INFLUENCE OF ROBOTICS ON SELF-EFFICACY AND INTEREST IN COMPUTER SCIENCE

by

RAYMOND TODD SCHENK

(Under the Direction of John Mativo)

Abstract

This study sought to understand the *influence of robotics on self-efficacy and interest in computer science*. It provides the background that has led to the current research need.

The study conducted was a double pre-test, repeated measures post-test quasi-experimental study, performed by the researcher to determine the effects of a robotics treatment on the interest and self-efficacy of introductory computer science students in secondary education in the United States. The Social Cognitive Career Theory of Lent et al. (2020) was used as the theoretical model for this study. The study found out that robots were a viable alternative to promote motivation and learning.

INDEX WORDS: Computer Science, Career Decision Making, Secondary Computer Science, Robotics, Career Interest, Self-Efficacy

INFLUENCE OF ROBOTICS ON SELF-EFFICACY AND INTEREST IN

COMPUTER SCIENCE

by

RAYMOND TODD SCHENK

B. A., Messiah College, 1984

MSITM, Touro University International, 2004

A Dissertation Submitted to the Graduate Faculty of the University of Georgia in Partial Fulfillment of the requirements for the Degree

DOCTOR OF PHILOSOPHY

ATHENS, GEORGIA

2022

INFLUENCE OF ROBOTICS ON SELF-EFFICACY AND INTEREST IN COMPUTER SCIENCE

by

RAYMOND TODD SCHENK

Major Professor: John Mativo

Committee: Elaine Adams
In Heok Lee
Timothy Foutz

Electronic Version Approved:

Ron Walcott
Dean of the Graduate School
The University of Georgia
May 2022

Acknowledgements

No great task is achieved alone. My faith, my family, and my friends are all the reason this process has been a successful journey. My wife Lynne, my kids – and even my grandkids – have been there to support, encourage, and even push me along this path. The endless sacrifices they have made to allow dad to frequently sequester in his office are without tally.

My entire faculty team, led by a true mentor in Dr. Steven Sweigart has enabled me the time and space as a full-time educator to achieve this body of work without missing a beat. Their support and encouragement enabled this study in a very dynamic period in history. This work was completed despite remaining COVID-19 protocols and limitations and would not have been possible without such an amazing group of professionals.

My advisor Dr. John Mativo has empowered everything about this work. His leadership and mentoring have been above and beyond at every turn. My entire committee: Dr. Elaine Adams, Dr. In Hoek Lee, and Dr. Timothy Foutz have each brought their expertise and wisdom to bear in a way that has enhanced every aspect of this work. They each represent the model of what great minds can achieve together with a common purpose. Thank you for sharing your wisdom throughout this entire journey.

This thought would not be complete without expressing gratitude towards my students. They are a constantly growing source of inspiration. My students never cease to amaze me at what they can achieve. To the extent that this work will lead to enhancing their lives, therein will I gauge its success. This chapter is completing, only to open others - to a world of new ideas and possibilities.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

The educational system in the United States is not producing enough highly qualified computer programmers for industry. Despite global need for technically trained Information Technology (IT) workers, educational systems struggle to meet demand. Since before the economic turn-down of 2007, technical positions represent one of the fastest growing job sectors (Akbulut & Looney, 2007). Despite growth, enrollment in postsecondary educational programs in IT have fallen up to 75 percent since the turn of the century, in both the United States and Europe (Akbulut-Bailey, 2012). Job growth continues despite recent off-shoring aimed at providing an increased lower-cost workforce. Domestic labor pools have room for expansion, as underrepresentation of women and minorities across industry remains (Ash et al., 2009).

To combat resulting shortages, educators seek innovative ways to attract and keep students in computer science and related studies. They seek to understand job selection from students' perspectives, (Ash et al., 2009) and change student perceptions of industry (Akbulut, 2015). Unfortunately, the educational system in the United States is not producing enough computer science graduates for industry demand. Recent Bureau of Labor Statistics (BLS) (U.S., 2021) show expected growth of 22% for software developers between 2019 and 2029, yet Darrell Bowman of the University of Indianapolis (2018) points to enrollment decline in computer science majors since 2004.

Educational programs seek ways to increase and maintain enrollments to close this gap. One recent attempt is the use of robotics, emerging from elementary schools to universities.

Currently in after-school programs and clubs, it has recently found new use within curricula. Educational programs are also providing students with exposure to industry to help dispel prevailing myths and stereotypes. The ultimate desired outcome is improved enrollments and highly skilled workers available to industry (Akbulut-Bailey, 2012).

Expanding the IT workforce begins with training pipeline growth across educational institutions. Increased participation of women, minorities, and other under-represented populations must play a significant role in this effort. Domestic workforce increases can address recent offshoring trends that seek workforce personnel overseas (Ash et al., 2009). This expansion is more apropos with the explosive advent of the Internet of Things (IoT).
Here again, BLS projections from 2019 through 2029 show growth in computer and mathematical occupations of 12.1 percent, the third fastest growing industry behind healthcare support and social services (U.S., 2020).

Postsecondary educational programs are focused on recruiting students into technical majors and career technical programs to close the manpower-gap with industry. A natural focal point is the decision-making process students use when selecting college majors. The Social Cognitive Career Theory (SCCT) helps explain how students choose career paths and how those decisions can be influenced. Lent et al. (2017) describe SCCT as a model for how people make personal career decisions, and how they control their own educational and professional development. Lent's model shows self-efficacy and outcome expectations as joint predictors of exploration goals, and self-efficacy as a quality predictor of career decidedness.

Choice of college majors often begins in high school. Students apply for college as early as their junior year. For secondary computer science educators, attracting students to computer science is an important recruiting practice. Point Loma Nazarene University researchers postulate that students do not select computer science because they often receive little-to-no actual information about what computer science entails, or are given *incorrect* information about the science, and what jobs are available to professionals in the field. They recommend better student engagement, with exposure to benefits of computing careers - to overcome misperceptions or stereotypical assumptions (Carter, 2006).

Blouin's (2011) research into self-efficacy and interest in computer science careers helps explain the effects of computer science courses on high school seniors. Blouin did not find a significant gender difference in terms of self-efficacy in computer science but did conclude a difference in self-efficacy when students are exposed to at least one computer science class. Blouin found males more interested in computer science, and that participation in high school classes could impact decisions towards a computer science career. Blouin reiterates that choice goals are influenced by the variables of interest, self-efficacy, and outcome expectations.

Project-Based Learning (PBL) is a modern educational attempt to generate interest leading to self-efficacy by engaging students in collaborative, hands-on pedagogy. Literature suggests robotics in this type of hands-on classroom create an engaging and highly motivational environment for student learning despite some divergent results as in Lykke et al. (2015). Students more engaged and actively involved in learning are better able to master and apply what they learn, and deal with challenges they will face in real-world occupations (Pullan, 2013).

Numerous studies have been done at post-secondary introductory computer science to determine the benefits and effects of robotics in the classroom, including Lykke et al. (2015), Cliburn (2006), Fagin and Merkle (2003), and Korkmaz (2016). Lykke et al. (2015) found robotics to be engaging, interesting and fun for students. Cliburn (2006) found students gained a better understanding of computer science. Fagin and Merkle (2003) showed negative results because of specific issues related to robot setup and limitations. Korkmaz (2016) found Lego Mindstorms Ev3 robots to have no meaningful effect on student attitudes towards programming.

There are numerous studies using robots in elementary and middle-school programs. They study robotics impact on student motivation, performance, and interest-development in STEM-related programs. Witherspoon et al. (2017) did an important study with virtual robotics and middle school students. They demonstrated strong results in favor of robotics usage in a virtual environment, and that robotics helps develop generalizable skills and maintain interest in programming in middle school. Kazakoff, et al. (2012) showed elementary robotics to offer improvement in students' ability to perform sequencing. Both studies demonstrate clear support for robotics and call for further research.

Secondary educational use of robotics suffers from a paucity of research. Liu et al. (2013) compared physical robots against virtual systems. Physical robots were actual Vex robots that students manipulated and programmed. Other students interacted with virtual robots in a simulated environment. They found no significant differences between approaches but identified measurable time savings using virtual learning. Majherová and Králík (2017) also compared virtual and physical robotics in the classroom. They found physical robots particularly motivating for students because students build them but expressed concern over classroom time spent building instead of programming. They found better programming focus using virtual robots because physical builds were not required. Tochacek et al. (2016) used Lego Mindstorms to determine if educational robotics helped secondary students develop programming skills. They concluded that secondary use of robots does increase quality of educational processes.

Building on Blouin's (2011) contributions is a logical next step to understanding how secondary educators can motivate students towards more computer science education in high school. It is important to understand what drives student interest and self-efficacy, and how to overcome obstacles that might otherwise dissuade them from pursuing computer science careers.

Robotics has been studied extensively before and after secondary education. It makes natural sense to determine if robotics can be of utility in high school. If robotics can drive interest in computer science learning, it can drive participation, and give educators an opportunity to remove stereotypes and promote greater self-efficacy. This in turn can shape future expectations, choice-goals, and ultimately majoring decisions for college.

This study is positioned to advance the knowledge base for secondary educators in computer science and related educational pathways, seeking answers to motivational questions within an introductory programming course in computer science. It can help bridge the continuity gap of knowledge as to how robotics can play a part in educational programming courses.

This study combined the best of prior research activities using robots, side-stepping issues others found to be confounding or attenuating of their results. Specifically, this study used pre-assembled humanoid robots programmable in five computer languages, removing assembly time completely. This allowed classroom insertion with little effort, avoiding student frustration with associated builds.

If better understanding of student motivation to remain in computer science programs can be achieved, better strategies to influence students to complete computer science pathways should provide more computer science undergraduate majors, leading to more qualified industry participants. Recruiting is further aided by destruction of stereotypes related to IT careers and shaping perceptions and career expectations (Akbulut & Looney, 2007). These misconceptions can be addressed repeatedly across secondary computer science courses.

Purpose

This double pre-test, repeated measures quasi-experimental study sought to determine the influence of robotics on motivations of secondary students to complete a computer science pathway in high school. Participants consisted of high school students enrolled in an Introduction to Digital Technology (IDT) class at a suburban high school in the southeastern United States. The data collection instrument was a student survey taken twice before and three times across a three-week study period. Treatment used Revolution: JD humanoid robots programmed in the Blockly language. Control group programming occurred in the Scratch Programming language.

Research Questions

1. How does the use of robotics in programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?
2. How does the use of robotics in programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?
3. Is there a difference between genders in how the use of robotics programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?
4. Is there a difference between genders in how the use of robotics programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?

Theoretical Framework

Social Cognitive Career Theory

This study used the theories of Lent, Brown, and Hackett (2000) known as the Social Cognitive Career Theory, or SCCT. Social Cognitive Career Theory emphasizes dynamic processes that shape interests in occupations and education (Lent & Brown, 1996). SCCT helps explain how people make career decisions. It is largely derived from Bandura's (1986) general social cognitive theory. SCCT focuses on cognitive-personal variables, including self-efficacy, outcome expectations, interests, and goals (Lent et al., 2000). Treatment used in this study was designed to affect self-efficacy and interest of students, while collecting information to observe any gender differences in outcome.

Self-efficacy is a key variable addressed by SCCT. It is considered "the most central and persuasive mechanism of personal agency" (Bandura 1986, p. 391). Self-efficacy is not a static trait. Rather, it is a dynamic set of self-beliefs linked to specific domains of performance and activity in both occupational and educational tasking (Lent & Brown, 1996). Self-efficacy is most affected by personal accomplishments, but the theory states four primary sources for acquiring these self-beliefs: Personal accomplishments, vicarious learning, social persuasion, and physiological states and reactions (1996). This study used a treatment involving robotics to provide students engaging, interactive programming experiences and provide key points of accomplishment geared towards improving self-efficacy through tangible robot interaction. Key to this was the robots' responsiveness to programming instructions created by students. Students programmed each of the three components of the Böhm-Jacopini theorem: Sequencing, selection, and iteration.

