting Computer Science Education to Students' Passions: A Critical Step Toward Supporting Equity in CS Education (Ryoo et al, 2019)
- A threat in the air. How stereotypes shape intellectual identity and performance (Steele, 1997)
- Leveraging Technology: How Black girls enact critical digital literacies for social change (Garcia et al, 2020)
- A qualitative investigation of factors promoting the retention and persistence of students of color in STEM (Palmer et al., 2011)
- Willing, Able and Unwanted: High School Girls' Potential Selves in Computing (Kelly et al., 2013)

Community Perspectives and Funds of Knowledge

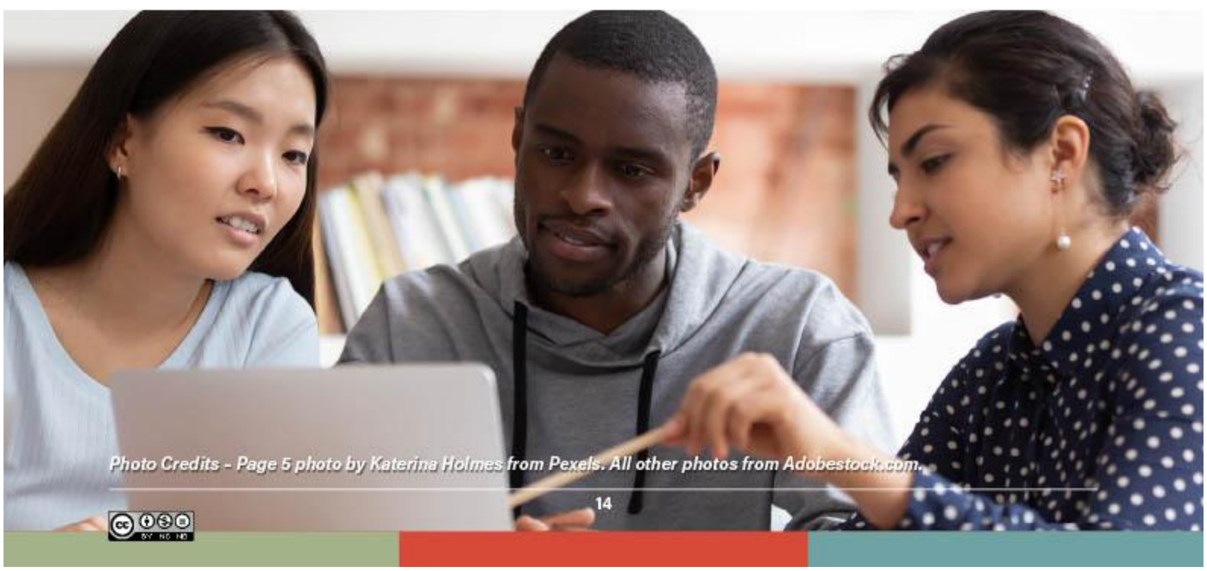
- Leveraging Local Resources and Contexts for Inclusive CS Classrooms: Reflections from Experienced High School Teachers Implementing E-Textiles (Shaw et al, 2020)
- Elaborating Funds of Knowledge: Community-Oriented Practices in International Context (Moll, 2019)
- Parent Involvement, African American Mothers, and the Politics of Education Care (Cooper, 2009)
- Prevention through Collaboration: Family Engagement With Rural Schools and Families Living in Poverty (Blitz et al., 2018)
- Funds of Knowledge for Teaching: Using a Qualitative Approach to Connect Homes and Classrooms (Moll et al., 2005)
- Funds of Knowledge and Discourse and Hybrid Space (Barton, 2008)

Project-Based Pedagogy & Curriculum

- Project-Based Teaching Practices for K-12 (PBLWorks)
- Attitudes towards STEM in a project-based learning environment (Tseng et al., 2013)
- How STEM Project-Based Learning affects high, middle, and low achievers differently: The impact of student factors on achievement (Han et al., 2014)
- Learning-Goals-Driven Design Model: Developing Curriculum Materials that Align with National Standards and Incorporate Project-Based Pedagogy (Krajcik et al., 2008)
- Middle School Students' Self-Efficacy, Attitudes, and Achievement in a Computer-Enhanced Problem-Based Learning Environment (Liu et al., 2006)
- Project-Based Learning: A Primer (Solomon, 2003)
- Using Problem-Based Learning Software with At-Risk Students (Samsonov et al., 2006)

