Computer Science (CS) Education in Elementary Schools: Overcoming obstacles and looking at effective strategies to successful implementation

By Rajavi Shah

A research paper submitted in conformity with the requirements for the degree of Master of Teaching Department of Curriculum, Teaching and Learning Ontario Institute for Studies in Education of the University of Toronto.

Copyright by Rajavi Shah, April 2017
Abstract

This qualitative research study looks at the question: how do educators teach complex computer science concepts to elementary students in creative ways and what strategies do they use to engage their students in this field? Two educators who had experience teaching coding to students were approached via convenience sampling to participate in semi-structured interviews. The participants agreed to speak about their experiences, effective strategies, obstacles and benefits of implementing coding in the classroom. These interviews were transcribed, coded and analyzed for common emergent themes to help answer the research question. There were four main themes that emerged in this study: coding in the classroom has a wide range of benefits to students, teachers need many resources and support from staff to carry this work, educators face obstacles regarding lack of knowledge in topic and resources, and lastly to keep consistent with coding integration educators find motivation from their own interest and learning journey. After examining the findings, personal implications and implications for the educational community are discussed and recommendations for teachers, administration and faculties of education are suggested. Finally, further areas of research are suggested that allow for continued possibilities on exploration in coding education.

Keywords: Technology, Coding, 21st Century learning, Computer Science, STEM
Acknowledgements

I would like to acknowledge the efforts of the many people in my life that helped me continue my work and successfully finish my MTRP. My parents and my brother for being my constant supports and motivation to carry this work. My mom for sharing her ideas about being a teacher and her educational philosophy, my dad keenly trying to learn about my research and my brother for being my editor. Your support keeps me going.

Most importantly I would like to thank the staff at OISE - my professors Angela MacDonald and Sarah Cashmore as well as Erin Sperling for taking the time to read through my work and helping me improve it. Without your expertise and recommendations, this paper would be incomplete.

I would like to thank both my participants for sharing their wealth of knowledge with me and allowing me to share it with the world. I enjoyed seeing your dedication to your classrooms and your passion towards technology. It inspires me to become an educator like you who is willing to change and adapt to new times for the sake of their student’s future.

I want to thank my friends and colleagues- Lisa Cole, Matthew DeJong and Ryan Drakes for always being there to listen to my ideas, accepting them with open minds and helping me make them concrete. They themselves have tried their hand at coding education and show as much excitement as I do when trying it out.

I have learned a lot as an educator about coding and seen the education industry slowly adapting to this new idea, I look forward to seeing where and how this type of education will carry forward in Ontario.
# Table of Contents

Abstract ........................................................................................................................................... 2  

Acknowledgements ......................................................................................................................... 3  

Chapter 1: Introduction .................................................................................................................. 8  
  1.0. Introduction ............................................................................................................................. 8  
  1.1. Articulation of the Research Problem .................................................................................... 10  
  1.2. Purpose of the Study .............................................................................................................. 10  
  1.3. Research Questions ................................................................................................................ 11  
  1.4. Background of the Researcher .............................................................................................. 11  
  1.5. Preview of the Whole MTRP ................................................................................................ 12  

Chapter 2: Literature Review ......................................................................................................... 13  
  2.0. Introduction ........................................................................................................................... 13  
  2.1. What is Computer Science Education? .................................................................................. 14  
      2.1.1. Terminology ....................................................................................................................... 14  
      2.1.2. What does it look like at an elementary level? ................................................................. 14  
  2.2. Why Do We Need CS Education? ......................................................................................... 15  
      2.2.1. Preparing for the future .................................................................................................... 16  
      2.2.2. Application to other subjects ........................................................................................ 16  
  2.3. Students' Perspectives on CS Education .............................................................................. 17  
      2.3.1. The computer science stereotypes .................................................................................... 17  
      2.3.2. Female perspectives ......................................................................................................... 18  
      2.3.3. Male perspectives ............................................................................................................ 20  
  2.4. Teachers' Perspectives on CS Education .............................................................................. 21
2.4.1. Educators teaching CS ................................................................. 21
2.4.2. Resources available ..................................................................... 22

2.5. CS Education in Policy and Action .................................................. 23
2.5.1. CS education around the world ...................................................... 23
2.5.2. US policy .................................................................................... 24
2.5.3. Canada and provincial policy ....................................................... 25

2.6. Conclusion ...................................................................................... 25

Chapter 3: Research Methodology .......................................................... 26
3.0. Introduction ...................................................................................... 26
3.1. Research Approach and Procedures ................................................. 27
3.2. Instruments of Data Collection ........................................................ 28
3.3. Participants ..................................................................................... 29
  3.3.1. Sampling criteria ......................................................................... 29
  3.3.2. Sampling procedures ................................................................. 30
3.4. Participant Bios .............................................................................. 31
3.5. Data Analysis .................................................................................. 32
3.6. Ethical Review Procedures .............................................................. 32
3.7. Methodological Limitations and Strengths ...................................... 33
3.8. Conclusion ...................................................................................... 35

Chapter 4: Research Findings ................................................................. 35
4.0. Introduction to the Chapter ............................................................. 35
4.1. Coding in the Classroom has a Range of Positive Outcomes that Help Student Learning and Preparation for the Future According to Educators .................. 36
4.1.1. Motivation and engagement in students is an almost immediate result of having coding in the classroom .......................................................... 37

4.1.2. Participants acknowledge that coding can be integrated into any subject topic at any age and produce results ........................................................................................................ 38

4.1.3. Integrating coding in the classroom prepares students for 21st century learning. 40

4.2. Participants Mentioned that Teaching Coding is Dependent on the Availability or Knowledge of Various Resources to Help Support the Teaching Process .......... 41

4.2.1. Teachers frequently use online resources to support their work in integrating computer science education .............................................................. 42

4.2.2. Hardware resources were used to enhance coding learning in the classroom ...... 43

4.2.3. Participants relied on various people and organizational resources to develop their strategies in teaching coding ........................................................................ 44

4.3. Participants Acknowledge Various Challenges When Attempting to Integrate Coding into the Classroom ...................................................................................................................... 45

4.3.1. A lack of understanding of the topic is a challenge when trying to implement coding ................................................................. 45

4.3.2. Lack of resources pose a big challenge to teaching coding ...................... 46

4.3.3. Participants indicated that the lack of a standard curriculum is a challenge to teaching coding ........................................................................ 47

4.4. Participants Maintain a Commitment to Teaching Coding by Pursuing Their Natural Teaching Interests and a Lifelong Commitment to Learning .................... 49

4.4.1. Participants educate themselves using various mediums to learn new ways of teaching coding ........................................................................ 49
4.4.2. Participants rely on various supports that help them stay committed to teaching coding ........................................50

4.4.3. Personal interest and the requirement to teach coding in their professions are what helped educators start and stay committed to this work ........................................52

4.5. Conclusion ........................................................................................................................................53

Chapter 5: Implications ...............................................................................................................................54

5.0. Introduction.........................................................................................................................................54

5.1. Overview of Key Findings and their Significance .............................................................................54

5.2. Implications ........................................................................................................................................56

5.2.1. Educational community .................................................................................................................56

5.2.2. Personal implications .......................................................................................................................58

5.3. Recommendations .................................................................................................................................58

5.3.1. Ministries of education .....................................................................................................................59

5.3.2. Schools and administrators ...........................................................................................................59

5.3.3. Teachers ...........................................................................................................................................59

5.3.4. Faculties of education .......................................................................................................................60

5.4. Areas for Further Research ..................................................................................................................60

5.5. Conclusion ...........................................................................................................................................61

References ..................................................................................................................................................63

Appendix A: Letter of Consent for interviews .............................................................................................68

Appendix B: Interview Protocol Introductory Script ..................................................................................70
Chapter 1: Introduction

1.0. Introduction

The Canadian education system ranks among the best in the world, yet it faces a challenge in educating a generation that interacts with a rapidly changing socio-economic and highly technological world (Boudreault, Haga, Paylor, & Sabourin, 2013). We live in a different world that changes quickly and requires its citizens to adapt to the changes. STEM (Science, Technology, Engineering and Mathematics) education has been one of the emerging ways to help students be prepared in adapting to this changing world. A large part of STEM education is the education of technology, not how to use it but the thinking involved in making technology. The term we use to describe this study of technology and algorithmic processes, including the principles, the hardware and software designs, the applications, and their impact on society is called Computer Science (CS) (Tucker, Derek, Jones, McCowan, Stephenson, & Verno, 2003). It is to be noted that the term Computer Science and Coding are used interchangeably in this paper and refer to the same concept. Coding is a term commonly used in teaching and educators are more familiar with this term. In simple terms Computer Science is fundamentally about algorithms, which are recipes for solving problems and performing tasks. In this paper I explored the impact of introducing Computer Science as part of STEM education within the elementary school curriculum and why it is necessary.

CS is rigorous, multi-faceted and intellectually vibrant with an immense impact on modern day lives and professions. Its importance in our lives today and in the future highlights the reason it needs to be introduced earlier in education (Tucker et al., 2003). Early introduction of computer science concepts not only helps us prepare students for the future, but also helps children in mathematics and in their cognitive development. CS is commonly believed to be an
abstract and difficult subject, thus the notion of elementary students learning it may sound ridiculous to some. However, according to recent research, elementary school children are actually capable of learning basic coding concepts which is the foundation to them grasping much more complex concepts faster and at an earlier age (Martinez, Gomez, Benotti, 2015; Franklin, Hill, 2015; Tucker, Derek et al., 2003). At the elementary level this may look like giving clear instructions to find your way out of a maze or following a recipe. However simple the task may be, it is essential that students are exposed to the field of computer science as early as possible—as it is foundational in transforming the way a student thinks about the world. It not only teaches them about technology, it also teaches them how to think differently about any problem (Computing in the Core, 2014).

After overcoming the initial confusion of having CS education in elementary school, one may also be concerned with the difficulties in implementation. These difficulties include financial investments, additional workforce, time consumption and more (Google, 2014a). Fellows (1991) showed that the basics of CS are not just about machines and do not require computers or complex technology in order to learn. Furthermore, these concepts are easily learnable by new staff and can be combined with the current curriculum. (Fellows, 1991; Franklin, Hill, Iveland, Killian, & Harlow, 2015; Martinez et. al 2015). Admittedly this type of proposal poses a lot of questions, many of which have been answered by research and implemented by countries of similar demographics like UK by providing a separate technology curriculum. This paper explores the research and works on clarifying, exploring and emphasizing the need for CS education in our education system.
1.1. Articulation of the Research Problem

The introduction of coding education in elementary schools is important to solve three major issues that have become obvious from previous research. First, there is a significantly smaller amount of people in the field. Student stereotypes about the culture of these fields and common perception of it being difficult steers students away from choosing or even exploring a career option within the field (Bergey, Ketelhut, Liang, Natarajan, & Karakus, 2015). There has been an overall decrease in interest with careers related to CS. Students often believe CS has a higher degree of difficulty, are often unaware of its vast career aspects or have stereotypical perception of a career in CS (Marcela, Katherine, & Marisa, 2010). With an ever-growing technological industry, the next generation’s interest and development in CS is vital to our economy. Students need to grasp the power of critical thinking, problem solving and its connection to technology. They need to understand its true potential and the magic that lies beneath the screens they use every day. All of which they may stay ignorant to if the educators of today do not expose them to their future.