Outcome expectations are beliefs people hold about outcomes or results of specific performance of given behaviors. According to SCCT these beliefs are shaped by learning experiences. Expectations about a given career come from experiences relative to that field, and information obtained about different fields (Lent & Brown, 1996). Even though expectations are a contributing variable to behavior, SCCT posits that outcome expectations can be overtaken by a person feeling they might not have the skills to perform a given course of action, *despite* positive outcome expectations (1996). This study involved a treatment providing confidence-building activities in robotics to promote accomplishment and achievement in programming. It introduced students to robot interaction programmatically, expanding their understanding of robotic operations and the overall embedded-programming domain.

The SCCT Interest model is a subset of the Choice model, in which self-efficacy and outcome expectations result in variant levels of interest, which leads to goal selection. Increasing self-efficacy and outcome expectations per the model requires setting very task-specific and domain specific measures (Lent & Brown, 2019). Measuring self-efficacy and interest within the specific domain of secondary computer science can help determine if the treatment increases motivation to continue in secondary computer science.

Goals, according to Lent and Brown, involve a person's intent to perform a particular behavior. Goals are significantly affected by self-efficacy, interest, and outcome expectations (Lent & Brown, 1996). Goals are represented in two categories of SCCM, choice-content goals and performance goals. Choice-content goals are activities in which a person wants to participate, and performance goals are levels of expertise or performance to which a person aspires (Lent & Brown, 2006). This study looked at student self-reported motivation goals towards completing a secondary computer science pathway.

Self-efficacy, interest, and outcome expectations do not exist in a vacuum. Recent research has centered on contextual supports and barriers to career choices, delving into socio-demographic issues like strong parenting, role models, economic hardships, and a variety of other issues (Lent & Brown, 2006). Environmental factors can be split into two groups: objective and perceived. Objective factors include things that influence a person's career development, including financial support received and quality of education received (Lent et al., 2000). Perception factors exist because a given objective factor might influence two different people at different levels.

Selecting the Choice Model

There are three fundamental models within the Social Cognitive Career Theory. These include the career-interest, performance processes, and choice models. These models are all inter-related within the overall model. They are an interplay between social cognitive variables of self-efficacy and choice-goals, in how they guide an individual's career development (Lent & Brown, 1996).

Vocational interests develop from childhood and are influenced by environmental factors including sports, activities, crafts, mechanical activities, and others. Any of these factors can have potential career relevance (Lent & Brown, 2006). Reinforcement of parents, family, teachers, and peers, also influence an individual (2006). Working within a feedback loop of continued activity and feedback, adolescents refine skills and develop performance standards. This helps form a sense of self-efficacy with any given task. This affects outcome expectations for continuation in the future (2006).

Career-related performance modeling in SCCT has two distinct aspects: The level of attainment achieved by an individual with respect to the activity, and the proficiency with which the activity is performed. In this model performance is heavily influenced by self-efficacy, outcome expectations, performance goals, and ability (Lent & Brown, 1996).

Assumptions of the Study

Social Cognitive Career Theory's Occupational Choice model is the model for which this study was most aligned. In this model, SCCT a person's career interests tend to be centered on specific domains in which they perform better and interact with others like themselves in important ways. Self-efficacy and outcome beliefs jointly promote career interests. Both career interests and career choice behaviors can be influenced by self-efficacy and outcome expectations, with career choice being the most directly influenced (Lent & Brown, 1996).

The instrument for this study measured two domain-specific constructs: Interest and self-efficacy. The design and testing of this instrument were carefully considered to handle validity threats and provide confidence in the measurements thereof.

Rationale

This study sought to determine if robotics use in a programming course motivates students to continue secondary training in computer science by improving computing domain-specific self-efficacy and interests. Career interests lead to selection of career goals, whereas career choices involve decisions for which the individual has higher self-efficacy and for which they believe will lead to better outcomes (1996).

Positive influences can help shape opportunities into goals, and goals become actions. Interests are most likely to turn into actual goals whenever an individual believes environmental conditions are good. Some influences are driven by a person's seniors, cultural paradigms, and even hiring discrimination. Distal influencers are factors that affect learning experiences. These experiences are how domain-specific self-efficacy and outcome expectations are formed.

Examples include exposure to career role models and support for participating in specific academic or other learning experiences (Lent et al., 2000). This study provided domain specific learning experiences aimed at creating positive and engaging learning, to influence goals towards further educational choices in computer science.

Social Cognitive Career Theory provides “a vantage point from which to view the school-to-work transition process,” (Lent, et al., 1999, p. 297) and the choice model helps frame the results of applying a robotics treatment aimed at improving student motivation. It helps to frame any discovered motivational differences - or lack thereof, as well as any observed gender differentials.

This research is an important step towards understanding the impact of robotics as a motivator to attract and retain secondary computer science students. If robotics drives interest and self-efficacy for new students, new innovations to draw students into the programming domain of computer science are enabled.

CHAPTER 2

Literature Review

Bringing computer science to life for students is one of the true joys shared by computer science teachers everywhere. Dedicated educators want to see the excitement of discovery emerge within their students. Literature displays a landscape of studies and papers aimed at improving the ability to motivate students into better achievement. Computer programming advances at an ever-quickening pace, with computers touching every aspect of daily life.

A *triadic* set of concepts that influence each other exists to encompass the literature from which modern epistemologies and ontologies towards computer science education can be shaped. First, definitions of the origins and rapidly expanding domain of this emergent science must be considered. Second, it is important to understand the advent of pedagogy in this field, how it began, where it stands, and where it must progress. Finally, a theoretical framework from which to analyze research findings must be identified. These three broad inquiries are all rife with literature that explains an industry domain moving faster than the U.S. educational system. Solutions for the future must be grounded in literature, without a proverbial moment to spare.

Computer science education today faces continual growth and expansion of the field. Recent Bureau of Labor Statistics (BLS) (U.S., 2021) show that between 2019 and 2029, an expected growth of 22% in software developer need, with a median salary in the low six figures. Darrell Bowman of the University of Indianapolis (2018) points to enrollment decline in computer science majors since 2004 despite this growth. His work highlights several things. Primarily, he found manpower shortages in the Information Technology (IT) sector from the late

1990's and early 2000's, caused many companies to offshore hiring to achieve software development goals. This was followed by insourcing. Both trends attributed to a lack of talent emerging from US Colleges. This led to a serious misconception that computer science related careers were subject to potential out-sourcing and were therefore not a good career decision. He found many college major dropouts were attributable to lack of proper secondary preparation.

Few would argue that expanding use of computer science has come to touch every aspect of modern daily life. To begin to find solutions and understand a path towards increasing the technical workforce needed by industry for the future, clear definitions within this domain become foundationally essential.

Terminology and Definitions

James McGuffee (2000) refers to the debate on *any* definition of computer science as being as old as computer science itself. His paper on the subject explains that there are multiple narrow definitions people use that often exclude others who work well within the discipline, some definitions being concerned with only algorithmic and theoretical abstractions, ignoring the implementation concerns thereof altogether.

Computer science has grown from what Subrata Dasgupta (2016) traces back to a mathematician named Charles Babbage. Dasgupta asserts a primary caveat to the definition of computer science as, in fact, *science*. He has an entire book devoted to the understanding of computer science as the premiere newcomer to the world of sciences.

Merriam-Webster (2021) defines computer science as "a branch of science that deals with the theory of computation or the design of computers." This definition alludes to what sources agree are generally two broad branches of the field – hardware and software.

The Association of Computing Machinery and the Institute of Electrical and Electronic Engineers Computer Science Task Force (ACM/IEEE-CS) originally defined computer science as that which is drawn from the mathematical, science and engineering fields (McGuffee, 2000), but has since grown to the point that this same organization has shifted in its definition. They no longer define computer science with a knowledge-based curriculum, but rather a *competency*-based curriculum around an ever-growing set of new sub-fields like data science and artificial intelligence. Artificial Intelligence (AI) remains a new un-sponsored competency – still to be added into their curricular programs (Computing, 2020).

Artifactual Definition

In his book on the genesis of computer science, Dasgupta (2014) defines computer science not as a natural science but rather a science of the *artifact*. Artifacts that encompass computer science include computers themselves, programs that run on the computers, and resulting computational artifacts – results of work done by a computer. Programs are the most interesting of these to Dasgupta because they are *liminal* artifacts. Liminal artifacts are somewhere in-between the physical and abstract. Programs themselves are abstractions, but they must have a physical component aspect to them to be used. Dasgupta analogizes thoughts or ideas needing the brain to exist, to describe the liminality of programs. He believes liminal and abstract artifacts now dominate - and help define - computer science.

The literature consistently refers to the physical design and maintenance of computer equipment with the term *computer engineering.* Even though this study is using physical robots, it is seeking to determine if a physical robot can drive interest and self-efficacy for introductory computer programming students. Therefore, liminal abstractions known as programs, the former half of the dictionary definition – were of primary concern to this study. Using the Böhm-

Jacopini (structured programming) theorem as the curricular basis for the study, the research is grounded in a foundational theory of computer science that fits within all these definitions, as Böhm-Jacopini is a Turing equivalent. Students learned the three pillars of the structured programming, while producing computational artifacts: one virtually on a computer screen, the other in live physical space.

For this study, the term *secondary computer science* referred to subjects taught at the secondary level whose primary objectives are to teach students how to create, edit, and maintain computer programs (*programming*). A computer program was defined for this study as a set of instructions that a computer executes.

The term *computer languages* refers to the programming languages used to construct programs, which are translated by compilation or interpretation into machine code, that the processor of a given computational device can then execute. A computational device refers to any device that can run a computer program, to include robots themselves.

Origins of Computer Science

The roots of computer science go back as far as Charles Babbage. They give rise to a science touching every aspect of society. What began with a simple desire to automate tedious math has evolved into a domain in which Hollywood writers decry a future in which machines eventually take over. Nevertheless, mankind continuously invents new ways to weave automation into the daily lives of everyone.

Subrata Dasgupta (2014) took an exhaustive historical approach to the origins of computer science. He begins with the Latin root for the word computation, which is *computare.* This means to calculate, reckon, or figure out. From here the term *computer* originates.

Rather poetically, computers were originally *people*. People used extensive tables and repetitive calculations to complete solutions. This time-consuming and tedious work was what Babbage sought to automate. His impetus for machines to do repetitive calculations birthed the origins of computer science - firmly within the field of mathematics.

Babbage built his *Difference Engine*, comprised of a set of wheels and gears that could compute polynomials to the second degree (Dasgupta, 2014). Now a museum artifact, this early achievement was proven with prototyping and experimentation. This led Babbage to the bigger dream of a more *general-purpose* machine usable for *any* mathematical computation.

Early Skills for Computer Science

Historical qualifications of the founders of computer science show the skills they needed to bring computer science to life. Babbage was a mathematician, as was Alan Turing. They had engineering skills. They also worked with machinists and mechanical or electrical engineers to create their early prototypes, as well as design future systems (Dasgupta, 2014).

Babbage became unhappy with the inability of his Difference Engine to perform more than one type of computation. He discovered a loom inventor, named Joseph-Marie Jacquard, who used punch card patterns to operate an automatic weaving machine. This unusual machine became a foundational inspiration for his *Analytic Engine* designs (2014). His ability to apply technology across industries and competencies directly contributed to the rapid advancement of fledgling computer science.

These skillsets were central to many of computer science's founding inventors both in their ability to abstract mathematical processes and theories into instruction sets or machine sequences, and to how they electro-mechanically created machines to carry out instructions in physical space-time (2014).

Herman Hollerith earned his PhD in 1890 with a dissertation describing a system for using electromechanical machines to tabulate census data. His degree is considered the first doctoral degree in computer science. His view held that automatic computation would lead to automated processing of all manner of data. The ensuing tabulators that emerged, like the IBM 407 accounting machine, began the early and heady days of data processing. These systems had large components like keyboard punches, sorting boxes, and tabulators that processed counts. Results were still hand printed, but Hollerith began to reshape punched cards in both size and function. US Army health records were one of the first card systems that used *fields* to represent data elements, with values that could be punched into them. This was a manual form of *memory* on the cards (Dasgupta, 2014).