APPENDIX 3. ADDITIONAL RESOURCES

- AccessCSforAll
- Addressing Race and Trauma in the Classroom
- Common Sense Media Digital Citizenship Curriculum and Lessons
- CSforAll
- ComputerScience.org
- Computer Science Education Week
- Culture in the Classroom — PD Resource for Teachers, Teaching Tolerance
- Digital Learning Day Activities and Resources
- Discovering My Identity: Lesson Plan
- Embedded EthiCS
- Nepris
- REAL-CS Initiative
- RoadTripNation
- Tool Kit: Identity Development
- Universal Design for Learning



References

Asgari, S., Dasgupta, N., & Stout, J. G. (2012). When Do Counterstereotypic Ingroup Members Inspire Versus Deflate? The Effect of Successful Professional Women on Young Women's Leadership Self-Concept. *Personality and Social Psychology Bulletin*, *38*(3), 370–383. https://doi.org/10.1177/0146167211431968

Banilower, E., Smith, P. S., Malzahn, K., Plumley, C., Gordon, E., & Hayes, M. (2018). *Report of the 2018 NSSME+*. 442.

Bathgate, M., & Schunn, C. (2017). The psychological characteristics of experiences that influence science motivation and content knowledge. *International Journal of Science Education*, *39*(17), 2402–2432. https://doi.org/10.1080/09500693.2017.1386807

Bers, M. U. (n.d.). *Http://ecrp.uiuc.edu/v12n2/bers.html*. 20.

Bullard, R. (1983). "Solid Waste Sites and the Black Houston Community", Sociological Inquiry, 53, pp. 273-88. In *Environmental Justice* (pp. 25–40). Routledge.

Carlone, H. B., & Johnson, A. (2007). Understanding the science experiences of successful women of color: Science identity as an analytic lens. *Journal of Research in Science Teaching*, *44*(8), 1187–1218. https://doi.org/10.1002/tea.20237

Charleston, L. J., George, P. L., Jackson, J. F. L., Berhanu, J., & Amechi, M. H. (2014). Navigating underrepresented STEM spaces: Experiences of Black women in U.S. computing science higher education programs who actualize success. *Journal of Diversity in Higher Education*, *7*(3), 166–176. https://doi.org/10.1037/a0036632

Cheryan, S., Master, A., & Meltzoff, A. N. (2015). Cultural stereotypes as gatekeepers: Increasing girls' interest in computer science and engineering by diversifying stereotypes. *Frontiers in Psychology*, *6*. https://doi.org/10.3389/fpsyg.2015.00049

Coenraad, M., Mills, K., Byrne, V. L., & Ketelhut, D. J. (2020). Supporting Teachers to Integrate Computational Thinking Equitably. *2020 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, *1*, 1–2. https://doi.org/10.1109/RESPECT49803.2020.9272488

College Board AP Program Summary Report, 2018 Calculus AB & BC, Computer Science A, Computer Science Principles (p. 1). (n.d.).

Computer and Information Technology Occupations: Occupational Outlook Handbook: : U.S. Bureau of Labor Statistics. (n.d.). Retrieved November 10, 2019, from https://www.bls.gov/ooh/computer-and-information-technology/home.htm

Darling-Hammond, L. (2006). Constructing 21st-Century Teacher Education. *Journal of Teacher Education*, *57*(3), 300–314. https://doi.org/10.1177/0022487105285962

Duncan, C., Bell, T., & Atlas, J. (2017). What do the Teachers Think?: Introducing Computational Thinking in the Primary School Curriculum. *Proceedings of the Nineteenth Australasian Computing Education Conference on - ACE '17*, 65–74. https://doi.org/10.1145/3013499.3013506