1.2. Purpose of the Study

In light of these problems the goal of my research was to learn what strategies teachers use to educate elementary students coding concepts in creative and immersive ways. I also wanted to consider what these educators do to help motivate learners of all backgrounds and identities and themselves to continue to do this work. This brought attention to possible problems the educators had to overcome in the classroom in order to successfully teach CS.
1.3. Research Questions

The main question guiding this research study is: how do educators teach complex computer science concepts to elementary students in creative ways and what strategies do they use to engage their students in this field?

Subsidiary questions that have been investigated include:

- What resources support these teachers in this work?
- How do these teachers develop their commitment to integrating CS in elementary education, and what experiences helped prepare them?
- What challenges do these teachers encounter when teaching CS concepts to elementary students and how do they respond to these challenges?
- How do these teachers think elementary teachers in Canada can incorporate certain elements of CS education in their learning to assist their students in other courses like math and science?

1.4. Background of the Researcher

As someone who comes from the world of technology, I have firsthand experiences in learning and adapting to this different world. When growing up technology was not too prominent in schools. We were not taught to think like programmers or to know what code is. I did my undergrad degree in Computer Science. Many of my colleagues had become interested in this field of study through video games and fascination for new devices. However, very few knew what computer science was, or were prepared for the tough ride in an undergraduate degree. Many dropped out of the program, especially females, because they thought it was too hard, and they feared failure. Since many courses are highly intensive in problem solving they required a cyclical process to find a solution. Students were encouraged to go back and find
issues in their code and try to fix them to make it work. There was never one common solution to the problem, nor was the path to get to the solution direct. This I think was the most frustrating skill to develop in the program.

Individuals who were capable of adapting to this idea that the learning is different from what we learned in school were able to overcome this challenge. In school there we were usually taught the strategies to get to one solution, often, it would take one or two tries to solve a math problem. In computer science, there is no one solution, there were many possibilities and each of them required lots of testing and fixing. This process was equally frustrating and rewarding for me and my colleagues - when some could not adapt to it, they left the program.

When I graduated, I found out that there are so many jobs available in STEM fields, especially computer science but not enough graduates to fill them. This is where I started to think that maybe we needed to advance our education system and teach these ideas from a younger age to get more people to be persistent and interested in this field. I decided to do research on what coding looks like at a younger age, and how it affects students. I feel that introducing these concepts earlier in age and becoming familiar with them will allow all students to gain more interest and confidence in the field, resulting in more people wanting to pursue careers in technology.

1.5. Preview of the Whole MTRP

To respond to the research questions, a qualitative research study was conducted using purposeful sampling to interview 2 teachers about their instructional strategies in teaching computer science concepts to elementary students and the outcomes they observed in terms of student interest and impact on overall learning. In Chapter 2 the literature was reviewed in the areas of computer science education in elementary schools and obstacles to implementation
within the educational pathway in the field. Next, in Chapter 3 the design of the research study was discussed. In Chapter 4 research findings and their significance were discussed in light of the existing research literature, and in Chapter 5 the implications of the research findings were identified for my own teacher identity and practice, and for the educational research community more broadly. Furthermore, a series of questions raised by the research findings were articulated to point to areas for future research.

**Chapter 2: Literature Review**

**2.0. Introduction**

In this chapter the key themes in literature revolving around computer science education are explored. First a few key terms used throughout literature and their meanings are analyzed. This helped in understanding the in depth discussions taking place among various researchers in the industry. What CS education is and what it looks like at the elementary level is examined – it is quite surprising to see how aptly CS education can fit into the curriculum. This analysis also shines light on why we need CS education in our schools today. The review of literature is continued with the students’ perspectives on CS education and a look at the gap of interest for CS education. Perspectives of the teachers and the obstacles they may face when trying to incorporate CS education are also studied. Lastly the current state of CS education in the world and the policies surrounding this issue within US and Canada are discussed. This chapter provides a greater understanding of the issues surrounding CS education and its importance in the lives of students today.
2.1. What is Computer Science Education?

2.1.1. Terminology

Students, parents, teachers and school administrators are often confused between computer science education activities and general computer literacy (Google, 2014a). Computer science (CS) is the study of computers and algorithmic processes, including their principles, their hardware and software designs, their applications, and their impact on society (Boudreault, Haga, Paylor & Sabourin, 2013). Computer Literacy is the basic non-technical knowledge about computers and how to use them (Boudreault et al., 2013). It is important for students to understand the breadth of computer science and the value of computer science skills so they can make informed decisions about whether to learn. It is equally important for school leaders to understand what constitutes computer science as they try to engage students in developing these foundational skills (Google, 2014a; Tucker et al., 2003).

Another source of confusion, for the stakeholders within and outside of the education system is the difference between computer science and computer engineering (CE). Scientists and engineers have a common interest in how objects and ideas fit together. However, the scientist seeks the answer to how things work as an end in itself. The engineer seeks answers to build the objects. Consequently, CS consists of the theory of computation while drawing from mathematics, while CE consists of the design of physical objects drawing from physics and chemistry (The State University of New York, 2010).

2.1.2. What does it look like at an elementary level?

When we speak of CS we speak of it as algorithms, theory and other complex terms that are difficult for adults to understand, let alone children. So then what does CS education look like in younger grades? In the primary grades CS education is extremely simplified to the basics
of problem solving. It may look like giving complete step-by-step instructions or what is known as an algorithmic definition for: finding your way out of a maze, a dog retrieving a thrown ball, baking cookies, going home from school, making a sand castle, arranging a list of words in alphabetical order and more (Tucker et al., 2003). These are all simple tasks that each student may do without even knowing they do. The idea is to help them become aware of it and perfect these instructions so that a computer can understand them.

Another example of a simple application in primary is the idea of sharing and sorting. Students can learn to sort themselves by birthday and month without speaking and then reflect on what they did. They can do this in several different ways to see which way was the fastest. This activity is a simple introduction to sorting algorithms in CS, which are part of the first few topics new students in CS learn (Code.org, 2015). These lessons can easily be combined with any subject, specifically mathematics and can be used to gain student interest as well. For junior level students, activities can be brought to an actual computer where they begin to learn, develop and debug simple software themselves (Code.org, 2015). CS education at an elementary level is not a difficult process, it simply requires adopting a lens of coding when teaching.

2.2. Why Do We Need CS Education?

Now that the notion of Computer Science is clarified, another important question needs to be answered – why is computer science education important at the elementary level? Current research from Google (2014a) clearly emphasizes on the rise of careers in STEM – of which technology is the most in demand. However, with this rise the number of students graduating in these careers is declining (Google, 2014a). Furthermore, not only technology but also all of the other STEM strands somehow involve usage computer science knowledge in some form. Apart from a lack of people in these careers, another major problem is prominent – the gender gap in
STEM careers is no secret and research suggests that introducing computer science at an earlier age helps motivate more females to enter these professions (Bergey, Ketelhut, et al., 2015; Lightbody, Siann, Tait, & Walsh, 1997). Computer science education has a multitude of benefits like future career and cross-curricular opportunities that enhance an elementary student’s learning in various ways. Next, I review some of these.

2.2.1. Preparing for the future

More than 50 percent of STEM jobs are predicted to be in CS related fields by 2020, this is a significantly large number. In total there will be about 1.4 million jobs in this field and only 400,000 graduates to fill them (Office of the Press Secretary, 2014). We are already in the year 2016 and the Ontario education curriculum is nowhere close to emphasizing this importance – we have a shortage of computer scientists (Beyer, Rynes, Perrault, Hay, & Haller, 2003). As Lemaire (2014) states, the next generation needs to able to think critically and resolve problems related to technology. They need to be capable of keeping up with the quickly changing world of software that we are being dependent on. A research project conducted by Google reports that early exposure to CS can help students develop strong interests and be more competent in the field (Google, 2014b). Given the current situation in the profession and the future, it is important for educators and educational faculties to start the process of mandating the CS curriculum in order to prepare the next generations.

2.2.2. Application to other subjects

One of the most unique things about CS is that it can be useful in any profession one chooses, as it is multifaceted. Students of all learning capabilities and strengths can make use of computer science in one way or another (Google, 2014a). The visual learner may like web design, the logical learner can work on CS theory, the kinesthetic can focus on Artificial
intelligence and robotics, and the musically intelligent can work on video game. Whatever their interest, foremost Computer Science teaches students to think about problem solving itself. Students learn to define the problem clearly and then working with others to reach the next solution. As Tucker et. al (2003) state, “There is an unmistakable link between success, innovation, and computer science. Movies like The Incredibles and Lord of the Rings required the development of new computing techniques. Progress on understanding the genetics of disease or of creating an AIDS vaccine requires professionals to think in terms of computer science—because the problems are unsolvable without it” (p. 3). Evidently computer science can apply to virtually every profession directly or indirectly, its vast diversity and malleability make it an ideal subject to be included in the curriculum and in turn help our students get the skills they need.

2.3. Students’ Perspectives on CS Education

Although the research question for this study focuses on the exposure to computer science for all young students, the industry comes with a lot of stereotypes and gender biases. To get a complete understanding of education in CS, it is important to explore the vast amount of research on perspectives of older students from both genders. Their experiences help to inform some issues that younger students may encounter in the future and look at possible solutions to them. This section looks at this research.

2.3.1. The computer science stereotypes

The term Computer Science has been often attached to many stereotypical notions. From the lonely geek in school to the nerds’ club with 4 participants, it doesn’t have a popular image. When a group of undergraduates were interviewed about their beliefs of computer science it all
came down to one image which is perfectly summarized by a student in Fisher, Margolis, & Miller’s (1997) study,

“I ask them, ‘How can you sit in front of a computer for eight straight hours and then when you go home you start to play on computer games again?’ And then they say, ‘oh, because it's fun.’ I say, ‘don't you spend time with your friends?’, and they say, ‘no, I just like sitting in my room and just play these games.’ So I just felt really different because, I don't know, I don't know... if you want to major in computer science, what you are supposed to do? Like just play on the computer all day? I don't, so I felt different” (p. 106).