Hollerith founded the Tabulating Machine Company, merging in 1911 with two other companies. By 1917 the company was being run by Thomas Watson, who renamed the company's Canadian division to International Business Machines (IBM). The parent company adopted the name in 1924. There was now a major manufacturer able to produce large tabulation and computing machines for government *and* private sector use (2014).

Mathematical talent as a key underpinning of computer science has *always* run deep from some of the very axiomatic questions *about* mathematics itself. German mathematician David Hilbert posed three fundamental questions about math. First, is it complete? Second, is it consistent? Thirdly, is it *decidable*? Hilbert believed these all to be answered with a yes, because of his belief in absolute proofs, and formal mathematics systems. Kurt Gödel disagreed, showing that axiomatics are limited, and that building a complete axiomatic system of arithmetic using just whole numbers is not possible. This became known as Gödel's Incompleteness Theorem. In short, mathematical systems are incomplete and inconsistent (Dasgupta, 2014).

This was *not* the answer Hilbert was looking for and left the third question unanswered - until Alan Turing came along. During his time at Kings College as a fellow, Turing took on the third problem - that of *decidability*. Turing used his vision for an abstract symbol processing machine, as opposed to a physical device, to answer this question (2014).

Alan Turing

Turing tried to understand what he called *computable* numbers - numbers calculated by a *finite* process. While he was at it, he solved Hilbert's third question of decidability. His abstract machines, now officially known as *Turing Machines,* showed there are problems that cannot be decided, meaning they are unsolvable. This was the final *no* to Hilbert's questions (Dasgupta, 2014).

With this discovery, Turing was on his way to designing a system that could perform algorithms that *could* be solved in finite time, according to a *strictly precise* set of rules. He understood that an effective algorithm would guarantee a correct solution. Turing formalized his theory with a logician named Alonzo Church, developing the Church-Turing thesis. This thesis shows that any computer that can perform computations can be performed on a Turing machine, or in technical parlance, *there exists a Turing machine equivalent* (2014). Turing did not realize he was founding a brand-new scientific field of inquiry, because he believed all his work to be directly mathematical. He is often referred to as the father of modern computers nonetheless, because his work paved the way for modern electronic computers, and his theoretical computer is still in use today across academia. The reach of his work was discussed in a later series of works called *Automata Studies* in 1956 (Yatsko, 2017).

Species Bloom of Machines

In 1947, the Northrop Aircraft Company commissioned the Binary Automatic Computer (BINAC), the first of its kind. It was an electronic digital computer, with a main processor section, input/output equipment, and a memory system made up of liquid mercury tubing, that could 'store' 512 words of data. It sported crystal diodes arranged as gated circuits. The BINAC performed the four basic arithmetic operations. It followed switching instructions, moving data between registers, and had a go-to/jump capability across instruction sets (Auerbach et al., 1952). It was never usable in production at Northrop but does brag to be the first computer ever delivered with a *user's manual.* Despite their limitations, these early computers as machines reflect the contexts that created them (Tatnall et al., 2013). These are the ancestral parents of devices held in pockets and worn on wrists today. These early machines taught the very lessons from which modern technology has evolved.

Across the proverbial pond, in 1956, a British computer engineer named Tom Kilburn completed production runs of several machines. His first was the Manchester Baby, followed by the Cathode Ray Tube (CRT) equipped Mark I, and then the Mark II version known as the Mega-cycle machine, or Meg as it was affectionately called. Meg replaced valve diodes with solid state versions, offering 10 times the clock speed of the Mark I. To prevent serial CRT speed bottlenecks, Meg used parallel CRT memory over 10 bits. Kilburn followed these projects with Muse, a higher speed computer that compared well to the current IBM Stretch and Univac LARC projects (Anderson, 2014). Kilburn pushed for a department of computer science to be established in U.K. higher education. His final build project was the MU5 which focused on being able to run higher level languages (2014).

Computer Memory: Enabling Generalization

One of the most prescient 20th Century advances of computer science emerging from the late forties to mid-fifties was *core memory* in a computer. Attributed as far back as 1946, IBM didn't use core memory until 1955, but they were first to use it in their 701 and 702 computers.

With increases in system memory, *applications* for computers began to emerge. While the world was moving to core, Sperry Rand sold drum memory computers to the US Civil Aeronautics Administration to track aircraft flight plans, and the John Plain mail-order company used a special purpose Sperry Rand machine to tabulate which ladies clothing items sold fastest (Gray & Smith, 2004). Business had suddenly entered the fray of computer science.

As core memory emerged, it upgraded many drum systems until 1960, when transistor memory marked obsolescence for both. Born in 1948 at Bell Labs, transistors took over to world of computer hardware, marking the dawn of the second generation of computing devices (Gray & Smith, 2004). By the early 1970s, Random Access Memory (RAM), rapidly followed by Dynamic Random-Access Memory (DRAM), marked the first time that large amounts of information could be stored on a single chip without serial access (Singh & Singh, 2016).

Computer memory today is a field unto itself. USB memory chips are fashioned into jewelry that can be worn. Memory *forensics* analyzes criminal and malicious software presence in memory, both in volatile and non-volatile memory alike. New tools preserve volatile memory like Random Access Memory (RAM) that loses memory if power is removed. Investigators can often retrieve what would be otherwise lost forever (Hannon, 2020).

Early Languages

Formula Translation, better known as FORTRAN, was written to enable mathematical formulae to be calculated by a computer. From there, two dominant languages emerged as the first machine-dependent higher-level languages: ALGOL and COBOL. Algorithmic Language (ALGOL) was written between 1958 and 1960 to support scientific calculations. Common Business-Oriented Language (COBOL) handled business-logic data processing. These languages propelled software development. Writing programs emerged as a response to rising costs of coding large computers requiring *machine-dependent* instructions (Nofre et al., 2014).

Auto-coding routines began to automate routine conversions and other machine processes. One example is converting from binary to decimal. Low-level translating paved the way for *translation* to emerge within computer science, ultimately evolving into modern compilers and interpreters, capable of translating not just one language into machine code, but from one language into *another*. Universal languages removed programmers from dependency on a given manufacturer's specific machine code (Nofre et al., 2014).

Language vs. Math

Mathematics has always been the ancestral tree-trunk from which computer science has grown. Yet it was a language theorist whose work was often disregarded in the language theory domain, who became a central figure in the initial shaping of modern computer language syntax theory. Noam Chomsky's theories developed in computer science over the years, despite resistance from the computer science community writ large (Dasgupta, 2016).

In 1985 a dutchman named Edsger Dijkstra stated his philosophy of computer science, in a manifesto: Computers are mathematical machines, programs are math expressions, and a language is a mathematical theory. His axiomatic approach has also developed steadily but faces non-trivial skepticism in today's real-world programming environments (Dasgupta, 2016).

With modernization of processing, memory, and programming languages, global communications required communications standards - to govern the language of a new phenomenon called the Internet. The ensuing standard became known as the Open Systems Interconnect (OSI) model. The field of computer science is *still* in rapid-expansion and remains one of the most prolific academic publishing domains, ahead of both physics and engineering (Fathalla et al., 2020).

Modern Computers

Robots are the lore of countless works of science fiction. Robots assemble many of the products in use every day. With product expansions and diversifications across a global marketplace, industrial robotics are considered the best solution for manufacturing productivity and flexibility. Industrial programming, however, takes a lot of time and money. Pan et al. (2012) provide an example of how an arc-welding robotic programming setup can take eight months to complete, preventing small-to-medium size businesses from harnessing this technology. This is true even with robots that can be programmed online. Robotics has become its own rapidly expanding *programming* discipline.

Robotics has created a new type of computer programming - *live* programming. In live programming, a robotics system is actively operating (robots are considered *state* machines) so that the state of the robot can be seen and provide real-time feedback to the programmer. This process is relatively new. There is not a lot of research as to its benefits, but it is already introducing great potential to simplify robotic programming itself, and therefore improve accessibility across the domain (Campusano et al., 2019).

As if to make this point, two of the largest engineering growth disciplines from 2016 to 2017 at engineering schools in the United States were both computer engineering and computer science (referred to as *inside* engineering). Mechanical engineering remains the largest growth category in engineering, but the fact that engineering schools are emphasizing computer science underpins the growing need for people who can program these machines (Wang, 2018).

Computer science has a partner in engineering because of robotics. Computer engineering is considered a marriage between computer science and electrical engineering according to Professor Samuel John of Computer Systems and Network Engineering at the Faculty of Engineering, Department of Electrical and Electronic Engineering at the Nigerian Defense Academy (Computer, 2020). This new cross-disciplinary approach is becoming better acknowledged and well formulated. Rosenbloom (2004) has even gone as far as to create a framework for computer engineering which splits the discipline into two camps. His *analysis* section looks at the theoretical relationships between computer science and other science domains, and his *synthesis* components use a practical systems-hierarchy approach to engineer how various layers of inter-disciplinary issues interact.

New Foundations

Perhaps the most interesting merge-plot of computer science, mathematics, robotics, and engineering is the Internet of Things (IoT). Devices can be connected to each other and can cross-communicate. The IoT has rapidly advancing partners as well. Blockchain crypto currency is providing secured cashless transactions, and Cognitive Radio (CR) offers new communication channels and protocols for millions of devices around the world seeking to communicate with each other (Villamil et al., 2020). The *volume* of data to process is growing just as fast - big data is another domain requiring advanced analysis and processing.

Finally, rising use of computer science and mathematics creates more and more artificial intelligence. Deep learning premiered in literature around 2014. This research now accounts for thousands of papers annually and is considered critical for national economies and security (Tobin et al., 2019). There seems to be no limit as to the future within this domain.

Modern Skills in Computer Science

The ACM/IEEE-CS production of the Computing Curricula 2020 report is an international accomplishment of tremendous proportions. This document modernizes and defines competency-based skills curriculum for computing engineering, computer science, cybersecurity, information systems, information technology, software engineering, and now data science. Still to be added remains artificial intelligence. This document identifies skills among all supported competencies, defining curricular content for educators from which students can be made workforce ready. These skills run the gamut from programming to hardware engineering and support, from specializations in data science, to software development in cybersecurity. Computer science today enables specialization in a variety of competencies that are all well within generally accepted computer science (Computing, 2020).

Secondary Computer Science Education

The second triadic element to understand is how secondary computer science education has grown from its roots. Here the literature becomes thinner, begging for more study and review. Nevertheless, what extant literature does exist, provides insight as to how it all began.

Nothing highlights the dawn of secondary computer science quite like a paper written by Professor William F. Atchison from the University of Maryland (1973). In this paper he sets the stage for the advent of secondary CS education. His paper is a historical marker adeptly characterizing the needs of his day, and foreshadowing issues some 50 years later.

Dr. Atchison is considered by many to be a pioneer of computer science education. In his paper, he calls for computer science to be taught by math teachers. His description of computers as machines to be used for applied mathematics explains the connection. He believed math teachers could best bridge the training gap to teaching computer science, also known as *informatics* at the time. His paper followed the transitional growth of computer science education from graduate to undergraduate to secondary learning institutions (1973).

Dr. Atchison saw computer use exploding onto the educational scene, and teacher education was rapidly becoming an issue. He saw this issue pedagogically on three levels: computer appreciation, transferring computer use to other subjects, and teaching programming itself. He called for formal teacher education of at least a computer science minor. He laid the groundwork for vocational computer science education, paving a connection to modern Career, Technical and Agricultural Education (CTAE) department hosting of computer science in many states. He called for teacher training - even returning them for college courses in computer science. He also cited the National Science Foundation's (NSF) support for teacher education on the subject. Since computer science involved manipulating numbers, storing data in variables, and performing logical sequences, math teachers were best equipped to jump into the pedagogy of computer science according to Atchison (1973).

Anecdotally, the first computer science course of the researcher was taught by a math teacher, using the Beginners All-Purpose Symbolic Instructional Code (BASIC) programming language, on a terminal that connected to the county's computer system, because the entire school had but a single terminal to use as its sole computer. Atchison was largely correct - math teachers did kick-start secondary computer science.

American Secondary Computer Science

The practice of computer science at the secondary level (high school level) in the United States can be considered at best, as a tapestry that varies by state. At worst a dated, disjointed attempt to create a skilled workforce unable to keep pace with industry. As with all education in America, each state determines their own standards, certifications, and curricula. In many states, authority to set these standards is further delegated to individual counties or cities, complicating the matter (Adrion, et al., 2016).