Dwyer, H. A., Boe, B., Hill, C., Franklin, D., & Harlow, D. B. (2013). *Computational Thinking for Physics: Programming Models of Physics Phenomenon in Elementary School*. 133–136. https://www.compadre.org/per/items/detail.cfm?ID=13126

Eglash, R., Bennett, A., O'donnell, C., Jennings, S., & Cintorino, M. (2006). Culturally Situated Design Tools: Ethnocomputing from Field Site to Classroom. *American Anthropologist*, *108*(2), 347–362. https://doi.org/10.1525/aa.2006.108.2.347

Eglash, R., Gilbert, J. E., Taylor, V., & Geier, S. R. (2013). Culturally Responsive Computing in Urban, After-School Contexts: Two Approaches. *Urban Education*, *48*(5), 629–656. https://doi.org/10.1177/0042085913499211

Ehsan, H., Rehmat, A. P., & Cardella, M. E. (2020). Computational thinking embedded in engineering design: Capturing computational thinking of children in an informal engineering design activity. *International Journal of Technology and Design Education*. https://doi.org/10.1007/s10798-020-09562-5

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

Franklin, D., Conrad, P., Aldana, G., & Hough, S. (n.d.). Animal tlatoque: Attracting middle school students to computing through culturally-relevant themes. *In SIGCSE '11*.

Garvin, M., Killen, H., Plane, J., & Weintrop, D. (2019). Primary School Teachers' Conceptions of Computational Thinking. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 899–905. https://doi.org/10.1145/3287324.3287376

Gay, G. (n.d.). *Culturally Responsive Teaching 9780807758762 | Teachers College Press*. Retrieved March 23, 2022, from https://www.tcpress.com/culturally-responsive-teaching-9780807758762

Gomez, K., Lee, U.-S., & Amy, B. W. (2022). The Role of Teacher Beliefs, Goals, Knowledge, and Practices in Shaping Computer Science Education: Teacher Participation and Learning in the Codesign of Elementary Computer Science Curriculum. *Teacher Learning in Changing Contexts: Perspectives from the Learning Sciences*.

González-Patiño, J., & Esteban-Guitart, M. (2019). Affinity online: How connection and shared interest fuel learning. *Mind, Culture, and Activity*, *26*(4), 383–386. https://doi.org/10.1080/10749039.2019.1680695

Goode, J., Chapman, G., & Margolis, J. (2012). Beyond curriculum: The exploring computer science program. *ACM Inroads*, *3*(2), 47–53. https://doi.org/10.1145/2189835.2189851

Goode, J., & Margolis, J. (2011). Exploring Computer Science: A Case Study of School Reform. *ACM Transactions on Computing Education*, *11*(2), 12:1-12:16. https://doi.org/10.1145/1993069.1993076

Gorski, P. C. (2009). Insisting on Digital Equity: Reframing the Dominant Discourse on Multicultural Education and Technology. *Urban Education*, *44*(3), 348–364. https://doi.org/10.1177/0042085908318712

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

Gürbüz, H., Evlioğlu, B., Erol, Ç. S., Gülseçen, H., & Gülseçen, S. (2017). “What’s the Weather Like Today?”: A computer game to develop algorithmic thinking and problem solving skills of primary school pupils. *Education and Information Technologies*, *22*(3), 1133–1147. https://doi.org/10.1007/s10639-016-9478-9

Hidi, S., & Renninger, K. A. (2006). The Four-Phase Model of Interest Development. *Educational Psychologist*, *41*(2), 111–127. https://doi.org/10.1207/s15326985ep4102_4

Hiebert, J., Morris, A. K., Berk, D., & Jansen, A. (2007a). Preparing Teachers to Learn from Teaching. *Journal of Teacher Education*, *58*(1), 47–61. https://doi.org/10.1177/0022487106295726

Hiebert, J., Morris, A. K., Berk, D., & Jansen, A. (2007b). Preparing Teachers to Learn from Teaching. *Journal of Teacher Education*, *58*(1), 47–61. https://doi.org/10.1177/0022487106295726

How Vocational Education Got a 21st Century Reboot—POLITICO Magazine. (n.d.). Retrieved November 10, 2019, from https://www.politico.com/magazine/story/2019/09/12/vocational-tech-school-training-policy-228049