In the midst of this common belief there exists an even deeper problem – the gender gap. Women make up only 26% of the Computer Science and mathematical profession in the United States (Butler, 2000). Thus in the already low rate of Computer Science professionals, women are an even rarer find in this brilliant field. Technology is handled and perceived differently by both genders, and this is an important factor in understanding why this gender gap exists (Butler, 2000; Siann, Macleod, Glissov, & Durndell, 1990)

2.3.2. Female perspectives

Women who are talented in mathematics or computer science often choose to go into professions in medicine or law because they believe the stereotype of CS being a male dominant field and a less socially interactive one (Lightbody, Siann, Tait, & Walsh, 1997; Bergey, Ketelhut, et al., 2015). More specifically the The American Association of University Women (AAUW) report identifies the following reasons why fewer women are majoring in computer science (CS) (Werner, Hanks, & McDowell, 2004):

- A career in computing is not well rounded and does not accommodate family life
• The computing profession is oriented towards a more competitive environment rather than a collaborative one
• The perception of CS as a more individualistic occupation for introverts, not a social occupation

An important note once again is that these notions are all stereotypical biases. Anyone who currently works in this profession knows that none of these reasons are true. A study done by Google (2014b) on women who choose computer science suggests that encouragement and exposure are the keys to solving these problems. A confidence in self and an interest in problem solving combined with a belief that these skills can translate into a career is the first step for all females to succeed in CS. Positive reinforcement from family and peers also helps make a drastic difference in interest levels. After interest levels have been driven up, the quality of academic exposure help sustain and develop the interest into a blooming career. The availability of resources and opportunities for participation, course work and engaging extra-curricular activities all help enhance the female students’ interests and learning (Google, 2014b). Women function differently from men, and the way they perceive interest and careers is different too. A study has found that women seem to use computers in order to accomplish a goal or a task, while men have a different attitude discussed later in the review (Fisher, Margolis, & Miller, 1997). When women talk about computer science they often contextualize it in a more goal-oriented manner. For example, asking the question “What can I do in the world with this skill” (Butler, 2000). They look at technology as a means to an end, and pay less focus to the actual process of using it. Additionally, when asked to rate themselves on attitudes about computer ability girls consistently rate themselves lower than boys (Beyer, Rynes et al., 2003; Butler, 2000; Boser, Daugherty, 1996). This shows that women lack confidence in themselves in the
initial stages of their interest, which can only decline given the environment they are placed in. Often perceptions of males, their educators, and the education culture are subconsciously male-centric driving females further away from a profession in Computer Science.

Perceptions of computer science as a career, self-confidence, role models within the profession, pedagogy and culture within Computer Science education are all factors which influence the female student – mostly in driving them away from a web of infinite career opportunities.

2.3.3. Male perspectives

On the contrary to female perspectives, male perspectives are a lot more positive and passionate. Although the common stereotype still affects their participation in computer science professions, men who have a dedicated interest in CS are more likely to pursue it further than females. This interest stems at an early age (before 10yrs) when several males describe epiphany moments of their computing experiences. They seem to become consumed earlier on and their practices become a consistent part of who they are. In an interview conducted with one of the participants in Fisher (1997) et al.’s study, one student describes what got them interested,

“Well, I think it was sometime in middle school, sixth grade about then, my dad borrowed a computer from a friend, it was an old black and white Macintosh, just totally self-contained one unit thing, and I remember just playing with that all the time and trying to figure stuff on it. And that got me really hooked ... I was really getting into figuring things out on computers and I just knew that that was going to be something for me” (p. 107).

Boys often describe their experiences while using the computer as playing or having a good time. They give detailed accounts of their experiences as well not giving as much
importance to having a goal like their female colleagues. In connection with their female colleagues, Boser and Daughtery (1996) studied that males often have stereotypical views of female abilities when it comes to Computer Science. This could be due to the culture all students are exposed to and difference in attitudes and learning styles of both genders.

An interesting fact is that both females and males start to use computers at an early age and are both interested in them, their attitudes somehow diverge as they grow. A study conducted by Christensen et. al (2005) for two years (2001-2002) in a middle school in Texas showed that from Grade 1 through Grade 3, there appeared to be no consistent differences between males and females on attitudes toward computers. At Grades 4 and 5, girls were significantly higher than boys in computer enjoyment for both years studied (2000 and 2001). From Grade 6 through Grade 12, males appeared to be consistently more positive in their attitudes toward computers than females, and especially higher in computer enjoyment. The divergence by gender in computer enjoyment for 2001 was best described as a decline on the part of girls rather than an increase on the part of boys. Although not many studies have been conducted to show clearly where the divergence stems in the early ages, many researchers agree that periods of transition between elementary to middle school and middle school to high school may be fruitful to look at (Christensen, Knezek, & Overall, 2005; Wilder, Mackie, & Cooper, 1985).

2.4. Teachers’ Perspectives on CS Education

2.4.1. Educators teaching CS

Bringing Computer Science into the elementary school curriculum successfully involves one significant factor – teachers who are knowledgeable of the content and capable of teaching it in an appropriate, engaging manner. However, this is an extremely difficult task for any country
or province to accomplish. Data from a Google (2014a) study conducted in the United States shows that 42% of principals and 73% of superintendents say that there are no teachers available at their schools/in their districts with the necessary skills to teach computer science. This can be applied to situations in Canada as well, since we know people interested in Computer Science are few and most teachers come from a non-CS background. Within the slim pool of qualified teachers, academics Reinen and Plomp (1993) have noticed that fewer women than men are computer educators – this explains fact the necessity of female students needing role models within the profession to look up to.

Another issue with educators is that they do not have time to focus on Computer Science as it is not a mandate in some places and other things take priority. Nearly a third of principals and 24% of superintendents say their school or district has to devote most of its time to courses related to testing requirements, and these courses do not include computer science (Google, 2014a). With so many courses and requirements to cover, teachers are overloaded with work; this is simply another addition to their work. According to the same study teachers, administrators and principals do see the importance of CS education however they do not perceive a demand from their students or communities. Given these obstacles it may take time to fully incorporate CS education however it is undeniable by all stakeholders that it is needed.

2.4.2. Resources available

Despite having less CS educators, there are many educators around the world who have taken up the challenge of bringing CS into their classrooms. Two key issues these teachers faced were that their students did not have enough time at computers, and that teachers needed extra planning time for technology lessons. Other concerns were out-dated hardware, lack of appropriate software, technical difficulties, and student skill levels (Bauer & Kenton, 2005).
Many of these problems require technology upgrades and funding from the government. However as in a previous section about what CS teaching looks like, CS education does not always require technology - teachers can educate students to think critically about problems and solutions using simple pen and paper.

While issues surrounding funding do exist, certain techniques can be used to accommodate for them. Pair Programming is a common technique used by CS professionals and students. Two people work together on one device to solve the problem at hand. Pair programming has proved to be beneficial in many ways other than requiring less technology; it improves student confidence, comprehension and enhances learning from peers (Werner, Hanks, & McDowell, 2004). Furthermore, free software and materials are available for new CS teachers to use such as Code.org, Code Academy, and barefoot computing. These websites focus on CS education in both early grades and high school and are affiliated with the government as well as leading organizations like Google, Facebook and Microsoft. Code.org is one excellent example where teachers can access complete lesson plans along with the software and videos needed. It also provides teacher preparation workshops, student level tracking, around the world statistics and support (Office of the Press Secretary, 2014). With these materials any teacher with or without a background in CS can learn and teach it to their students.

2.5. CS Education in Policy and Action

2.5.1. CS education around the world

Computer Science education had been getting a lot of attention all around the world and many countries have implemented or begun to incorporate it in their primary curriculums. Of these countries UK, Australia, and United States are the leaders. In Australia the curriculum starts in Grade 2, the UK curriculum has mandated CS education and started teacher training. A
collaborative website with Barefoot Computing provides lesson plans and content to help teachers satisfy the curriculum requirements, many of these lesson plans are adaptable by teachers anyway around the world (Balanskat & Engelhardt, 2014; Australian Curriculum, Assessment and Reporting Authority, 2015). Many states in the US have started adding CS to the curriculum with the help of code.org.

A big initiative to raise awareness for CS education around the world that was started by code.org is called an “Hour of Code”. Anyone, teacher or not around the world can conduct an hour of code, it is simply a mock lesson that introduces CS concepts. Kids older than 4 up to any age can participate in the hour of code and learn a little bit about CS. This year 198,473 hours of code events took place all around the world – 4,655 were held in Canada (Code.org, 2015). Moreover, the hour of code has happened in 40 different languages. This is an excellent example of how the world is responding to the importance of understanding Computer Science. As educators we need to begin incorporating this in the classroom.

2.5.2. US policy

The United States has a good start on researching and implementing CS education in various schools. Formal Commitments by the government of USA published clearly emphasize this. Commitments have been made by more than 60 school districts, including the seven largest school districts in the country, to offer computer science courses to their students. That is a total of 4 million students in more than 1,000 high schools and middle schools, in partnership with Code.org. There is over $20 million in philanthropic contributions to train 10,000 teachers by fall 2015 and 25,000 teachers to teach computer science to in time for the school year beginning in fall 2016. Additionally, new steps to increase the participation of women and under-represented
minorities in computer science, including a new computer-science classroom design prize and innovative outreach efforts are being put in place to get help overcome the gender gap.

2.5.3. **Canada and provincial policy**

Within Ministry of Education policies the importance of technology-enabled pedagogical practices that focus on developing higher order skills is clearly emphasized with the 21st Century Teaching and Learning Initiative. Upcoming teachers are being provided with pedagogical and instructional strategies knowledge about technology as a teaching tool – as part of their faculty of education program’s mandatory core content. This is a requirement in the recently amended Ontario Regulation 347/02 Accreditation of Teacher Education Programs, under the Ontario College of Teachers’ Act (Ministry of Education, 2013). However, more of this push is with regards to Computer Literacy. The idea of Computer Science education in elementary schools is still a vision and needs to be acted upon (Ontario Public School Boards Association, 2013).

Currently in the province of Ontario, CS education starts at Grade 10. The curriculum states that students will gain an understanding of computer studies concepts; develop the skills, including critical thinking skills, and the knowledge of strategies required to do research, conduct inquiries, and communicate findings accurately, ethically, and effectively; develop lifelong learning habits that will help them adapt to computer advances in the changing workplace and world; make connections that will help them take advantage of potential postsecondary educational and work opportunities (Ministry of Education, 2008).

2.6. **Conclusion**

In this literature review I examine what computer science is, why we need it in elementary schools, students and teacher perspectives on CS as well as the policies and actions currently associated with CS around the world. This review emphasizes the importance of presenting CS
in its truthful nature as stereotypical beliefs play a major role in the way individuals perceive it. It provides concrete evidence to show that CS education is important within our schools at an earlier age to prepare the next generation and their critical thinking skills. It also explores the obstacles that students face within the profession, specifically females and possible solutions to the issue. Whether or not these solutions work in practice remains to be found.