Computer science in all 50 states is historically an elective. Arguments abound as to whether it should be a required subject - and if so, what subject(s) would be displaced from those currently required. Today, more computer science courses are offered under the business umbrella than either math or science. Advanced Placement (AP) courses follow their respective learning objectives, but there is larger variance among non-AP courses on topics like game design, embedded coding, or computer repair (Burgiel et al., 2020).

Computer science began its life congenitally conjoined with mathematics. Sources are uniform in this regard. The very first machines performed basic mathematics, were seen as a source of reduction in human calculations and were programmed to solve repeated mathematical calculations faster than humans. It made sense that early educators would come from mathematics (Atchison, 1973).

Touching Everything

Computer science is a part of everyday life and educating students in the science is critical to their success, and to creating a workforce that can meet industry needs (Century, et al., 2020). Yet the age-old political football in the United States has been the fight between states' rights and federal control. Computer science education is just as affected by these forces as other subjects. States set their own standards and graduation requirements. These standards, and the assessments that measure them, are considered the backbone of today's educational system (Wilson & Harsha, 2009). This is not to say there are no *federal* laws regarding education.

Some national federal legislation acts seeking accountability, and ties funding or sanctioning to performance on standardized testing. This is where the fight is frequently centered. The No Child Left Behind Act of 2001 has been controversial for years, with direct sanctions that punished states not meeting standard assessments in reading, mathematics, and science for specific grade years. This program drove a lot of administrative emphasis for school systems, and computer science education is not yet considered *core* curriculum, despite the explosion of technology in society (2009). There are many who seek to make computer science a core subject, but it has not happened to date.

There is, nevertheless, reason for optimism. Initiatives aimed at bridging the gap between current educational practices, and those needed for the next thirty years are emerging. The Computer Science Teachers Association (CSTA) uses a network of advanced CS educators to share information and keep administrators and legislatures at state and local levels informed. The college board runs the AP Computer Science A course and newly added AP Computer Science Principles. These courses come with their own issues and challenges, but they do seek a more standardized secondary training level. States like Georgia and Texas are beginning to allow AP courses in computer science to count as science or mathematics credit towards graduation, as opening recognition of the importance of this field. The National Collegiate Women in Technology (NCWIT) organization has created the K-12 Alliance to reach more girl programmers in high school and post-secondary programs (Wilson & Harsha, 2009).

Pedagogical Content

Secondary computer science content taught matters. A recent study by Heidi Burgiel et al. (2020) found coursework that emphasized programming itself was the *only* type of learning that directly contributed to success in collegiate computer science. They concluded that being an AP class was *not* predictive for college success either. What mattered was *actual programming*.

Specifics in secondary curriculum vary by state. Georgia is a strong exemplar of how common course mappings are offered. According to their website (Georgia Department of Education, 2021), there are 10 available Information Technology (IT) pathways, each requiring three courses for pathway completion. All but one of the pathways begin with *Introduction to Digital Technology.* The sole exception is the *FinTech* pathway that begins with *Introduction to Financial Technology*. Pathways include: *Computer Science*, *Game Design*, *Internet of Things*, *Programming*, *Web Development*, *Web and Digital Design*, *Cybersecurity*, *Financial Technology*, *Networking*, and *Information Support and Services*. None of these courses are required for graduation. Qualified faculty is typically the driving factor as to what courses are available to students at any given school.

Under-Represented Populations

Computer science is stereotypically viewed as a male-dominated field. Unfortunately, this overlooks some very successful and important pioneers in computer science, like the ever-famous Dorothy Vaughan of NASA, who taught her entire department the language FORTRAN as part of the Mercury and Apollo space programs. She exemplified what ingenuity and creativity can accomplish, paired with grit and determination. Neither her gender nor her race hindered her determination to transition her team into the future (Hodges, 2018).

There is nonetheless a significant male domination within the computer-science related fields. If the talent pool in industry is to be expanded, no one can be left on the sideline - for any reason. In 2016 only 17.9% of all computer science bachelor's degrees were awarded to women

(Zweben & Bizot, 2016). That number demonstrates a significant under-representation of women in the field.

One of the best recent approaches to closing the gender gap in computer science *education* is being conducted at Carnegie Mellon University (CMU). CMU's program is actively creating an environment responsive to female underrepresentation in computer science. They actively recruit women. Of profound interest in their approach, is that they are bucking the trend of gender-difference approaches, claiming that women do not need a female-friendly curriculum, rather a cultural change. Stereotypes of male domination are challenged, and women are offered opportunities to show their skills and obtain leadership positions. They promote training *across* genders and highlight wherever women are succeeding. Their goals are laudatory as they seek to show how women can master the field, how cultural factors outweigh gender differences, and how traditional *Men are from Mars, Women are from Venus* (Gray, 1992) paradigms must be relegated to the dust bins of history. At CMU this is only the beginning, a relative first step at solving other under-representations as well (Frieze & Quesenberry, 2019).

Education Key to Recruiting Women

This study observed any differences between genders in support of a broader industry desire to broaden inclusion in computer science. Education is seen by many as a key to increasing female participation in a field that anticipates ever-growing demand. The literature has a plentiful supply of exemplar studies and efforts to attract and increase the presence of girls. One example by Gee et al. (2020) built analog games that teach computer science concepts like algorithms. Lynn et al. (2003) suggested that content needed to be better aligned with feminine values and interests, as well as recommending girl-gamer action heroes and adventure type programs that empower girls' assertiveness and competitiveness as well.

This underrepresentation is identified on an international scale in Kemp at al. (2019) who document the same lack of female presence in computer science in the United Kingdom, and by Khenner et al. (2020) who compare this issue between the United States and Russia. Both works claim that gender imbalances begin within education. They both see education as a primary starting point for improvement of underrepresentation.

Lucia Happe et al. (2020) did a literature review in search of effective measures to recruit more girls into programming among the many studies on the subject. One of the most salient comments made in their work relates directly to this study. They concluded that one important factor was to provide female students with engaging *hands-on* experiences. These, according to the writers, could increase their sense of belonging, by helping remove stereotypical assumptions of non-friendly environments for girls. They believe girls should be motivated to approach programming from a more social manner with positive role models to whom they can relate, among their many recommendations. If robotics has a different effect on girl programmers than boys, it is most definitely worth observation and measurement.

Other Populations

Women are certainly not the only populations that are underrepresented in computer science. Equally important are other groups to be brought into the field as well. Available workforce will never meet demand if everyone is not involved.

According to a recent Google-Gallup national research study, black and Hispanic students are 1.5 and 1.7 times more likely than their white counterparts to be *interested* in a computer science education. Eighty-four percent of parents responded that they believe computer science education is equally as important as traditional subjects, with twenty four percent believing it to be *more* important. Growth has occurred in secondary education, but barriers to entry for under-represented populations still exist. Black students report less access to computer science classes than white counterparts, as well as less exposure at home. Hispanic students are similarly disadvantaged (Wang, 2017). These disparities do not support manpower growth on the back end of the educational system.

Educator Shortages

The most common reason given for a lack of CS classes in schools, is a lack of qualified teachers. Sixty-three percent of administrators and seventy-four percent of superintendents made this claim. Teacher training was second place in reasons listed by over half of the same principals and superintendents (Wang, 2017). Another complicator revealed in this study is an apparent cognitive disconnect among administrators. Less than eight percent of them feel parents really want their students to get a CS education, when actual numbers exceed ninety percent (2017).

One solution schools use to cut shortages of computer science teachers is rebranding teachers from other subjects. Most computer science teachers today in the researcher's county are rebranded business teachers. Problems they report are supported in literature – specifically they cite need for additional training beyond certification programs and professional development opportunities currently available. This is particularly poignant as most computer science teachers report creating their own curricular resources (Sadik et al., 2020).

Menekse is relevant here as well. He attributes a distinct lack of coordination between higher education institutes and local school organizations as a contributing factor in the lack of teacher development. A majority of computer science professional development programs do little to improve or impact the way teachers deliver content and have little end results on student learning outcomes (Menekse, 2015).

Future Optimism

There are clear obstacles in preparing students for the workplace of tomorrow. Nevertheless, there are reasons for optimism. American motivational speaker William Arthur Ward is quoted as saying “The pessimist complains about the wind; the optimist expects it to change; the realist adjusts the sails.” With new attention to computer science at a national level, and more calls for computer science in classrooms everywhere, there is hope that more momentum will lead to increased training and staffing for education.

Fifteen states have mandated at least one computer science class be available every year. Some, like West Virginia and Arkansas require students to complete at least one such course. That, with increases from 27 to 38 teacher certification programs from 2017 to 2019, show promising opportunities for reaching more students. (Sadik et al., 2020).

Dominant Career Theories in Literature

The third component review for this study encompasses theories pertaining to how students select career paths and what motivates them toward these decisions. This grounded the study’s research in a comprehensive framework from which to explain results.

There are several historical career theories to have blazed the trail towards the modern career theories of today. Beginning with foundational trait-modeling theories, some have evolved over time into a post-modern or even qualitative tac towards career decision making. Others have remained more traditional - extending early theories and building on previous knowledge. They have roots as far back as the late 19th century, and the state of New Jersey.

Career Counseling Theory

The trail of inquiry into how career decisions are made, began with a man named Frank Parsons. No analysis would be complete without a look at the life and work of a man who scholars agree is considered the *father* of career counseling. Frank Parsons was born in 1854, in Mount Holly, New Jersey. William Briddick (2009) details the significant controversy to the historical understanding of his life, specifically the presence or absence of siblings, and circumstances resulting from his mother's early death. Sources agree he was at least in part raised by his aunts. Parsons overcame significant obstacles in his life to earn the myriad of professional successes he achieved.

Parsons founded the vocational guidance movement of the United States between the years of 1906 and 1908. In this brief span of history, he launched a powerful vision of how to help youth choose occupations (Jones, 1994). Helping youth better themselves was clearly his vision. Jones reports that Parsons died a short nine months after opening his Vocation Bureau in Boston. Fortunately, the bureau remained funded, and his work continued by a litany of others.

One of the key thrusts of Parsons' writing most germane to this study is this:
With respect to a well-chosen career, it should involve a clear understanding of an individual's interests, ambitions, aptitudes, limitations, resources, and what causes these things (Jones, 1994). Understanding one's interests and aptitudes is a common thread across Parsons' literature.

Parsons saw *interests* as the grouping of brain-activity impressions that cause a gravitational pull for similar impressions, ending up in a solidly formed interest (Briddick, 2008). Parsons famously told the story, according to Briddick, of a farmer, an architect, an artist, and an entomologist walking together outside. As they each describe different things they see, his story brought to bear the concept of differing interests, something he referred to as the constructive powers of human nature. Interests were clearly a focus of Parsons, but as a subject it did not have strong theoretical clarity for years to follow. Parsons passed away with an optimism for the future of the scientific method, and the expanding literature of his day (2008).

Karen O'Brien (2001) makes the point that Parsons was a champion for youth and underprivileged. He paved the way for career guidance within the education system and stressed the need for career exploration and highly individualized career guidance. O'Brien points out that Parsons' vocational bureau gave free career guidance for special populations. She later extended Parsons' focus on interests, capabilities, and environment into today's social justice movements.

Parsons watched the construction of the first subway built in the United States, outside his Boston home. He lived as Model T cars were emerging onto the streets, and Thomas Edison was creating motion pictures, the phonograph, and early plastics. He was alive as the Wright Brothers took to the skies. What impacted Parsons, however, was the darker side of the Gilded Age. He saw what he deemed excesses of capitalism, and he aligned with growing liberal thought, focusing on the poor and underprivileged - as well as how he might help them better their personal condition. His efforts led to a large collection of occupational information and increased training for teachers as career counselors (Zytowski, 2001). His life's work was picked up by others after his death - those who believed in his vision for America's youth.

Parsons began the landslide development of career theory because he had a passion for helping underprivileged youth understand how their interests and abilities could match up with career opportunities. He believed they should be carefully understood and matched with solid guidance and information on available careers.