Howard, N. R. (2019). EdTech Leaders’ Beliefs: How are K-5 Teachers Supported with the Integration of Computer Science in K-5 Classrooms? *Technology, Knowledge and Learning*, *24*(2), 203–217. https://doi.org/10.1007/s10758-018-9371-2

Hynes, M. M., Moore, T. J., Cardella, M. E., Tank, K. M., Purzer, S., Menekse, M., & Brophy, S. P. (2016, June 26). *Inspiring Computational Thinking in Young Children's Engineering Design Activities (Fundamental)*. 2016 ASEE Annual Conference & Exposition. https://peer.asee.org/inspiring-computational-thinking-in-young-children-s-engineering-design-activities-fundamental

Israel, M., & Lash, T. (2020). From classroom lessons to exploratory learning progressions: Mathematics + computational thinking. *Interactive Learning Environments*, *28*(3), 362–382. https://doi.org/10.1080/10494820.2019.1674879

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 and Education*, *82*, 263–279. https://doi.org/10.1016/j.compedu.2014.11.022

Jones, B. D., Ruff, C., & Paretti, M. C. (2013). The impact of engineering identification and stereotypes on undergraduate women's achievement and persistence in engineering. *Social Psychology of Education*, *16*(3), 471–493. https://doi.org/10.1007/s11218-013-9222-x

Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, *52*, 200–210. https://doi.org/10.1016/j.chb.2015.05.047

Kapor Center. (2021). *EquitableCS*. Kapor Center. https://www.kaporcenter.org/equitablecs/

Kemph, J. P. (1969). Erik H. Erikson. Identity, youth and crisis. New York: W. W. Norton Company, 1968. *Behavioral Science*, *14*(2), 154–159. https://doi.org/10.1002/bs.3830140209

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 and Technology*, *29*(1), 174–188. https://doi.org/10.1007/s10956-019-09798-4

Kong, S., Chiu, M., & Lai, M. (2018). A study of primary school students' interest, collaboration attitude, and programming empowerment in computational thinking education. *Computers & Education*, *127*. https://doi.org/10.1016/j.compedu.2018.08.026

Legewie, J., & DiPrete, T. A. (2014). The High School Environment and the Gender Gap in Science and Engineering. *Sociology of Education*, *87*(4), 259–280. JSTOR.

Leonard, A. E., Dsouza, N., Babu, S. V., Daily, S. B., Jörg, S., Waddell, C., Parmar, D., Gundersen, K., Gestring, J., & Boggs, K. (2015). *Embodying and Programming a Constellation of*. *11*(2), 30.

Lim, M., & Barton, A. C. (2006). Science learning and a Sense of Place in a Urban Middle School. *Cultural Studies of Science Education*, *1*(1), 107–142. https://doi.org/10.1007/s11422-005-9002-9

Luo, F., Antonenko, P. D., & Davis, E. C. (2020). Exploring the evolution of two girls' conceptions and practices in computational thinking in science. *Computers & Education*, *146*, 103759. https://doi.org/10.1016/j.compedu.2019.103759

Margolis, J. (n.d.). *Stuck in the Shallow End | The MIT Press*. The MIT Press. Retrieved November 22, 2020, from https://mitpress.mit.edu/books/stuck-shallow-end

Margolis, J., & Fisher, A. (2002). *Unlocking the Clubhouse | The MIT Press*. The MIT Press. https://mitpress.mit.edu/books/unlocking-clubhouse

Master, A., Cheryan, S., & Meltzoff, A. N. (2016). Computing whether she belongs: Stereotypes undermine girls' interest and sense of belonging in computer science. *Journal of Educational Psychology*, *108*(3), 424–437. https://doi.org/10.1037/edu0000061

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

McLoughlin, C. (1999). Culturally responsive technology use: Developing an on-line community of learners. *British Journal of Educational Technology*, *30*(3), 231–243. https://doi.org/10.1111/1467-8535.00112

Menekse, M. (2015). Computer science teacher professional development in the United States: A review of studies published between 2004 and 2014. *Computer Science Education*, *25*(4), 325–350. https://doi.org/10.1080/08993408.2015.1111645

Migus, L. H. (n.d.). *Broadening Access to STEM Learning through Out-of-School Learning Environments*. 17.