There are many barriers that prevent CS from easily being a part of the curriculum. These barriers include the lack of knowledgeable CS teachers, technological resources needed and most importantly finding time in the busy school life to incorporate these lessons. Many countries like UK, Australia and US have begun to overcome these obstacles and organizations are raising CS awareness around the world. Our neighboring countries show that it is possible to successfully teach CS to elementary students despite these obstacles, now it is up to Canadian teachers to take these lessons and do the same.

Overall, with my research I hope to understand how teachers adapt their classrooms to integrate CS education. I believe having real examples of the difficulties they faced and how they overcame them will contribute to helping and inspiring other teachers to do the same. The research highlights issues with perceptions of CS, resources available and lack of policy in Canada when it comes to CS education, I would like to explore ways that teachers overcome these issues to help inform the educational society that it is both possible and beneficial to have a CS integrated curriculum.

Chapter 3: Research Methodology

3.0. Introduction

In this chapter I describe the research methodology. I begin by reviewing the general approach, procedures, and data collection instruments, before elaborating more specifically on
participant sampling and recruitment. I explain data analysis procedures and review the ethical considerations pertinent to my study. Relatedly, I identify a range of methodological limitations, but I also speak to the strengths of the methodology. Finally, I conclude the chapter with a brief summary of key methodological decisions and my rationale for these decisions given the research purpose and questions.

3.1. Research Approach and Procedures

This research study was conducted using a qualitative research approach involving a literature review for the topic at hand and semi-structured interviews with two teachers. As Ritchie, Lewis, Nicholls, and Ormston (2013) mention, it is important to recognize that there is no single way of doing qualitative research. It is highly dependent on a range of factors including but not limited to the researcher’s ontology, epistemology, purpose and goals of the research, characteristics of participants and more. Qualitative research claims to describe observations ‘from the inside out’, from the point of view of the people who participate. By so doing it seeks to contribute to a better understanding of social realities and to draw attention to processes, meaning patterns and structural features. In recent years qualitative research has developed into a broad and sometimes confusing field of study. It has become part of the training in empirical research methods in a variety of subjects and disciplines (Flick, 2009).

On the other hand, quantitative research is heavily based on data that is often numeric or a rigid structures and calculative process which produce patterns and set results. This approach is known to follow the scientific methodology with an emphasis on hypotheses, methodology, and generalizations. However, this approach lacks in understanding, finding meaning or reasoning behind a particular concept or ideology (Ritchie et al., 2013). Often with quantitative research questions related to “what”, “where” and “when” are focused on versus the “why” and “how”
deeply explored in qualitative research. Given the purpose of my research—which is to find out how educators teach computer science; I find that a qualitative approach is much suitable. Since I would like to explore the educator’s lived experiences, their observations and strategies to teach these concepts a qualitative study allows me to get in-depth information on these educators and make meaningful observations. A qualitative study will allow me to develop a detailed understanding of the journey these educators go through both within the computer science field and as emerging educators. Doing so will help me shine light on some strategies new educators seeking to pursue the path of CS education can implement in their own practice. All of this insight can only be provided through a qualitative methodology.

3.2. Instruments of Data Collection

Within the qualitative approach, I conducted semi-structured interviews with two educators who have experience teaching computer science to students. Semi-structured in-depth interviews are the most widely used interviewing format for qualitative research and can occur either with an individual or in groups. They are often the sole data source for a qualitative research project and are usually scheduled in advance at a designated time with the participants. They are generally organized around a set of predetermined open-ended questions, with other questions emerging from the dialogue between interviewer and interviewee(s) (DiCicco-Bloom, Crabtree, 2006). Most commonly they are only conducted once for an individual and take between 30 minutes to several hours to complete.

For the purpose of this study the interview took between 45-60 mins. I had a set of predetermined questions (Appendix A) and an interview protocol to be followed, but the interview was guided by the protocol rather than dictated by it. Most importantly there was an attempt to establish a good rapport with the respondent as is important and common within semi-
structured interviews. The goal was to have the participant share their insights in a comfortable and honest manner. Consequently, to create this environment, the ordering of questions was less important. The interviewer was free to probe interesting areas that arise and the interview could easily follow the respondent’s interests or concerns. This type of structure in individual in-depth interviews allowed the interviewer to delve deeply into social and personal matters learning about personal experiences and exploring novel areas that produce richer data (Smith, 2007).

My particular interview protocol had questions that focused on four main categories: the background of the participant in education and computer science, their beliefs on the importance of CS education being a part of the elementary curriculum, their teaching experiences with K-12 students and CS concepts, and lastly inquiring about the supports and challenges as an educator in the field they are in. The interview concluded by discussing what the participant sees as the next steps in CS education and what actions they believe will prove to be most effective to move this forward.

3.3. Participants

Here the sampling criteria I established for participant recruitment is reviewed, and a range of possible avenues for teacher recruitment are discussed. I have also included a section wherein each participant is introduced and how they fit the sampling criteria is explained.

3.3.1. Sampling criteria

Below are the criteria that was applied to all participants:

- Participant does not have to be OCT certified, they just need teaching experience in CS
- Educators with some (6+ months) teaching experience in programming, coding or some sort of computer science related course
- Educators with some (6+ months) experience working in the computer science industry
• Educators with a committed dedication to promoting and teaching computer science concepts to children ages 5-12yrs

Since this study focused on the lived experiences of educators in computer science, it was important to select participants that were able to provide answers to the questions involved.

Computer Science is not a mandated subject in the curriculum, thus I sought participants that are not necessarily certified teachers but work as educators leading camps and programs that teach computer science. These had experience working in the technology industry so they can speak to their experiences within the workforce. Since the study looked to identify key strategies that can be used in educating young children about CS concepts, it is important for these individuals to have a dedication to the younger age group and can speak to their observations within the classroom. Fulfilling the listed criteria allowed for an in-depth interview giving the research lots of insight and data to interpret.

3.3.2. Sampling procedures

To recruit participants, I attended professional development conferences hosted by technology start-ups such as Code.org, Kids Learning code and so on. I also participated in Kids Learning Code camps over the summer where professionals within the industry come and teach kids about computer science. I contacted professional associations, teacher education programs, and subject-area specialization organizations within the Code.org community to seek educators outside of Canada that fulfill the success criteria. I provided the participant criteria and asked that these individuals/organizations distribute my information to teachers they believe may fulfill the criteria. My sampling procedure mainly relied on convenience sampling (Teddlie & Yu, 2007) as I was immersed in a community of professionals in the industry and mentor teachers, and also relied on snowball sampling from on my existing contacts and networks to recruit participants.
Considering many ethical and security issues that may arise when participants are asked to share their personal information, I provided my contact information. It was much more comforting for the participants rather than ask these individuals/organizations to provide me with the names and contact information of people they think would be suitable. This helped ensure that teachers were volunteering to participate rather than feeling pressure or obligation to participate (Teddlie & Yu, 2007).

3.4. Participant Bios

Two participants, Diane and Stacy were interviewed for this study. Stacy is a professor in education technology for individuals learning to become teachers in a faculty of education. She has taught both junior and primary grades for 4 years and then she moved on to being a technology lead at her school. As a technology lead she also ran a student council club which became a technology club. She taught interested students in the club the basics of web design for the purpose of maintaining the school website. She currently teaches teacher candidates the basics of coding, website designs and app making as well as strategies that help them incorporate this material in the classroom. Stacy has a master’s degree in education technology and a PhD in women’s studies. Stacy is a self-taught person and continuously pushed herself to learn more about the technology world. She is very passionate about both learning and teaching coding.

Diane is a Grade 6 teacher at a private school. She has taught this grade for the past 3 years. She teaches an all-girls classroom and tries to incorporate various coding activities in her planning. Although Diane has no formal education in technology she was compelled to learn it because of the demands of her job. Diane makes it a point to attend different workshops and fairs to learn more about how she can incorporate and work with coding in her class. It is to be noted that both Stacy and Diane are related (they are sisters) and come from a family of teachers. They
both accredit where they are as educators to that background however their experiences with coding are very different. Both Stacy and Diane have young children with whom they try out different coding materials with. They both understand the importance of technology in their own and their kids’ lives and are dedicated to bringing their philosophy into their classes.

3.5. Data Analysis

In qualitative research, data analysis first consists of preparing and organizing data for analysis (Creswell, 2013). So foremost I transcribed my interviews to prepare them for analysis. From there I coded my transcripts using my research questions as an interpretive tool. I coded each transcript individually and identified categories of data (e.g. instructional resources) and themes within categories (e.g. online resources are often used to start students off with CS education). After this process of categorizing, I read the categories and themes beside one another – synthesizing themes where appropriate. A later stage of analysis was the meaning-making process whereby I spoke to what matters about these themes and findings given what existing research has already found (i.e. the literature review). Throughout the process I also looked for “null data” – that is, what participating teachers did not speak to, and why it matters. This type of qualitative analysis provided a good depiction of the connections and relationships between different concepts and themes (DiCicco-Bloom & Crabtree, 2006). It helped in further contextualizing the responses and recognizing patterns which emerge in helping to answer the research question.

3.6. Ethical Review Procedures

There are four ethical issues related to a semi-structured interview process (DiCicco-Bloom et. al, 2006). First was reducing the risk of unanticipated harm. There are no known risks to participation in this study. Since a lot of care was taken in reducing the risks associated with
the research, unanticipated harm is highly unlikely to occur. The second issue was protecting the interviewee's information. As a solution to this issue, all participants were assigned a pseudonym and they were be notified of their right to withdraw from participation in the study at any stage of the research study. Participant identities were kept confidential and any identifying markers related to their schools or students were excluded. Participants had the opportunity to review the transcripts and to clarify or retract any statements before data analysis was conducted. All data (audio recordings) will be stored on my password-protected computer/laptop/phone and will be destroyed after 5 years. The third issue was effectively informing interviewees about the nature of the study. Participants were asked to sign a consent letter (Appendix A) giving their consent to be interviewed as well as audio-recorded. This consent letter provided an overview of the study, addresses ethical implications, and specified expectations of participation (One 45-60-minute semi-structured interview) so that the participant was well informed of the nature of the study. Last was the issue revolving around risk of exploitation of the participants and their information. Each participant was given an anonymous biography in the paper and were credited for their ideas anonymously. They received a copy of the research study at the end of the study. The study is being conducted purely for purposes of research and informing educators of these new findings, there is no personal gain involved.