Career Theory

Another career-centric theory was literally *named* career theory. It was the brainchild of a personnel clerk in World War II with a degree from the Municipality University of Omaha.

John Holland developed his theory resplendent with self-directed instruments to help individuals make career decisions.

Born in 1919, Holland was the son of an immigrant, who at one point considered becoming a musician. Instead, his career theory emerged despite significant anti-theoretical skepticism at his university. Professor Herbert Feigl convinced him that theory was central to the organization of useful information (Holland et al., 2009).

Holland broke his theory into a few main ideas. Work environments are not all the same. They can, nonetheless, be described with a topology. Individual personalities could likewise be described. Some environments are better matches for individuals than others. The typologies became known as Holland's RIASEC model: Realistic, Investigative, Artistic, Social, Enterprising, and Conventional. People resemble one or more of these typological values and can be described as such. These types are arranged in RIASEC order in a hexagonal arrangement with adjacent values having the most in-common traits with each other.
By matching environments with personality for an individual, a greater professional congruence can be achieved (Holland et al., 2009).

Holland worked extensively on his fundamental theory, with revision and modernization of his instruments for improvements and ease-of-use. In a paper by Holland et al. (1980), a diagnostic scheme was created to simplify older schemes and improve the process of helping individuals identify their *vocational identity*. In this paper they achieved better understanding of how his scales can be applied as checklists for counselors. They used *identity* as a construct for linking vocational psychology and personality profile.

Fundamentally, Holland's ideal process achieves congruence between an individual's personality and a given workplace environment. A strong similarity should lead to predictably better job satisfaction, performance, and stability (Holland et al., 2009). Results from testing congruence and differentiation, however, had mixed results. One test of Holland's theory showed no significant correlation between job satisfaction and congruence with two of his indexes. Another test was consistent with prior efforts failing to correlate congruence and job satisfaction. Further, their study found a negative correlation between another set of instruments running counter to Holland's theory. Others found significance but negligible effect size (2009).

One support for Holland's theory was conducted by Edwards and Whitney (1972), which sampled almost 800 people taking the Holland-created Self-Directed Search (SDS), a job guidance tool based on his model. This study concluded the ideal number of types is between four and six. It confirmed adjacent types as similar with opposite types indeed divergent. The study also supported Holland's SDS instrument.

One reason there are complex questions remaining about the utility and effectiveness of Holland's theory is that different authors use differing terms for their personality factors and inventories. Studies like Gottfredson et al. (1993) demonstrate that some of these tools cannot substitute for each other. Many of these tools are very useful in career counseling, but are mixed in support of Holland's theory, and there are significant calls for further research.

Social Cognitive Theory

Albert Bandura is a professor from Stanford University. His agentic perspective underpinning his Social Cognitive Theory (SCT) is a key foundation for later theoretical models in use today. He states that consciousness is the substance of mental life. It makes life manageable and fulfilling. To Bandura, cognitive factors are positive psychological factors independent of specific explanations of how they occur. Bandura believes that cognitive factors often predict human behavior and can be used to guide effective interventions (Bandura, 2001).

Bandura states that agency is seen as acts done with intentionality. He teaches that people look at the world around them, observing conditional relations between environmental events, and create outcome expectations from them. His theory states that as agents, individuals are planners and fore thinkers. They are also motivators and self-regulators. They set goals, but such goals are not necessarily what brings motivations to action. It is moral agency, however, that governs self-regulation. This agency can cause both inhibitive and proactive behaviors as people regulate themselves (Bandura, 2001).

Belief in one's ability to perform an activity and the capability to exercise control over their environment is foundational to human agency. This self-efficacy is important because efficacy belief is critical to adaptability and change tolerance. Self-efficacy plays a strong role in self-regulation of motivation according to Bandura. It also plays an important role in how people shape their lives and choose activities in which to participate (Bandura, 2001). This is a concept central to the determination concepts being tested in this study.

In a social context Bandura's theory holds up well. Agency helps individuals and groups adapt flexibly in diverse environments. It allows people to develop devices that compensate for a great deal of sensory and physical limitations or environmental constraints. It allows creative experimentation and effective social modeling. To Bandura, individuals value self-efficacy because it is applicable both individually and collectively in groups in which people pool resources for common good (Bandura, 2002).

An interesting aspect to Bandura's theory is in self-regulation. Accordingly, Bandura believes social cognitive theory on moral thoughts and actions shows the falsity of dualistic views he calls the *person-society* relation. Bandura believes that moral standards are not fixed regulators; they must be activated. There exist numerous processes in which people can disengage self-sanctioning from inhumane conduct. In short, people can relatively justify actions that can hurt others, based on the situation or context of the behavior (Bandura, 1990).

Social cognitive theory says that individuals are multi-faceted in their causal structures, which means that actions are not completely based on self-efficacy. Educators interact with students and institutions. This, according to Bandura, can influence the self-efficacy of students and teachers themselves in both positive and negative ways (Bandura, 1990).

A key tenet of social cognitive theory is *triadic reciprocal causation*. Behavior, cognitive function, and environmental factors function as independent determinants but also influence each other bi-directionally. SCT involves guided modeling towards developing mastery of a given skill, leading to greater self-efficacy. As skills are developed, practice and further guidance lead to perfecting those skills. Interestingly, simulated conditions serve well to increase efficacy, important for those guiding and providing modeling. Feedback is critical to that process. As self-efficacy increases, motivation can be enhanced. Self-efficacy can provide a profound effect on the direction of an individual's skill development. Mastery events can also lead to increased self-efficacy, as successes play a vital role in efficacy development (Bandura, 2001). Self-efficacy is important for deployment of a person's optimal use of their cognitive resources. If a person thinks they are not effective, they can become more self-judgmental and less task-focused (Wood & Bandura, 1989).

Social persuasion is another way people become convinced they have a skillset. Realistic encouragement can drive greater effort for success than in the presence of internal doubts. Others can help self-efficacy development by assigning tasks that lead to successes (Bandura, 2001). This is pertinent for educators. Understanding self-efficacy in students is a central aspect for this study.

Lifespan, Life-Space Theory

Donald Super was born in Honolulu, Hawaii. He moved to New York City when he was six, as his father was transferred to the national office of the YMCA. His mother was a teacher. His early work experiences showed him he needed more career exploration. Super considered himself fortunate to find *any* job in 1932, where he was hired as a half-time teacher at what is now known as Cleveland State University. There he became interested in issues surrounding unemployment, which is where his interest in career psychology was born, despite no formal psychology training (Pappas, 1978).

Suzanne Freeman (1993) elicited some interesting thoughts from Super in an interview. Super recalled that the term *career development* was non-existent when his own career began. Focus, at that time, according to Super, was about choosing a vocation, and was treated as a singular event. Super's lifespan, life-space theory was, in his words, segmental. Critics consider it fragmented. Super's stages of life were growth, exploration, establishment, maintenance, and decline. These stages represented the major phases of an individual's career. Super himself would later admit that he did not originally account for the fact that people often recycle through the stages when they change careers. He contrasted himself from the matching principles of John Holland by stating that his theory was more a matter of *successive approximations*.
To Holland, as people select and traverse their careers, they have experiences that help make better approximations towards their goals - over time.

Not entirely dissimilar to Bandura's theory, Super's theory involves the concept of self-concept. He first brought the term to light during a 1949 speech he gave in Colorado that was published in 1951. Super defines self-concept as the group of self-attributes that a person considers relevant to their occupation (Betz, 1994).

Super's work has been extended, revised, and refined for years since his death. His theory has lived on, with recognition for his own efforts at refining and improving his own framework. He considered his work a loosely organized set of theories unified around specific aspects of career development, held together by self-concept. He even allowed for the fact that his self-concept term could have been a mistake (Herr, 1997).

An interesting paper by Steven Weinrach (1996) involved studying the vocational interests of both Donald Super and John Holland. These two men are among the most prominent career-development theorists. In December of 1990 they both completed an identical battery of psychometric interests. They received their results in 1991. Each was going to write a published response concurrently to Weinrach's paper. Unfortunately, Super passed away in June of 1994. Mark Savickas completed the task for Super. Results showed an interesting array of similarity and differences between them, using the very instruments that they both helped create.

Transition Theory

Schlossberg's transition theory requires understanding transitions, how to deal with them, and then apply them to transitions within careers. Her transitions are broken into three groups: Anticipated, unanticipated, and non-events. Non-events are those that never happen, as in a job offer that doesn't occur. Unanticipated transitions run the gamut of anything unexpected, like car accidents, illness, or even surprise promotions (Schlossberg, 2011).

People cope with transitions according to Schlossberg's 4 Ss. *Situation* is the individual's given situation at the time of transition. *Self* is the individual's inner strength at dealing with the transition. *Supports* are the things available to an individual to help them cope with the transition, and *Strategies* are those efforts and actions aimed at changing the situation (Schlossberg, 2011).

A diverse set of studies exemplify the use and utility of transition theory. Transitioning to becoming a Registered Nurse was studied. It explored the transition of Enrolled Nurses (EN) to Registered Nurses (RN) in Australia. Results enabled development of a four-stage transition model, and development of important supports for student nurses navigating each transitional stage (Wall et al., 2017). Jane Goodman and Mary Anderson (2012) applied transition theory to retirement. They used Schlossberg's four Ss to develop supports and advocacy strategies to benefit professionals approaching and navigating retirement.

A similar study used transition theory to study withdrawal from sports by athletes. Many athletes consider leaving their sport a traumatic event. Using interviews of withdrawing athletes, they found both diversity and commonality in how athletes adapt to withdrawal. They did not find generalization but showed support and extensions to the model (Swain, 1991).

On a more pedagogical front, Schlossberg's model has been used to frame success barriers for veteran students in post-secondary scholastics. A recent study interviewed a wide array of faculty, staff, student affairs personnel and veterans themselves, to better understand the struggles and supports needed for veterans transitioning to civilian life - specifically those seeking to complete or continue their education. The study emphasized specific supports for veteran students and identified ways to help them in transition (Griffin & Gilbert, 2015).

Happenstance Learning Theory

Happenstance Learning Theory (HLT) is the brainchild of John Krumboltz. His theory states that everything people do causes unforeseen happenings or *happenstance*. He believes people can capitalize on these events. Happenstance theory is used to help career counselors improve outcomes for their clients, and help clarify and brainstorm career goals (Krumboltz, 2011).

Happenstance learning theory has four major tenets. The goal of career counseling is to help clients take proper steps to obtain a more satisfying career and life. Assessments for careers are used to *cause* learning, not to perform trait-occupational characteristic matching. Clients learn exploratory actions to benefit from unforeseen events. Counseling success is measured by how well clients do in the real world - outside the counseling domain (Krumboltz, 2009).

Happenstance learning theory emphasizes vectors over perfection - moving in the correct *direction* is more important than making perfect decisions at every turn. Gradual goals to move a person forward are preferred. *Successive approximations* over time create better outcomes at achieving career goals. Bringing together an understanding of what tasks can lead to successful careers, and successive approximations is central to successful career guidance (Krumboltz & Duckham-Shoor, 2001).

A study of HLT with regard to corporate downsizing was instrumental in identifying significant factors faced by employees in a recent economic downturn. Recessions are beyond the repair of career counselors but using HLT to frame events within the career of an individual helped clients make the best of their situation and take steps for future success and recovery (Krumboltz et al., 2013).

Another work of Krumboltz and Roger Worthington (1999) emphasized use of HLT in preparing students for school to work transitions. This paper identified the need to empower students to take actionable steps towards a *career*, not simply make a singular occupational choice. This was a significant approach and another example of HLT in action to provide proactive career decision-making.

Career Construction Theory

Mark Savikas founded modern Career Construction Theory (CCT). Nevertheless, he credits David Tiedeman as having established the blueprint from which the theory was birthed. His 2008 paper is complete with detailed crediting of Tiedeman's work, beginning as a constructivist, and moving towards an appreciation of Super's concept of agency. It ultimately ended up birthing the theory that individuals are *self-organizing* systems. Tiedeman differed from Super's theory of self-concept. Super saw self as a *state*, where Tiedeman saw self as more of a *process*. He saw people navigating work as a series of discontinuities transformed over time into a *career*. Purposeful action links the discontinuity of career changes into a navigational path of career (Savikas, 2008).