Naphan-Kingery, D. E., Miles, M., Brockman, A., McKane, R., Botchway, P., & McGee, E. (2019). Investigation of an equity ethic in engineering and computing doctoral students. *Journal of Engineering Education*, *108*(3), 337–354. https://doi.org/10.1002/jee.20284

NASEM. (2018). *Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments*. The National Academies Press. https://doi.org/10.17226/24926

NASEM (2021a). *Cultivating Interest and Competencies in Computing: Authentic Experiences and Design Factors*. https://doi.org/10.17226/25912

NASEM. (2021b). Science and engineering in preschool through elementary grades: The brilliance of children and the strengths of educators. Washington DC: National Academies Press.

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

Rodriguez, S. L., & Lehman, K. (2017). Developing the next generation of diverse computer scientists: The need for enhanced, intersectional computing identity theory. *Computer Science Education*, *27*(3–4), 229–247. https://doi.org/10.1080/08993408.2018.1457899

Ryoo, J. (n.d.). Pedagogy that Supports Computer Science for All. *ACM Transactions on Computing Education*, *19*(4), 1–23.

Santo, R., DeLyser, L. A., Ahn, J., Pellicone, A., Aguiar, J., & Wortel-London, S. (2019). Equity in the Who, How and What of Computer Science Education: K12 School District Conceptualizations of Equity in 'CS for All' Initiatives. *2019 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*, 1–8. https://doi.org/10.1109/RESPECT46404.2019.8985901

Scott, K. A., Sheridan, K. M., & Clark, K. (2015). Culturally responsive computing: A theory revisited. *Learning, Media and Technology*, *40*(4), 412–436. https://doi.org/10.1080/17439884.2014.924966

Searle, K. A., & Kafai, Y. B. (2015). Boys' Needlework: Understanding Gendered and Indigenous Perspectives on Computing and Crafting with Electronic Textiles. *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, 31–39. https://doi.org/10.1145/2787622.2787724

Sengupta, P., & Farris, A. V. (2012). Learning kinematics in elementary grades using agent-based computational modeling: A visual programming-based approach. *Proceedings of the 11th International Conference on Interaction Design and Children*, 78–87. https://doi.org/10.1145/2307096.2307106

Stets, J. E., Brenner, P. S., Burke, P. J., & Serpe, R. T. (2017). The Science Identity and Entering a Science Occupation. *Social Science Research*, *64*, 1–14. https://doi.org/10.1016/j.ssresearch.2016.10.016

Sullivan, A., & Bers, M. U. (2016). Robotics in the early childhood classroom: Learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. *International Journal of Technology and Design Education*, *26*(1), 3–20. https://doi.org/10.1007/s10798-015-9304-5

Terry, L. (2011). Mathematical counterstory and African American male students: Urban mathematics education from a critical race theory perspective. *Journal of Urban Mathematics Education*, *4*(1), 23–49.

The Integrated Postsecondary Education Data System. (n.d.). Retrieved November 10, 2019, from https://nces.ed.gov/ipeds/

Toom, A. (2017). Teachers' Professional and Pedagogical Competencies: A Complex Divide between Teacher Work, Teacher Knowledge and Teacher Education. In D. Clandinin & J. Husu, *The SAGE Handbook of Research on Teacher Education* (pp. 803–819). SAGE Publications Ltd. https://doi.org/10.4135/9781526402042.n46

Vogel, S., Santo, R., & Ching, D. (2017). Visions of Computer Science Education: Unpacking Arguments for and Projected Impacts of CS4All Initiatives. *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, 609–614. https://doi.org/10.1145/3017680.3017755

Warped Reality: Joy Buolamwini: How Do Biased Algorithms Damage Marginalized Communities? (2021). In *TED Radio Hour (Podcast)*. NPR.

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

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

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*. https://www.tandfonline.com/doi/abs/10.1080/08993408.2018.1560550