3.7. Methodological Limitations and Strengths

Although a qualitative study using in-depth semi structured interviews was an effective methodology for this research, it had some limitations. One of these limitations was that the data found and conclusions made cannot be generalized to the larger population (Creswell, 2013). These interviews also went in-depth with a few only people’s experiences. Thus, data that is
analyzed is highly biased and opinionated on the perspective of the participant and cannot be applied to all participants that fit the criteria. The interviews tend to form close relationships between the participant and the interviewer (Creswell, 2013). The weakness of such a close relationship is the likelihood that it may become pseudo-therapeutic. The bond can also potentially form unknown biases in interpreting the research as a whole.

Furthermore, given the ethical parameters that we had approval for, the MTRP can only involve interviews with teachers, and consequently it is not possible to interview students or parents, or to conduct surveys or classroom observations. This is a drawback as it would be beneficial to see how parents and students think CS education affects them and their enthusiasm about the topic. It is important to consider other perspectives and pick different participants to interview to gain further insight into the research problem (and include citation support). Since the topic is still being introduced in the Canadian education system, there are a very limited number of teachers that fit the criteria of the participants that is another obstacle in the process. Furthermore, as there was a limitation on the number of teachers that can be interviewed, thus it is important to acknowledge that while the findings can inform the topic at hand, they cannot generalize the experience of teachers more broadly speaking.

In terms of the methodological strengths, interviewing teachers using this methodology allowed me to hear from them in more depth than a survey could allow for, and it also created space for teachers to speak to what matters most to them when it comes to the topic at hand (Creswell, 2013). In this way, interviews validated teacher voice and experience, and are an opportunity for them to make meaning from their lived experiences. The interviews also acted as an opportunity for teachers to reflect on their practices and to articulate how they conceptualize particular topics in theory and in practice.
3.8. Conclusion

This chapter explained and discussed the research methodology being used in the research project. First the research approach was identified to be a qualitative approach, followed by the benefits and significance of using this methodology. Semi-structured interviews were identified to be the type of interviews being conducted for the study. The benefits behind this interview strategy and why it is suitable for the research at hand was discussed as part of the qualitative study. I then identified the participants of the study, specifying the criteria applied to all interviewees – participants must have some experience in both teaching and coding, etc. Then I provided reasoning for the criteria selected relative to the research problem. Examples of organizations that can be potential resource banks to recruit participants that fit the criteria were also presented. The idea of convenience sampling was also discussed as I am connected to a pool of potential educators within the industry. I proceeded to describe the process of how the data will be analyzed by coding individual interviews and looking for common patterns and themes across the data. Furthermore, four major ethical issues in qualitative research methodology were discussed and addressed. Lastly, I discussed the methodological limitations and strengths of the study. In the next chapter, I report on the findings of the research using the discussed methodology.

Chapter 4: Research Findings

4.0. Introduction to the Chapter

This chapter presents the findings that surfaced from the research interviews done with Stacy and Diane, two educators who incorporate coding in their classroom. I explored four main themes that emerged from a detailed data analysis of the transcribed interviews. These themes stem from the focus of being able to find observations for my research question - how do
educators teach complex computer science concepts to elementary students in creative ways and what strategies do they use to effectively incorporate coding concepts? In this discussion, connections are drawn between participants’ experiences, perceptions and the literature review. Findings are organized into four main themes:

1. Coding in the classroom has a range of positive outcomes that help student learning and preparation for the future according to educators.
2. Participants mention that teaching coding is dependent on the availability or knowledge of various resources to help support the teaching process.
3. Participants acknowledge various challenges when attempting to integrate coding into the classroom.
4. Participants maintain an overall commitment to teaching coding using various factors that surround their work as a teacher.

These themes also have sub-themes that further illustrate how they play out in a classroom. For each theme, I will first describe it, then report on the data, and finally discuss the significance of each theme within the context of the existing literature. In the end I provide a conclusion that summarizes my discussion of the themes.

4.1. Coding in the Classroom has a Range of Positive Outcomes that Help Student Learning and Preparation for the Future According to Educators

When students are exposed to coding tasks, participants notice an immediate change in the environment of the room. Students feel motivated and engaged through the tasks. According to the interviewees, coding can be added to many subjects and can be done at any age which makes it easy to do. More importantly, being able to understand these concepts prepares students for the
future they will live in. All of these advantages of why coding is important discovered in conversation with the participants are discussed in this theme.

4.1.1. Motivation and engagement in students is an almost immediate result of having coding in the classroom

Participants find that coding projects and activities are very engaging. Stacy said, “It is a powerful experience for them. It is the product you create using the technology but also the process is so engaging and I think that is why I keep coming back to it …it’s fun.” Diane also agreed with Stacy, she mentioned that this motivation also helps them stay committed to the project and keep trying to make something work. “They were highly motivated, interested, and engaged. So they were really obsessed with trying to make it work, and solve the problem they encountered.” Diane reported seeing students want to see the game they are creating, or the robot they are programming to work and accomplish the task they need it to do. The idea of seeing something that they developed from the beginning is the main motivating factor in these projects. Diane explains that students start being so relentless in making their programs work that they stop seeing mistakes but look at their program as an opportunity to improve on what they already have leading them to develop a growth mindset about their own learning which is so important in education.

This interest and engagement is also noted in the literature, however it is only observed when looking for the difference between boys and girls in perceiving coding activities. Boser and Daughtery (1996) noted that boys often describe their experiences while using the computer as playing or having a good time. They give detailed accounts of their experiences as well not giving as much importance to having a goal. Although this research only speaks to boys, this level of interest is also seen in all students in both Diane and Stacy’s experience. Thus there is
now some evidence, according to the participants, that all students are engaged when participating in coding activities

4.1.2. **Participants acknowledge that coding can be integrated into any subject topic at any age and produce results**

The fact that coding is an ability that has no age barrier was strongly emphasized by both Stacy and Diane. “I think they can learn some really basic concepts at a really, really young age, so I’m thinking kindergarten or even younger” said Diane. Specific concepts of coding may have age limitations—for example, the idea that a block of code does something is understood by a primary learner—however, the ability to recognize that the same block of code is being repeated several times comes later on in their development. Another example that is given by Diane is about a game that used some coding concepts,

> For example, in the light bot game, when we got to the stage where she had to understand the concept that one block can have many messages in it… that was a hard concept for her to get her head around, however I do think she’s capable of understanding that concept… I don’t think she knows there’s code behind it, but the concept of a block and you can place it somewhere and it will complete the function that it’s supposed to do. She can understand that and she is 6, just finished Grade 1.

Stacy adds an interesting comment that provides greater insight into how coding and age are related. She prefers to have an open mind about who can do what – a lesson taught to her by her son; “I was amazed at how much he could do …and he was often solving problems more effectively than I would solve them.” She continued to say that we often have assumption of what kids can do and are capable of. According to her, it is important to keep those assumptions
away when it comes to teaching coding. She states that it is enough to teach students the basic building blocks of coding and then let them be in charge of their own learning.

When asked about what she considered to be building blocks she mentions, “[In] kindergarten we do some stuff like look at instructions for example, that’s something we do as part of our everyday lives classroom agendas that are posted on the board, looking at recipes and cooking.” Stacy notes that many of these building blocks are already taught in classrooms, but teachers simply aren’t aware that this is also what coding is about. Teachers can easily add coding activities into their current curriculum to add more depth and motivation to an existing topic.

On a similar note, Diane also speaks to her coding projects to be multi-faceted,

There’s so much problem solving, I mean that’s what coding is about. Which is a huge major building block in math right, it’s all about problem solving. So, just the way you think, the process of your thinking is so mathematical, it’s so logical, and there’s so many other side mathematical skills you have to understand.

Diane follows up this statement with an excellent example of how one task works on so many attributes of a subject with the students. For one of her activities she had her Grade 6 students make a program in Scratch which made the cat travel to a random spot on the grid. Students had to first come up with a random spot, and then have the cat travel to it. Finding a random spot involved students using their knowledge of that x-y plane, coordinate mapping, and number sense. To get the cat to that spot involved geometry and spatial sense. It was coding project that had students applying all of these mathematical skills while having so much fun.

It is not complicated to teach coding or integrate it in the classroom, this is also confirmed in the literature too. In the primary grades, CS education is extremely simplified to the
basics of problem solving. It may look like giving complete step-by-step instructions, or what is known as an algorithmic definition, for finding your way out of a maze, a dog retrieving a thrown ball, baking cookies, going home from school, making a sand castle, arranging a list of words in alphabetical order and more (Tucker, Derek, et al., 2003). Coding is very versatile and adaptable to the age and topic as is also illustrated in the quote.

### 4.1.3. Integrating coding in the classroom prepares students for 21st century learning

Both participants strongly stated that coding prepared students for the 21st century and that it was important for them to introduce coding in their practice. Diane realised that early exposure to coding builds confidence in her students. She feels that if her students are not exposed to skills such as problem solving, debugging, creating code, working with technology, then they may feel like they do not have the opportunity to explore so many career paths that require the skill set that coding helps develop. Additionally, she believes that it is important for them to be able to be exposed to the languages of coding as they are the underlying blocks of all technology they interact so much with; “the languages behind [coding] are the building blocks of so much we use in everyday life and what they will use in the future, so it’s important that they know how they work.”

Stacy gave a very powerful and clear statement on why she thinks coding is an asset in the classroom:

For two things: First of all, equipping them for jobs in the future, high paying jobs especially like my little girl, I would love her to have the same opportunities that other people have and she loves computers and she is only 3 and I know every 3-year-old loves computers. But also it is about creativity, not just also but mostly its about creativity, it is
about out-of-the-box thinking. It is not about being programmed but actually programming something yourself.

Stacy extends the idea of 21st century learning to being creative, making something and carrying it forward from its birth to seeing it work. Students find coding activities to be very rewarding when they accomplish what they set out to do, and the path to that accomplishment teaches them a lot about how to tackle problems in general.

This observation connects directly to what is written in literature. Lemaire (2014) states, the next generation needs to able to think critically and resolve problems related to technology. They need to be capable of keeping up with the quickly changing world of software that we are being dependent on. Moreover, a research project conducted by Google reports that early exposure to CS can help students develop strong interests and be more competent in the field (Google, 2014). Both Stacy and Diane also confirmed this in their beliefs of why they taught coding. Thus, in both literature and participant voices we see that coding is an important part of the students learning and future.

4.2. Participants Mentioned that Teaching Coding is Dependent on the Availability or Knowledge of Various Resources to Help Support the Teaching Process.

In both literature and participant voices, we see that coding is a new topic to many people and people attempting to teach an unfamiliar topic need lots of resources. Many of these resources can be found online through websites for teacher who want to teach coding. Other resources are more physical and hardware-oriented, like toys for kids that help them see their code in action. Lastly there are many organizations that participants use as resources who have experts in the field of computer science willing to help out. These are three different ways participants gained support for their teaching.
4.2.1. Teachers frequently use online resources to support their work in integrating computer science education.