Career construction theory emphasizes a lifespan of development. It adopts a narrative perspective, which emphasize how individual life themes inform meaning on work experiences and shape career decisions (Rudolph et al., 2019). CCT has four career adaptability aspects: *Concern*, *Curiosity*, *Control*, and *Confidence*. Additionally, there are aspects of CCT that identify adaptation, including satisfaction and work engagement (Xie et al., 2016).

Career construction has been applied to career turnover analysis and was able to demonstrate evidence that career adaptability can mediate turnover. It was not able to reach a level of causation among the variables considered. Adaptability allows employees to build better satisfaction, which has a positive effect on turnover (Zhu et al., 2018). In a test of the theory, CCT was studied on adolescent career transition, providing significant support for the model when applied to adolescent students transitioning to work or higher education (Šverko & Babarović 2019).

Social Cognitive Career Theory

Social Cognitive Career Theory (SCCT) was the invention of Dr. Robert Lent and Dr. Steven Brown (2013). They unveiled their new theory in 1994. These men began their work in this field with three models they considered interconnected, yet segmental. Their models were designed to explain how people develop interest, make choices, and gain performance and persistence – all within vocational domains. They were seeking to bridge together all extant theories of the time in a complementary manner. They were interested in what else might lead to career choices besides general *interest*, and one variable that had their interest was *self-efficacy*.

The three models they developed have since been extended, with additional models added. A fourth model referred to as the *well-being* model in education and occupational contexts, was introduced in 2006. The original modes of *interest*, *choice-making* and *performance* had garnered significant attention by then. According to these scholars, the point of the original models was very content-centric. This means the models were to be used on a specific *domain*. They were all about the content, and little about the journey traveled by people across their careers (2013).

This theory was broad in its reach, and both Lent and Brown freely testify to standing on the shoulders of others. They openly adopted Bandura's (1986) triadic reciprocal causation framework. Therein Bandura described how behavior, cognitive function and environmental factors function as independent determinants but also influence each other bi-directionally across the years of a person's career. These variables are seen in SCCT's self-efficacy, social supports, and goal setting, and in both cases the assumption is that these factors affect career choices across the lifespan of a career. They also sought to expand and integrate a great deal of Holland's work as well (Lent & Brown, 2019).

Interest Model

The SCCT interest model is referred by Lent and Brown in their 2019 paper as a subset or built-in part of the choice model. They clearly state that this model can and should be studied independently as well. In this model self-efficacy is a predictor of outcome expectations, and each of these variables influences interest. In all three fundamental SCCT models, the independent variables and the dependent variables *all* come from within the *same specific domain*. If a researcher is studying nursing educational practices to increase available nursing students, then the variables under consideration as predictors and their dependent variables must be specific to *that* domain. Because of the direct affect self-efficacy has on interest, it is likely that simple confidence in one's ability to be successful can drive career choices even more than raw, objectively measured talent (2019). This is a significant, because in education when teachers want to inspire students to continue towards a specific domain of study - even if students are not fully ready to assume a career, confidence in their abilities can help propel them towards those talent-inducing behaviors.

Choice Model

The choice model simply adds the next step. Self-efficacy and outcome expectations drive interest, then interest drives choices people make in the direction of these interests. Specifically, Lent and Brown (2019) refer to the *choice goals* people make. For example, students in high school might choose to continue in a computer science pathway even though it takes three course electives. These types of decisions are driven by interest, which is driven as described in the interest model. Lent and Brown cite several examples of high interest to choice-goals correlation, as well as direct correlation thereto from self-efficacy and outcome expectations (2019).

The choice model also adds contextual barriers and social supports as two variables that influence choice goals, but with lower, more modest effect. Barriers tended to have the lower influence of the two according to Lent and Brown (2019). This does not make these variables unimportant, but rather explains the relative magnitudes in which they affect the model.

Performance Model

The performance model describes interplay between past performance and its ability to predict future goals and perseverance towards them. This is done via self-efficacy and performance goals which are influenced by outcome expectations. In short, if people perform well, and believe it is from their own success, this can boost self-efficacy. Boosted self-efficacy facilitates better performance by removing things like self-doubt or self-criticism and raising confidence. This can lead to increased persistence. Lent and Brown (2019) here again cite several studies demonstrating strong model support.

Modern SCCT

The newest use of SCCT as a model was tested using a longitudinal design on 420 college students. The test showed self-efficacy as a reliable predictor of decision outcomes, and if students were confident in their ability to perform processes of career exploration, that it did predict such activity. Cross-sectional studies also supported self-efficacy and measures of career indecision or decidedness. It also found support for relations between self-efficacy and outcome expectations to exploratory goals (Lent, Morris et al., 2018). A similar test found self-efficacy and outcome expectations are adequate, but not optimally fitting to data despite predicting exploration goals. Self-efficacy predicted career decisiveness (Lent et al., 2017).

A third test of the model was performed longitudinally to study the theory across gender and race/ethnicity. It held up in cross-sectional analysis integrating predictions from the interest, choice, performance, and satisfaction models. A longitudinal test in another study used two points one year apart, with similar results. These studies revealed the chicken-and-egg nature of flow between interest and self-efficacy. Flow runs from self-efficacy to interest, but opposite direction is now considered possible (Lent et al., 2014).

Focus on Self-Efficacy and Interest

Figure 1 (Lent & Brown, 2020) pictorially shows the focus of this study as part of the SCCT choice model. This study seeks to determine if a treatment of robotics can improve

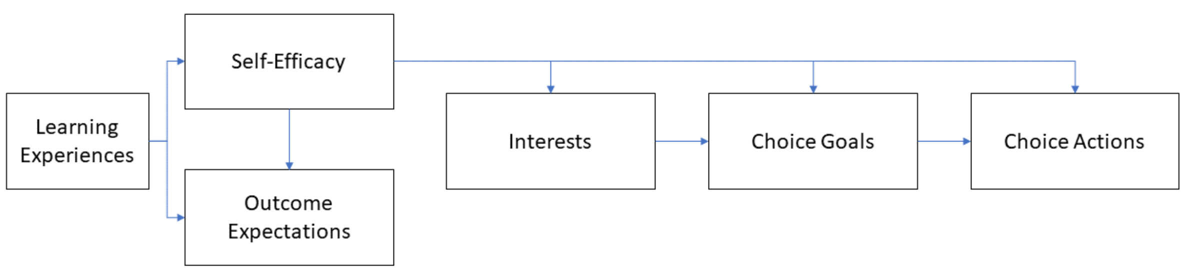
Figure 1

Focused (Selected) View of Choice Model

learning experiences for introductory programmers and thereby drive greater self-efficacy and interest which would lead to more students completing pathways in secondary computer science. This is an important part of recruiting more students into computer science.

Review Summary

Table 1 summarizes the career theory literature reviewed, determination if it is a theory



fitted to this study, and literature references reviewed for each theory. These theories show a brief overview of the evolution of theoretical knowledge from trait theories to modern comprehensive approaches to career selection.

Choice Model for Secondary Computer Science

In the SCCT Choice model, interests, self-efficacy, and outcome expectations are major contributors that shape goals. Self-efficacy drives interest, and influences outcome expectations. Outcome expectations can influence interest as well, and all three factors lead to major choice goals. A rise in self-efficacy can drive interest and outcome expectations, so an increase of self-efficacy and interest should indicate a positive effect on choice goals.

This study used a robotics treatment to determine its effect on student secondary career choices. The Choice model has strong correlational support across several model tests. Specifically, supports and barriers lead directly to goals through self-efficacy's path.
This supports the path to goals in Bandura's hypothesis that contextual variables relate to goals through self-efficacy. (Lent & Brown, 2019).

Table 1

Summary of Career Theory Review

Career Theory	**Key Attributes**	**Applicable**	**References**
Career Counseling	*Interests*, Skills Matching, Exploration, Individualized Guidance	*No*	Briddick, 1998 Briddick, 2009 Jones, 1994
Career (Holland)	RIASEC personality & workplace environment topologies (Trait)	No[a]	Edwards & Whitney, 1972 Gottfredson et al., 1993 Holland et al., 2009 O'Brien, 2001 Zytowski, 2001
Social Cognitive (Bandura)	Triadic Reciprocal Causation (behavior, cognitive function, environmental factors) *Self-Efficacy* and *Self-Regulation*	No[a]	Bandura, 1990 Bandura 2001 Bandura 2002 Wood & Bandura, 1989
Lifespan, Life-Space (Super)	Career Vocation in stages of successive approximations, later added recycling, *Self-Concept*	N	Betz, 1994 Freeman, 1993 Herr, 1997 Pappas, 1978 Weinrach, 1996
Transition (Schlossberg)	Careers through lens of transitions: Situation, Self, Supports, Strategies for coping with transitions	N	Anderson, 2012 Griffin & Gilbert, 2015 Schlossberg, 2011 Swain, 1991 Wall et al., 2017

Career Theory	Key Attributes	Applicable	References
Happenstance Learning (Krumboltz)	Benefit from unforeseen circumstances. Vectors over perfection. Capitalize on unexpected events.	N	Krumboltz, 2009 Krumboltz, 2011 Krumboltz & Duckham-Shoor, 2001 Krumboltz et al., 2013 Krumboltz & Worthington, 1999
Career Construction (Savikas)	Self as a process. Narrative on themes of career: Concern, curiosity, control, confidence	N	Rudolph et al., 2019 Savikas, 2008 Šverko, I., & Babarović, T. (2019). Xie et al., 2016 Zhu et al., 2018
Social Cognitive Career	Comprehensive models (4): Goals, Choice, Performance, Interest. *Self-Efficacy*, *Interest*, *Outcome Expectations* drive career goals.	Selected	Bandura, 1986 Brown et al., 2008 Lent & Brown, 2013 Lent & Brown, 2019 Lent et al., 2014 Lent et al., 2017 Lent, Morris et al., 2018

[a.] These theories were expanded by and helped shape SCCT.

A significant study supporting the choice model for this study occurred in 2008. The study involved *collegiate* computer science students. Findings held consistent to fundamental Gender data of males and females also fit, showing comparable interest and goal results. Results were consistent in Historically Black Colleges & Universities (HBCU) and Predominantly White Universities (PWU). It extended previous tests in computing disciplines, and results were consistent across a large diverse sample, suggesting good generalizability (Lent et al., 2008). Even though completed at the university level, this study demonstrated the Choice model as a strong foundational theory to support a secondary level study in the same discipline.

Need for Better Recruiting

Industry need for computer programmers continues to expand. There is a shortage of available talent documented in literature since the economic downturn of 2007 (Akbulut & Looney, 2007). Beyond the need for manpower, there is a growing demand to solve underrepresentation for women and minorities in this industry (Ash et al., 2009).

If robotics influences an increase in interest and self-efficacy among introductory computer science students an expectation of increased choice goals to persist in further training within a secondary computer science pathway should follow. Additionally, the model suggests comparable results across genders and ethnicities, which can provide further insight into findings. The SCCT Choice model is therefore the model best suited to explain findings from this study. If use of robotics can help attract and retain more students in pathways and classes, another tool is achieved towards producing a larger pool of talent heading off to post-secondary education with a desire to enter the field of computer science.

CHAPTER 3

Method

This double pre-test, repeated measures, quasi-experimental study studied the influence of robotics on motivation of secondary students towards completing a computer science pathway. The study was conducted with an Introduction to Digital Technology (IDT) class at a suburban high school in the southeastern United States. Learning concepts were restricted to programming introduction of the structured program theorem, also known as Böhm-Jacopini. This theorem posits that all programs can be constructed by using only three control structures: sequencing, selection, and iteration (Barone et al., 2014). These three constructs are considered the fundamental building blocks of structured programming. They were implemented in the programming languages Blockly and Scratch for study purposes.

The dependent variables in this study are domain-specific to computer programming: Self-efficacy and interest. Self-efficacy is a person's belief about their capabilities with regards to a given subject of study (Bandura, 1986). Improving self-efficacy can lead to greater interest, because an individual's learning experiences fosters intrinsic interest growth (Brown et al., 1989). This study examined the use of robotics on self-efficacy and the fostering of such interests in computer science – specifically a student's interest to continue in and complete a secondary computer science pathway. The independent variables for this study were the treatment use of robotics, gender, and socioeconomic status as defined by a student's eligibility for free or reduced fee lunches.

Research Questions

This study addressed the following research questions:

1. How does the use of robotics in programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?
2. How does the use of robotics in programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?
3. Is there a difference between genders in how the use of robotics programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?
4. Is there a difference between genders in how the use of robotics programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?

Design

William Shadish et al. (2002) provides a comprehensive look into designs for generalized causal inference, and their work is most applicable herein. They present a comprehensive look at the spectrum of designs ranging from weak quasi-experiments through pure randomized experimentation. Studies with randomized samples do not face nearly the structural and external validity issues as the quasi-experimental. More germane, however, is their guidance in how quasi-experiments can be designed to provide better resistance to validity threats and a more robust endurance under critical analysis. Treatment fidelity is also important, but with a single researcher that aspect of the study is controlled (Smith et al., 2007).

Quasi-experiments are conducted for the same fundamental purposes as true randomized experiments, with many of the same design elements, like control groups, pretests, and other elements that seek the effects of a treatment. The significant difference is lack of randomization, which is the defining separation (2002). As random selection is not feasible, this study therefore contended as quasi-experimental (Creswell, 2018). This is a commonly used design in educational settings as in Cera et al. (2020), Dsouza et al. (2019), Grimminger-Seidensticker & Möhwald (2020), and Mohd Radzi et al. (2019), because students are typically scheduled for classes outside the control of a teacher-researcher.

Short of time-interrupted series or regression discontinuity studies with random assignment, the strongest quasi-experimental designs have both a control group and a pretest. These two factors together provide increased inferential power to studies without randomization. In accordance with this wisdom, this study used both (Shadish et al., 2002).

One often under-used design idea to improve quasi-experiments is a double pre-test. This can help identify selection-maturation threats and reveal regression effects if the second pretest observation is unusually high or low. It can be used to help with correlation estimates between observations. Figure 2 shows the design notation as described by Shadish et al. (2002).

Figure 2

Design Notation

NR	O_1	O_2	X	O_3	O_4	O_5
NR	O_1	O_2		O_3	O_4	O_5

Treatment biases between the first two measures, may exist between the second two. This study design included a double pretest, symmetrically spaced a week prior to the study which included the second pretest on day one. Double pretests should be spaced at equal distances (2002). This table also shows the three repeated measures during the treatment period. Measures were likewise spaced exactly a week apart from each other.

Design Treatment Assertions

This study was designed to understand the effects of robotics as a treatment inserted into the academic pedagogy of teaching the Böhm-Jacopini theorem as part of an introductory computer programming class. The overarching goal was to determine if the use of robots in an introductory classroom could provide additional options for teachers to engage and inspire students, perhaps even improving student self-efficacy and interest by providing a more hands-on interactive experience that was more tactile and engaging than simply moving a sprite around the screen. Prior secondary educational studies in the use of robotics have shown mixed results. Lykke et al. (2015) cite numerous studies with results both good and bad, describing issues that include extra efforts to build and maintain or repair the robots. Other studies like Liu et al. (2013) have even gone as far as to study robot simulations as an alternative to actual physical robots that showed equal learning gains. Tochacek et al. (2016) found that the use of robotics educational experiences can improve the educational experience.

The Revolution J.D. robots used in this study were ready to use with a simple assembly by the researcher. These robots can be programmed in several programming languages and offer a solution to several problems cited in literature about past maintenance and assembly efforts required to use physical robots in a computer science class not typically tasked with mechanical assemblies and repairs. These robots offered the opportunity to provide tangible experiences without the previously burdensome overheads experienced in other studies. This could then provide a richer programming experience that could boost student confidence because of the tangible feedback robots provide, and drive interest because of the dynamic and stateful interactive nature of humanoid robots.

Böhm-Jacopini and Repeated Measures

The programming constructs of sequencing, selection, and iteration are taught, typically in that order. Sequencing is simply issuing sequential instructions to the processor, selection is branching based on the state of a boolean expression, and iteration is looping - repeating a set of instructions for some number of iterations. Teaching sequencing borders on trivial, in that any two consecutive instructions are considered sequenced, so this is clearly the easiest starting point. To use selection, sequencing is necessary. In turn, selection is necessary to understand how to exit a looping condition. As students move from sequencing to selection to looping, the content builds upon itself, making repeated measures a good measure of how their confidence and interest are being affected as they proceed along the path of adding each skill to their coding. Each part is integral to the next, so being able to see consistent growth not only helps a teacher understand where students begin to struggle, but it also reinforces what they have previously covered because of the additive nature of the content.

Contextual Historical Support

Quasi-Experimental designs have advantages and disadvantages that need to be clearly understood to fully appreciate where they sit within literature. As far back as 1966 Donald Campbell argued for the use of quasi-experimental designs and wrote to help encourage their use and bring awareness to situations in which their use is appropriate (Campbell, 1966). He acknowledged the necessity for researchers to be fully aware of which variables this type of design fails to control. He stated that all experimentation is flawed and use of a comprehensive list of validity criteria can help designs provide the ability to conduct research and experimentation, aware of results that might be vulnerable (Campbell, 1966).

Validity

Arguments against quasi-experimental design originate from validity threats caused by lack of selection randomization. Campbell (1966), like many, separates them into the broad categories of *internal* and *external* validity. Confounding and bias in selection are two of the most common problems associated with quasi-experimental designs (Waddington et al., 2017). Internal validity asks if, in fact, the experimental treatments make a difference in their specific experimental instance. External validity goes to generalizability – it seeks populations, settings, and variables to which an effect can be generalized (Campbell, 1966).

Cook & Campbell (1979) defined validity on a four-axis typology: statistical conclusions, internal, construct, and external. Statistical validity is based on how well the statistics used can properly demonstrate covariance (or lack thereof) between the independent and dependent variables. Internal validity is concerned with whether the observed covariance is a result of a causal relationship. Construct and external validities are generalizations. Construct validity is most frequently simplified across literature as the degree to which a test measures what it claims to measure. External validity goes to the extension of results across other settings, times, or populations for which the same questions are likely to be asked (Shadish et. al., 2002).

Internal Validity

Creswell (2018) lists major threats to a study's internal validity, considered below. Controlling for and understanding these threats to internal validity, minimizes any potential findings corruption.

History, Maturation, and Mortality are relevant to this study. They represent the largest exposure to risk based on normal classroom operations.

History: To protect against a history threat, the experimental treatment phase avoided unnecessary length that could allow events to affect results. Drills or other interruptions could distract from treatment or control unevenly. It was important to keep study length to such a time as to minimize this risk. To support this, the study treatment period was set to three weeks.

Maturation: IDT is an introduction class; students are not exposed to the pipeline when they begin this class. The experiment was conducted at the beginning of the year, completed within a three-week section, to mitigate maturation based on the nature of repeated instructions around the curricular content of this study. Maturation was a threat to this study because of the foundational nature of the Böhm-Jacopini theorem, and its repeated emphasis across all four years in the computer science pathway.

Regression to the Mean: Students were not selected based on extreme scores, rather by enrollment. This is not considered a significant threat; shorter treatment time nonetheless precluded larger chances of student interaction.

Selection: Selection was a threat to this study because it was non-random. Purposive sampling strategy was used, often referred to in literature as *representative* sampling. Students in these classes represent typical students for which this study is aimed. They are of similar age and grade level, and were purposefully *not* selected based on any socioeconomic, racial, gender, or other specific attributes. The two IDT sections were *representative samples* of the population of secondary students on whom this study was focused.

Mortality (Attrition): Normally more relevant to longer studies; a short duration minimized the possibility of students being switched out of class, dropping, or otherwise leaving. Freshman schedules are often load-balanced and moved around based on school/scheduling conflicts and requirements. As such, the study was scheduled after the initial school-wide enrollment 10-day count at the beginning of the year.

Diffusion of Treatment: If not managed, this threat could have been a factor for this study. The control and treatment groups were different periods. Different class periods minimized conversations between students and maximized separation. Here again, a shorter duration study greatly reduced cross-chatter between groups.

Compensatory/Resentful Demoralization: The treatment was provided to the control group right after the conclusion of the study. Students were told that all sections would receive identical training, but that various units may differ in sequence. Like the John-Henry Effect, control for this involved keeping the study shorter in duration.

Compensatory Rivalry: To ensure the control group did not feel undervalued, they were made aware of the value they contributed to the study. They were also made aware of the fact that they too would receive the treatment following the study.

Testing: Students in both groups were given a double pretest/posttest before and after the study. With a three-week treatment period, they were likely able to remember the types of questions they were asked, but questions in the survey were focused on their interest and self-efficacy. They were designed to not give away information about the treatment to minimize any testing effect. Questions were randomly delivered each time.

Instrumentation: Discussed at length below, the instrument before and after the survey were the same, controlling for this threat, and as mentioned, piloted for validity. This effectively prevented instrumentation from attenuating study findings (Creswell, 2018).

John Henry Effect: This effect is *reactivity* - when a participant knows they are being evaluated and works harder than normal as a result. For this study the two sections of classes knew they were participating in a study but were not briefed on specific details that might encourage over performance. This was not considered a significant threat to this study.

Hawthorne Effect: Much has been claimed about this effect, in which enthusiasm for being a part of a research project can cause increased performance. In education, this effect was studied extensively by Desmond Cook in the late 1960s. His work found that the effect was age-related, among adults and their awareness of being in a study. In education research, awareness among students of a research study has not produced empirically significant presence (Cook & King, 1968). This was not considered a significant threat to this study.

Rosenthal's Effect (Pygmalion Effect): This effect centers on how much a teacher's expectations can drive performance higher among students. Shortened design with no discussion of teacher expectations controlled for any potential Rosenthal effect, particularly since this effect is predominantly individual based (Szumski & Karwowski, 2019). This was not considered a significant threat to this study.

External Validity

Creswell (2018) also describes threats to external validity, which is when a researcher draws incorrect inferences from the results:

Interaction of Selection and Treatment: This threat goes to generalizability. Researchers cannot make claims or infer anything beyond the characteristics of the sample. This is particularly true in a quasi-experimental study lacking random selection. Results would only be applicable to individuals with similar characteristics to participants.

Interaction of Setting and Treatment: This study is not extensible beyond similar settings, as in other introductory computer science classes at the secondary level in similar schools. Here again results can only be extended to similar settings, as in other high schools with similar computer science programs.

Interaction of History and Treatment: Generalization beyond the specific time frame of this study is not warranted, past or future. Further study is called for and should be performed to recreate and duplicate the results from this study to provide further evidence of any results. Transparency in reporting results guards against *any* of these external threats (2018). There are several ways that additional research could be of benefit following this study.

Treatment

Treatment for this study used Revolution J.D. humanoid robots to program the three programming constructs that make up the Böhm-Jacopini structured programming theorem: Sequencing, selection, and iteration. The programming tasks for this treatment were identical in nature to the programming tasks for control, the only difference being the use of actual robots to perform program functionality.

Identical subject material for treatment and control is discussed below. Both used block-language software to create programs that perform sequences of actions, selection of actions, and iteration of actions. Determination of which group is control or treatment was random. The difference in the treatment group is that actions programmed were performed by an actual Revolution: JD humanoid robot in the lab, as opposed to a character sprite moving on the screen in Scratch. Lab assistants guided student operation of robots to further isolate researcher involvement. The control group did not see the robots or equipment. Robots were secured in a cabinet out of sight for the duration of their lab time across the entire study.

Pedagogy

Subject matter used in this study is known as the structured programming theorem, otherwise known as the Böhm-Jacopini theorem. It was developed in 1966 by two computer science theorists who identified three types of programming statements which they proved collectively to be a *Turing equivalent*. These constructs are sequencing, selection, and iteration. Sequencing means doing specific steps in a particular order. Selection means branching program flow based on a conditional state, and iteration is looping for some finite amount of time. By demonstrating that these three statement types collectively form a Turing equivalent, Böhm and Jacopini proved that *any* program can be written using nothing more than these three statement types (Dasgupta, 2016). The Böhm-Jacopini theorem is taught as what is commonly called the 'three pillars of computer science' in computer science classrooms around the world today.