In the last few years, there has been a push for coding education in many countries after many countries have witnessed a lack of people willing to work in this field (Office of the Press Secretary, 2014). In an attempt to get kids interested in coding, many technology companies have innovated software and toys. Diane and Stacy mention a list online resources that they use, some to self-teach and some to use in the classroom. Scratch is a popular coding software for kids that both participants use in their teaching. Scratch is accessed through a website that allows kids to make their own animations and games. It helps young people learn essential skills such as creative thinking and problem solving for life in the 21st century (Ontario, 2016).

Studio Code is another website run by code.org, an organization dedicated to bringing coding into schools all over the world. They run a campaign called Hour of Code that promotes coding in the classroom and with children. This website also has ready-to-teach lesson plans and programs for kids any age ready to go for any teacher to use. The lessons are easy to understand for anyone with no knowledge of coding and was suggested as an excellent resource for educators to start introducing coding with. Another excellent website with tutorials and ready-to-learn material is Khan Academy. Although intended for a junior-senior age group, Khan Academy provides lots of tutorials for both students and teachers to learn coding. These are all excellent online resources participants suggested teachers can access to add coding in their classroom.

Code.org was one of the organizations that was discussed in the literature review as a resource for integrating coding into the classroom. Kids older than 4 up to any age can participate in the hour of code and learn a little bit about CS. This year 198,473 hour of code
events took place all around the world – 4,655 were held in Canada (Code.org, 2015). In the literature we saw that there was a heavy emphasis on using the resources from code.org in US, however with the participants it is clear that they are useful in Canada and elsewhere as well.

4.2.2. **Hardware resources were used to enhance coding learning in the classroom**

Online resources were a suggested asset for teachers to start coding with, as they require minimal material and are easy to follow and implement. Participants indicated that an exciting and wonderful addition to online websites and programming was being able to see the programming in action. For example, students can actually code a robot to follow instructions with the Lego Mindstorms kit. This was a yearlong project at Diane’s school that was incorporated into their science units.

Another interesting device mentioned by both Diane and Stacy is the Makey Makey. It is a kid-friendly circuit board that can turn any object that conducts electricity into a remote control. For example, students can set up their Makey Makey so that tapping a banana registers into the computer as a mouse-click. It works well with Scratch – the coding allowing students to see their programs in action. For example, with Scratch kids can code a program to send a message when a button is pressed. They can then attach Makey Makey to their computers and maybe a toy or fruit they want to act as a button. The object they have attached to the Makey Makey becomes a remote; when they tap it, something happens on the screen. Diane mentions that this device can be used in a multitude of ways and is always exciting for the students to participate in. Many of these devices are new, thus it is hard to see them written about in research or journal articles- resulting in an absence about this topic in the literature review. These are current experiences by participants that have paved the way for future research in this area.
4.2.3. Participants relied on various people and organizational resources to develop their strategies in teaching coding

In addition to the online and physical resources available to teachers, there are many local organizations committed to bringing coding into the lives of children. Many of these organizations are led by developers who work with coding in some way and know its benefits. Diane has had the experience of meeting and working with two of these organizations, one called Kids Learning Code and the other called Maker Space programs run by STEAM Labs.

Kids Learning Code is an organization that runs workshops, after school programs and Camps for girls and boys between the ages of 8 and 17 are encouraged to collaboratively build upon a variety of creative projects and further develop their technology skills. Their workshops are led by developers and technology experts who take the concepts they work with and modify them for a younger age group. Their programs are very practical, hands-on and relevant to the current industry of technology. For example, students could collaborate in groups to create a company, its product and its branding for a week long camp using various design, planning and coding software such as Scratch and Makey Makey.

STEAM Labs is an organization that offers a maker lab space for both kids and adults to have access to the latest technology use these devices to bring their imagination to life. They provide workshops for people of all ages who wish to learn about digital fabrication, coding and electronics. Similar to the previous sub-theme, these organizations are very local to an educators’ environment and newly formed. They are not seen in literature but play a huge role in helping local teachers learn and adapt to coding.
4.3. Participants Acknowledge Various Challenges When Attempting to Integrate Coding into the Classroom.

Computer Science is thought of as a difficult subject, yet many people are not aware of what it actually is (Tucker, Derek et al., 2003). Participants confirm that most of this is a misconception carried by many educators, however this misunderstanding is one of the largest obstacles to practicing coding in the classroom. When combined with a lack of available resources and supports, it becomes even harder to accomplish for some teachers. Even if these two barriers were overcome, participants say the lack of documentation or curriculum expectations prevents the teaching of coding from being carried throughout the grades a student goes to. Participants acknowledge overcoming some of these obstacles and continuing to work on others because they realize the value of bringing coding into their classrooms.

4.3.1. A lack of understanding of the topic is a challenge when trying to implement coding

As a teacher cannot teach a topic that they themselves are unfamiliar with, lack of knowledge or experience is a primary challenge that educators may face. Teachers need to first learn what coding is and become familiar with it. “I think that they need something. I don’t think you can just say to a teacher who has absolutely no background, go teach computer science. No, I think you need something, an AQ course or something,” believes Stacy. Both Stacy and Diane indicate that learning and teaching coding work hand-in-hand. A teacher will never know the answer to everything, they learn with the students – they are simply a few steps ahead. I found my experience, parallel to their experience, especially with scratch, because I would say to them, “Girls, just yesterday I figured out how to make one.” Some of them were even teaching
me thing...just be open about working through the problems together. It was kind of a collaborative thing.

It is important for teachers to know that coding is a highly collaborative field of study; it involves high-intensity problem solving and inquiry skills which are all built and solidified in a team environment. Thus, learning alongside one’s students, as Dianne does, is one way to possibly teach it Furthermore, literature shows that it is important for students and teachers to understand the breadth of computer science and the value of computer science skills so they can make informed decisions about why it is important. It is equally important for school leaders to understand what constitutes computer science as they try to engage students in developing these foundational skills (Google, 2014; Tucker, Derek, Jones, McCowan, Stephenson, & Verno, 2003). All stakeholders that partake in this initiative of integrating coding need to have a clear understanding of the topic to teach it effectively.

4.3.2. Lack of resources pose a big challenge to teaching coding

Stacy summarised the issue of trying to teach coding, but not having the resources to do so perfectly:

Some of my challenges was people not having decent technology, their computer might look okay, but when you look at it, it does not work properly and they need an update but it is expensive to go out and buy an Apple computer and you will see that in schools- you have so many ideas and so much you want to do but if you are dealing with older computers or tech you just cant get the job done, or if you do not have proper Internet connection. Trying to do stuff that everyone can access. It is obviously the learning goals that you want to meet but at the same time you want to be innovative and cutting edge so
having that delicate balance of like bringing it down to every one’s level but also being cutting edge, that has always been a problem for me.

Diane also alluded to the availability of resources as a major issue in other schools. She mentioned that her school had a huge focus on technology and so she had the latest resources available. However, she mentions that other teachers may not have what they need to teach coding.

The lack of resources being a challenge in integrating coding in the classroom was also identified in the literature review, however it was not one of the prominent challenges. According to Bauer & Kenton (2005), the lack of teaching and planning time were key reasons, with lack of resources as only a minor challenge that needed to be overcome. Other concerns were outdated hardware, lack of appropriate software, technical difficulties, and student skill levels. Through the participants’ voices, it appears that lack of technology may be a larger obstacle than identified in literature.

4.3.3. Participants indicated that the lack of a standard curriculum is a challenge to teaching coding

Even if a teacher was willing, aware and knew about coding education, it becomes difficult for them to plan their lessons without a curriculum to follow. A lot of major coding concepts are embedded in many subjects like Math, Science, and Language Arts. Both Stacy and Diane use coding-oriented software for lessons in these courses. However they admit that, for a new teacher, it can become difficult to make these connections and form their learning goals in addition to all of the daily work they already have to do. Furthermore, Stacy stated that there is a lot of essential ideas that can be integrated within a coding curriculum. The entire learning process in coding that begins with ideas, planning, design, testing and debugging can be
connected to many 21st century learning skills that students need to know need to be put in a way teachers can understand and implement.

I would like to see a separate curriculum similar to the BC curriculum we looked at like an applied science curriculum. Probably it would fit in the science curriculum but it’s really packed there, but also language or math – because of algorithmic thinking so we could integrate it or as a separate curriculum and call it applied skills and learning. A separate curriculum is what I think would be best. I think there is so much we could do… so much we can create in meaningful ways [and] position ourselves for a new economy, 21st century skills that we need that I don’t think we are doing enough in school.

Diane agreed with Stacy on the idea of a standardized curriculum that includes coding, but meanwhile her school is thinking of making their own to ensure that all teachers meet the school’s expectations. Both participants strongly agreed that a separate curriculum or guideline for coding education would be helpful to teachers attempting to help students be ready for the 21st century.

In literature, too, it is seen that the Ontario education initiatives orient more towards pushing the use of technology rather than a separate curriculum that focuses on all aspects of technology, like understanding how to use is as well as how it works. According to a study by the Ontario Public School Boards Association (2013), the idea of Computer Science education in elementary schools is still a vision and needs to be acted upon—a statement that aligns with the concerns of the participants in this study. The reasons for the OPSBA wanting to integrate coding in schools are similar to the participants in the belief that it will help students with developing creativity, critical thinking and collaborative skills. From the participants’ responses
and literature, it is advised that some initiative should be taken in documenting the work teachers are doing to help other teachers add coding into their class.

4.4. Participants Maintain a Commitment to Teaching Coding by Pursuing Their Natural Teaching Interests and a Lifelong Commitment to Learning

Both participants showed passion and commitment to teaching computer science to students. They both agreed that this was based on many different factors, both internal and external. Both participants agree that their personal interest in learning about coding or technology in general is one of the main reasons they even began to teach coding. They also alluded to the importance of education in sustaining this teaching. Most importantly they highlighted the need of support systems in order to experiment and feel successful in trying to incorporate teaching coding into their programs. Both participants felt that these three factors were important in sustaining their commitment to teaching and integrating coding in their classroom.

4.4.1. Participants educate themselves using various mediums to learn new ways of teaching coding

Stacy and Diane had no background in coding or in how to teach it, so they both had to learn about it on their own. They both alluded to receiving a combination of formal and informal education to teach themselves about computer science concepts. Stacy agreed that her formal education was an asset in attaining her position as a technology lead in her school. She believes that it helped her in getting the position of a technology teacher when compared to other candidates with a higher seniority.

While Stacy’s formal education had a solid presence in her technology-oriented career, Dianne did not refer to her formal education as a factor in her commitment to teaching coding.
Instead, Dianne referred to various workshops and external training that keeps her informed and up-to-date on the latest trends in coding. She made it a continuous goal to attend various workshops and training sessions to advance her knowledge of coding and educate herself before she planned her lessons. Specific examples she mentioned are the Apple Coding Workshop which introduced teachers to Swift (an Apple coding language) and coding fairs, where she learned about tools that she could use in her classroom from different companies such as Maker Lab, Kids Learning Code, STEAM Labs and Dash and Dot Bots. Overall, both participants acknowledged that it is a continuous process of learning and that it took time to become comfortable with a difficult topic like Computer Science.

From the literature we know that shows that 42% of principals and 73% of superintendents say that there are no teachers available at their schools/in their districts with the necessary skills to teach computer science (Google, 2014). These are American statistics, there is a lack of Canadian or Ontario related statistics to this issue, however these statistics are still relevant to pointing a flaw in coding education. The lack of existing professionals indicates that current teachers would need to educate themselves if they wish to do coding work in their classroom. Although, Diane and Stacy had some technology education, they both keep learning and educating themselves in coding which is necessary for many teachers willing to take on this initiative.

4.4.2. Participants rely on various supports that help them stay committed to teaching coding

Given that this journey of becoming an educator with a coding lens is a daunting process, both Stacy and Diane continuously emphasize the importance of support in their journey. Diane referred to a colleague, James, who was familiar with coding and had a lot more expertise in this
work. James was an important asset to helping Diane in her journey to implement coding oriented activities in her classroom.

James made it easy on me, he gave me a sample project, he found out what I wanted to teach, what concept I wanted to teach in math, and he made a project for me. Then I went and re-did it so I would know how to do it myself. So he helped me a lot, he saved me a lot of time.

James was there to help support Diane’s learning because he knew the material. However, participants indicated that staff does not necessarily need to be familiar to coding content in order to help; they simply need to be supportive. “There is this philosophy at school that we are learning alongside [students], and it’s okay. [Staff] encourage you and congratulate you, like my principal loved that I was doing this… so I felt supported even though it was a challenge.” Diane had both support of experts in the field and the administration to feel motivated to continue experimenting on strategies to teach coding in the classroom.

Stacy was on the other side of the team; she was the technical expert. So she gives us the point of view of what she did to help other teachers who weren’t familiar with Computer Science or even just technology.

I would go into other classes and support teachers using technology, in different varying ways. Like I really listened to the teacher it wasn’t like me pushing an agenda. I was listening to where they were at, sometimes they were at a low level—they are still searching the internet, browsing and pulling resources and other times they are doing this really high level stuff like knowledge creation. So I didn’t scare anyone; I supported them in a gentle way.
Stacy also emphasized the need of comfort and motivation from others when it comes to teachers being able to learn coding education techniques. When it came to her own learning of coding, Stacy was fairly independent and self-taught. However, she sought help from the parent council and other businesses to raise funds to get technology in her lab. She noted that once the parents were aware of the initiative, they sought to help in any way they could.

Receiving staff and administrative support was very important to these participants and they were fortunate to have it. The literature suggests that nearly a third of principals and 24% of superintendents say their school or district have to devote most of its time to courses related to testing requirements, and these courses do not include computer science (Google, 2014). A lack of motivation and need on behalf of the principal and superintendent is also a cause for lack of support in some schools. This may pose a barrier to many teachers like Diane and Stacy seeking support of staff and resources in the school needed to successfully integrate coding.

4.4.3. Personal interest and the requirement to teach coding in their professions are what helped educators start and stay committed to this work

One of the main questions that arose from the literature review was teachers find the motivation to start this work of integrating coding in the classroom. Stacy suggests that since technology was always part of her life; it seemed to have just “found” her. She was always surrounded by technology from her childhood endeavors to even doing her Master’s degree and PhD. When it comes to teaching technology, Stacy said, “I do read a lot of forums but I would say 99.9% of the time was totally me. I am a big self-directed learner…It wasn’t easy, there were definitely some frustrating times when I couldn’t do what I wanted to do…” Diane states that she started teaching coding because of the environment she worked in. She recalls coming in as a Grade 6 teacher having to lead the Lego Mindstorms project – which contains a lot of coding
problem solving as the start of her journey. When asked if she would have started teaching coding if the Lego project was not a requirement she states,

I think that it would have come along some other way, because of where I work. For example, there was this parent a couple of years ago who was very interested into coding, or when I started coding club … it’s something that we are very much encouraged to learn to do so it would’ve been something that I would’ve gotten into it anyways.

According to the participants, the level of interest in technology or the need to do teach technology are key factors to teachers doing this work that later transforms into a continued practice in the classroom. Most of literature points to obstacles that teachers face that prevent them from doing this work (Bauer & Kenton, 2005). Bauer and Kenton (2005) point out that this work is carried by teachers who are interested in technology or see the importance of it, but then discuss the obstacles they face and not what drives them to keep going. Due to the lack of research, the possible incentives and motivation behind teachers who do this work was unknown. However, now after hearing participant voices, it is known that passion and demands based on work environment are some motivating factors for teachers.

4.5. Conclusion

In this chapter I analyzed transcribed interviews of both my participants to look for themes that emerged in their responses. I discussed four different themes which addressed a variety of areas in my research. I examined different positive outcomes of adding coding into the classroom which include student engagement and motivation. I also looked at various resources that help educators such as Makey Makey and code.org implement computer science concepts in the classroom. Furthermore, I went through some obstacles that both participants come across in this process of learning coding themselves and teaching it in a classroom. Lastly, I also looked at the
factors that contribute to teachers committing to this work of integrating CS in the classroom. These findings helped me answer my research question that asks how teachers integrate coding in the classroom and what effective strategies and resources help them do this work. In the next chapter I discuss implications for these findings, recommendations and further research opportunities for the educational community.

Chapter 5: Implications

5.0. Introduction

In this chapter I discuss the implications of my research study. First I provide a brief summary of my key findings and their significance. Then I discuss the implications these findings have on the broad educational community and for me as a teacher and researcher. I follow the findings with a concise list of recommendations for several stakeholders that are influential to this topic that I have come to know while doing this study. Next, I identify important areas of further research in coding education. Finally, I conclude by summarizing my findings and speaking to their significance to the educational community.

5.1. Overview of Key Findings and their Significance

My key findings in the previous chapters were organized into four main themes:

1. Coding in the classroom has a range of positive outcomes that help student learning and preparation for the future according to educators.
2. Teaching coding is dependent on the availability or knowledge of various resources to help support the teaching process.
3. Many challenges when attempting to integrate coding into the classroom arise when putting it in practice.
4. A teacher’s commitment to integrating coding in the classroom is sustained by the resources that surround them.

In terms of positive outcomes of coding, both Stacy and Diane admit that as soon as students are involved in a coding activity they see that their students are motivated and engaged. They also experience that coding is very versatile and can be incorporated at various stages and with different parts of the curriculum—confirming what literature presented. Lemaire (2014) also presents that coding teaches students to think critically and solve problems which is essentially teaching students 21st century skills that they need in the future. This ideology is also confirmed by both the participants’ beliefs about why they do this work.

Stacy and Diane both frequently refer to the various resources they had when trying to integrate coding into their classroom. Diane continuously names many online resources such as Scratch and code.org as helpful tools in her teaching. Hardware resources such as Makey Makey and Blocky were also used by Diane in her classes. We see that both online, hardware, and support staff resources have a strong presence in literature, however, both participants also mention the strong impact and presence of local organizations that focus on teaching coding to kids—this was a new finding that was not stated in the literature.

Coding is admittedly a new and seemingly tough topic to understand, this has been stated various times in literature. Thus when participants acknowledged the lack of understanding of the topic as a challenge it was not a surprise. Consequently, having a lack of resources posed as a potential problem for other educators as well according to the participants. Moreover, not having a standardized curriculum to follow made it difficult for teachers to make connections to coding and its relevancy to the current curriculum. The issue of the curriculum was acknowledged by the Ontario Public School Boards Association (2013) as a vision that still needs to be acted upon.
Lastly, participants state the importance of self-learning using mediums such as formal educations, workshops, online resources to help build their understanding of coding as a very important part of their journey. Diane strongly emphasizes the importance of having her colleague who knew coding and his support while Stacy retells her experiences as a technology support teacher to provide two different perspectives on the importance of supportive and knowledgeable staff. American statistics say that there is a lack of education professionals who know a lot about coding which can be used to anticipate a similar situation in Canada, however Canadian statistics are limited and provide no information on this situation. The experiences of the participants show that Canadian educator like themselves have to take the initiative and learn coding in order to carry it through in their classroom. Personal interest or the requirement to teach coding by administrators then played a big role in getting Stacy and Diane started and continue to be motivated in this learning journey.

5.2. Implications

In this section, I outline the implications of my research findings. I begin by discussing the broad implications of my research findings for the educational community (as a whole). Then, I discuss the implications of my findings for me both as a teacher and as a researcher.

5.2.1. Educational community

As demonstrated in literature and through participant interviews, coding in the classroom is beneficial in many ways. The educational community needs to know that coding is a big asset to the students and to increasing the motivation and engagement in their learning. This implies that coding can easily be incorporated in any subject and at any grade making it very easy to blend into any teachers’ classroom program. The students receiving this opportunity to explore
coding also learn many 21st century critical thinking and problem solving skills that are essential for them in the future.

The educational community should also be aware that coding comes with its own set of resource requirements. Teachers often will need online resources, different hardware resources, and support from staff and administration to carry this work effectively. The participants interview in this study consistently refer to the importance of support from their colleagues and principal to help motivate them. Furthermore, the local organizations that work on teaching students coding are an excellent help for the educational community.

It is also to be noted that integrating coding into the classroom is not only a journey of learning for the students but for the educator themselves. Many educators come in to coding with a misunderstanding of what it is, we have seen this in literature and heard it in participant voices. Thus educators need to teach themselves using resources that may be difficult to find. Most importantly the educational community needs to know that this is a journey that can be taken alongside your students. The educator does not need to know everything, but only enough to help students work through the problems together. That is part of learning computer science, it is part of the journey to make mistakes and learn from them.

Overall, we know from our participants that the dedication to continue teaching coding in the classroom stems from both self-driven and requirement driven teachers. Some educators who are driven by their interests, like Stacy, may seek to take this journey and learn on their own. While other educators like Diane may need to take on this journey as a requirement for projects and learn through the support of their colleagues. Whichever path the educator comes from, it is clearly implied through my research that a lot of self-learning and dedication are required to start this work.
5.2.2. **Personal implications**

When I started investigating this topic as a researcher, I was under the impression that coding has yet to set foot in Ontario education. Even through my research of literature, I seldom found research on coding specific to Ontario, although I found a lot of American literature. As I progressed in my research and sought out people to interview, researched local organizations and volunteered for some of them I began to realize the presence of coding education for students in the Greater Toronto area. After interviewing my participants, I learned that there are many educators who are taking on this work and bringing into their classrooms. The most important implication for me while doing this research was that educators do realize the importance and benefits of coding in school and are working towards it.

When I entered this research, I had the experience of working in the industry and starting to learn how to be a good educator. In the journey of learning about my topic I have tried to step into the shoes of a teacher and teach coding to students. I realized the complexity of teaching younger students how to code when I started an extracurricular coding club in one of my practicums. As an educator I tried many strategies and lessons to see which worked and why. Overtime I have improved in the way I incorporate these strategies using the information I have found in literature and in my interviews to guide me as an educator.

5.3. **Recommendations**

To make coding a consistent part of our education in Ontario, work needs to be done on all fronts. As this is a very new topic to educators and students it needs a lot of support and persistence from everyone involved. Below I list a few recommendations for each major group of individuals that impact the education community.
5.3.1. **Ministries of education**

- Establish a province-wide curriculum that incorporates coding ideologies, perspectives and activities for teachers to refer to and follow. This can be done as a separate curriculum or integrated into an existing curriculum.
- Provide ample opportunities for schools and teachers to attend workshops and seminars that allow them to learn more about coding and how it impacts their classroom.
- Communicate the importance of coding in the classroom through easy exposure to current research and videos showing proof of student engagement and learning.
- Provide funding to all school boards dedicated to providing technology upgrades and devices that can be used to increase student interaction with current hardware and software.

5.3.2. **Schools and administrators**

- Promote computer science education within the school and encourage extra-curricular programs focused on students experimenting with different technologies.
- Provide support and encouragement to teachers willing to take the step of integrating coding into their classes. Encourage them to try out various methodologies and not be afraid to make mistakes.
- Encourage and promote staff to support each other in this work and dedicate time in meetings and PD days to allow teachers to plan and carry forward this work.

5.3.3. **Teachers**

- Take the step to learn coding and strategies to teach it for the sake of your students and their future.
• Seek out support from within your school as well as local and online resources to help you in this work.

• Most importantly, do not be afraid to fail or to experiment with different strategies to teach coding until you find what works for your class.

5.3.4. Faculties of Education

• Expose student teachers to the idea of incorporating coding into their classroom and benefits of doing so.

• Dedicate a course in the teacher preparation program that focuses on technology and coding allowing student teachers to explore possibilities and have an opportunity to overcome their anxiety so they are prepared to carry forward this work and support teachers within the industry to carry it out, too.

5.4. Areas for Further Research

In this section I will highlight some further areas of research that I have discovered while carrying out research on the topic. Coding in education is a fairly new topic, especially in Canadian research, however I have noticed a recent shift in attention to this topic. So it would be interesting to see how coding has impacted the Canadian education system in a few years compared to now. It would also be interesting to see the student’s perspectives on learning coding and how it benefits them. Due to the nature of this research project, only two participant voices were heard – thus an extension to this research would be to see the views of various educators both that have experience in coding and those that do not. This type of study would help identify the barriers that teachers face even further and allows to reflect on how to eliminate these barriers.
I also noted the strong absence of curriculum documents for teachers to use as guidelines, it would be an interesting idea to see what resources teachers use to guide them instead. Moreover, what do organizations that hold workshops and lessons for students to learn coding use to plan them and how they organize their learning goals. The information from this type of study would help immensely in forming the initial stages of a curriculum document. Overall, I believe there are many different extensions to this research project that can help carry forward this work and see it in action in our educational communities.

In the literature review, the perspectives of both older male and female students were looked at, the literature claims there is a significant difference in the way both genders learn and perceive coding (Werner, Hanks, & McDowell, 2004). It would be interesting to see how this plays out in an elementary classroom. To see if the elementary classroom mimics what is seen in older students or not. This would help shine light on where the gender issue stems and possibly how to overcome it.

5.5. Conclusion

From literature, ministry action policies and from participants in this study it is evident that coding is important in a students’ education on many fronts. Coding allows students to learn and practice 21st century skills like critical thinking and problem solving but also exposes them to an understanding of technology that they will be using throughout their life. Like any new topic of study, coding comes with its own set of challenges for teachers. However, as we see with our participants, these challenges can be overcome by seeking various supports of staff, online resources and organizations that carry this work. Furthermore, we know that this barrier can be overcome if educational faculties, ministries of education, school staff and teachers all work
together to achieve this goal – for the sake of the one thing that drives educators to do their work every day – their students.
References


https://hourofcode.com/ca

http://liberatingsoftware.org/assets/Making_CS_Fundamental.pdf


http://www.eng.buffalo.edu/undergrad/academics/degrees/cs-vs-cen


Appendix A: Letter of Consent for interviews

Dear ______________________________,

My Name is Rajavi Shah and I am a student in the Master of Teaching program at the Ontario Institute for Studies in Education at the University of Toronto (OISE/UT). A component of this degree program involves conducting a small-scale qualitative research study. My research will focus on how educators approach teaching complex computer science concepts to elementary students in creative ways and what strategies they use to engage their students in this field. I am interested in interviewing teachers who have a background in teaching computer science and have demonstrated a commitment to computer science education. I think that your knowledge and experience will provide insights into this topic.

Your participation in this research will involve one 45-60 minute interview, which will be transcribed and audio-recorded. I would be grateful if you would allow me to interview you at a place and time convenient for you, outside of school time. The contents of this interview will be used for my research project, which will include a final paper, as well as informal presentations to my classmates. I may also present my research findings via conference presentations and/or through publication. You will be assigned a pseudonym to maintain your anonymity and I will not use your name or any other content that might identify you in my written work, oral presentations, or publications. This information will remain confidential. Any information that identifies your school or students will also be excluded. The interview data will be stored on my password-protected computer and the only person who will have access to the research data will be my course instructor Dr. Angela MacDonald.

You are free to change your mind about your participation at any time, and to withdraw even after you have consented to participate. You may also choose to decline to answer any specific question during the interview. I will destroy the audio recording after the paper has been presented and/or published, which may take up to a maximum of five years after the data has been collected. There are no known risks to participation, and I will share a copy of the transcript with you shortly after the interview to ensure accuracy.

Please sign this consent form, if you agree to be interviewed. The second copy is for your records. I am very grateful for your participation.

Sincerely,

Name: Rajavi Shah
Email: rajavi.shah@mail.utoronto.ca
Course Instructor’s Name: Angela MacDonald
Contact Info: angela.macdonald@utoronto.ca

Consent Form

I acknowledge that the topic of this interview has been explained to me and that any questions that I have asked have been answered to my satisfaction. I understand that I can withdraw from this research study at any time without penalty.

I have read the letter provided to me by Rajavi Shah and agree to participate in an interview for the purposes described. I agree to have the interview audio-recorded.

Signature: _____________________________________

Name: (printed) ________________________________________________

Date: ______________________________________
Appendix B: Interview Protocol Introductory Script

Thank you for agreeing to participate in this research study, and for making time to be interviewed today. This research study aims to learn about what strategies teachers use to educate elementary students computer science concepts in creative and immersive ways for the purpose of bringing attention to possible problems the educators need to overcome in the classroom in order to successfully teach CS. This interview will last approximately 45-60 minutes, and I will ask you a series of questions focused on your experiences in the field of technology, your beliefs about teaching CS and the strategies you find effective in the classroom when teaching CS. I want to remind you that you may refrain from answering any question, and you have the right to withdraw your participation from the study at any time. As I explained in the consent letter, this interview will be audio recorded. Do you have any questions before we begin?

Background Information

- What inspired you to pursue a career in the field of technology?
- What experiences contributed to developing your interest in computer science?
  - Personal?
  - Educational?
  - Professional?
- Can you tell me more about your experience working in the computer science industry?
  - Looking at your experiences as an individual in the industry, what was the biggest obstacle you had entering the industry
- Looking at your experiences as an individual in the industry, what are some of the benefits you see as part of it
- What inspired you to become an educator?
- How did you initially get involved in teaching CS?
- What range of roles have you had that involve teaching CS? Which organizations have you worked for in this role?
- What age groups do you typically teach?
- What is your current position in this area?
- [If applicable]: what grades and subjects are you currently teaching?
- [If applicable] Can you tell me about the school(s) you work in? (e.g. size, demographics, program priorities)
- [If applicable] How many males vs females are in your class?

**Teacher Perspectives/Beliefs**

- In your view, at what age are students capable of learning computer science?
- What specific CS concepts do you feel that students are capable of learning at particular developmental and educational stages? Can you provide some examples?
- Why do you think it is important for kids to learn about these concepts at an earlier age?
- In your view, is CS commonly taught in elementary schools? Why do you think that is? What are some of the barriers to implementing more attention to CS in elementary grades?
- In your view, is any regular teacher is capable of teaching basic CS, even without a background in the topic?
- Where do you think CS fits into the current school curriculum?
• Do you think it can be combined with any other strands? If yes then give an example. If no then why not?

• How do you think that learning computer science concepts might help develop students’ understanding of, and achievement in, math and science?

• In your view, is it necessary to have technology to learn computer science?

Teacher Practices

• Can you provide me with a few effective lessons that you may have done with students which you thought went very well?
  • Who / where were you teaching? (e.g. grade, context)
  • What concept of CS did you focus on?
  • What opportunities for learning did you create?
  • What was the students’ reaction towards this concept?
  • Why did you think this lesson went well? What outcomes did you observe from students?

• Can you provide an example of an activity that anyone can do with kids to help them get started with these ideas?

• What resources do you use to help you make lessons or exciting activities for kids to try?
  probe for specific resources

• How do you construct your learning goals for the lessons – how do you decide what is important to teach?
  • How do you know that a particular age group can learn a particular concept?

• How do you evaluate or assess that the students have understood and learned these concepts?
• Can you provide some examples?

• Looking at your experiences as an educator, what was the biggest success you had teaching CS? What made this lesson successful?

**Supports and Challenges**

• What range of factors and resources support you in your work teaching CS to elementary aged students?

• What range of challenges do you encounter teaching CS to elementary students? How do you respond to these challenges?

• What are some of the tools, software or hardware that you have used in your classes?
  - Which ones are free?
  - Where did you get funding to get the paid tools?

**Next Steps**

• What are your goals for developing your teaching of CS to elementary aged students?

• As a role model what advice would you like to give to a young mind who is just learning about computer science and what it has to offer?

• As an educator what advice do you have for beginning teachers that are new to CS and how to teach it?

Thank you for your participation in this research study.