Sequencing, selection, and iteration can be adequately covered at the introductory computer science level across three weeks of content. This has been the typical length of time the Böhm-Jacopini theorem is covered within a unit of Scratch Programming. Robots selected for this study can be programmed in a similar, modified visual block language called Blockly.

Selection of the Böhm-Jacopini theorem was made because it is the fundamental structured programming tenet that renders program *goto* statements unnecessary for programs, making the use of sequencing, selection, and iteration a theoretically *complete* set of programming control structures (Ledgard & Marcotty, 1975). This is applicable to any programming language. It is also conveniently divisible into short practical lessons.

Further, sequencing, selection and iteration can be performed easily in scratch, and easily on the selected robots. These programming steps provided simple points of achievement in both control and experimental groups. This allows a true side-by-side comparison between the treatment and the control scenarios. The control group moved a sprite around the screen, and the treatment group moved a live robot in the physical world.

The advantage of this treatment lies in its simplicity. The programming pedagogy in use is time-tested and represents common fundamental concepts taught in introductory computer programming courses. The programming assignments were identical for both the treatment and control groups. The robots selected were fully functional and previously tested for operations. From the beginning of the treatment period, there was a far less chance of mechanical interference or mechanical delays with the treatment. None of the robots malfunctioned during the study, despite heavy use by students in the treatment group.

The disadvantage with this treatment is that it is limited to the researcher's school, and randomized selection is not possible. These robots are the first set made available to the researcher to determine their applicability for use across the researcher's county. Nevertheless, design characteristics were established to minimize and mitigate threats to validity.

Participants

The population for this study was any secondary students enrolled in an introductory computer science course. The samples were purposefully and conveniently selected secondary computer science students from just such a course. The study sought to determine the ability of robotics to attract and retain computer science students. The researcher has access to students who are clearly in the population to be studied, and the researcher's school has two computer science pathways in which students can achieve computer science education.

School Profile

The researcher teaches at a suburban Atlanta area high school in Georgia. The school has an enrollment of 2,142 students in grades 9-12. Minority enrollment is 44 percent. Asian students are the largest minority at 26 percent. African American students are nine percent and Hispanic students are six. Three percent cover all other categories. Diversity score for the school is 0.61, slightly below the mean of 0.69 for Georgia (Profile, 2020).

The school is ranked in the top five percent of Georgia high schools, as measured by proficiency in both mathematics and reading/language arts. The graduation rate is also in the top five percent of Georgia. There is a 1:20 teacher to student ratio for the school.

Sample Recruiting

The sample for this study was a convenience sample of two periods (sections) of the researcher's Introduction to Digital Technology (IDT) class. This class is the first of three classes in both pathways: *Computer Science* and the *Internet of Things*. Convenience sampling is often called *opportunity sampling* and involves using nearby individuals to serve as participants because of their availability – which is often the case involving students when the teacher acts as the researcher. This is a common sampling technique for education research despite the limitations on generalization to a wider population (Cohen et al., 2007).

Students either select this course coming from their middle schools, or they are placed in the course by administrative scheduling personnel or counselors. The researcher does not control who is scheduled into these classes or to which section they are scheduled. Students have online and school tour information while completing middle school, from which they make their preferences known. During school tours, students are introduced to the variety of electives for which they can sign up, as part of the normal transition to high school each year.

Introduction to Digital Technology as selected, helps remove researcher influence on the sample. Some students enroll in the class, others are enrolled by counselors. There is no input from the researcher. Unlike AP follow-on courses, no recommendation is required from the researcher, nor are there any prerequisites.

Recruiting for this study involved a process for removing each participants identity from demographics collected, including gender, status for free or reduced fee lunches, grade level, age, and assigned section (Period 1 or 3 of the school day). A school administrator provided name separation and assignment of randomized participant numbers to each student, and any identifiable information as to matching identities with data remained outside the researcher's purview. Students used their participant number on all instrument collections. A separate department faculty member retained the ability to remind a student of their participant number outside the purview of the researcher.

Recruited students were not told whether they were in the treatment or control group. They were all informed that regardless of their group, they would all receive the same instruction by the end of the semester after the study has been concluded. Students were instructed to not share their participant number with anyone else or the researcher. They were advised as to the faculty member who was able to help remind them of their number if needed.

All students in both groups were given permission forms to be filled out by their parents or guardians unless they were of a legal age in which they completed the form themselves. All forms were collected and verified prior to the beginning of any research.

Pilot Study Recruiting

Pilot study recruiting was from a similar, nearby county school, using students from the same Introduction to Digital Programming class. The instrument was made available electronically to the school's computer science teacher to administer. Students accessed the instrument via an electronic quiz delivery website making it easier to take and easier to randomize presentation order electronically. Anonymous results were immediately available to the researcher.

Sample Size

The researcher's classes are the only ones in the county afforded robotics equipment for this study. Introduction to Digital Technology (IDT) is taught with widely divergent approaches across the county, despite common Georgia Standards. There exist significant training variations among teachers, precluding expansion to other similar schools. Further, Georgia caps 33 students per section. The researcher had two available IDT classes for study, one for treatment, the other as control. This was a significant challenge because of the smaller sample size.

Lack of Randomization

Randomization of students to any class was not possible. Students coming from middle schools were assigned by counselors based on factors outside researcher control. Fortunately, there is a far broader student-body inclusion for this course because no recommendations or prerequisites exist. It was as close to randomization as was available. After discussion with leadership, the researcher concluded that randomly assignment was not feasible.

Power

Power was clearly affected by study's design. As a single-site, single-effect study, blocking nor stratification were available to help add power. Power is tied to individual students. Power, as the probability of detecting significance of intervention, was hindered by the sample size (Song & Herman, 2010). This study used a standard alpha of 0.05 as the Type I error rate (2-tailed).

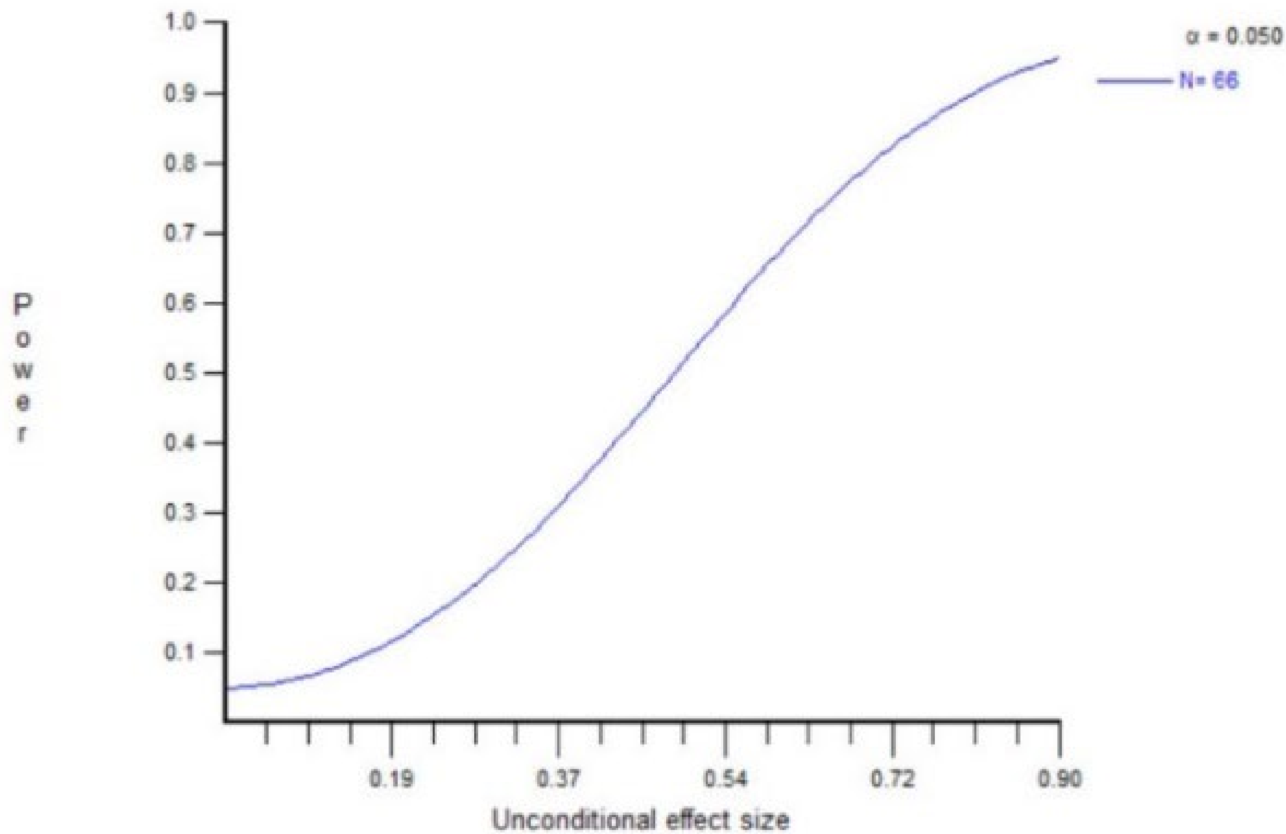
The larger the effect of a treatment, the easier it is to detect (higher power). Conventional practice uses 80% power which gives a reasonable chance of detecting significance at the .05

alpha. This also helps preclude a very large sample size (Cohen, 1988). Power discussion is then to the minimum detectable effect size and required sample size. Assuming an n=66 as the total of the two available sections, Figure 3 shows Power vs. Uncontrolled Effect Size power analysis as generated by a widely recognized program called Optimal Design (Spybrook, 2005).

This model assumes random selection, which is not possible, creating more need for design mediation.

Figure 3

Power Analysis



Controlling for Lack of Randomization

In a quasi-experimental study, the lack of randomization must be controlled through careful study design. Shadish et al. (2002) make several important design suggestions to improve power, reliability, and validity. Their suggestions were integrated within the design of this study. First, the study made use of their double pretest recommendation, which enables the uncovering of biases which may occur between pretests, leading to an understanding of their possible presence in posttest analysis. Second, there were three post-tests (repeated measures), which according to the authors allows examination of patterns of evidence within resulting effects. Because the design introduces known programming constructs each successive week of the study, there existed strong prior knowledge about expected patterns. This according to the authors, allows more confidence in causal analysis. Third, the use of ANCOVA could allow determination as to any lurking influences in data that might be attenuating results. In addition to these constraints, instrumentation design was carefully devised to support internal, external, face and construct validities.

Instrumentation

This study used of a double pre-test, repeated-measures instrument designed by the researcher, derived from the instruments provided by Dr. Lent. An initial comprehensive review of literature was used, beginning with the Computer Programming Self-Efficacy Scale (CPSES) list of questions (Tsai et al., 2019), and the survey design of Blouin (2011). The CPSES questions provided strong examples of simple single-focus questions but were too diverse in scope for use in this study. Blouin's survey questions were also reviewed, which provided more examples of properly formed questions, but here again the focus of the questions was on general computer use. This study needed to focus specifically on self-efficacy and interest of programming sequencing, selection, and iteration in visual programming languages. Additionally, the Motivational Strategies for Learning Questionnaire (MSLQ) was reviewed to determine if any of its questions could be of utility for this study (Pintrich & de Groot, 1990). Here again questions were properly constructed single-scope questions that investigate self-reflective opinions of learning and motivation but were not sufficiently within the domain of questioning needed for this study. For this reason, it was concluded that a specific instrument should be created.

The Social Cognitive Career Theory developed by Robert Lent and Steven Brown is a well-studied and tested theoretical model that is domain specific. As this is the theoretical model selected for this study, any instrumentation must be conceived as a domain-specific branch of what has already been tried and tested with support in the literature. As such, the researcher was able to contact Dr. Lent himself, at the University of Maryland. Dr. Lent graciously forwarded two of their actual instruments, along with their instrumentation guide (Lent & Brown, 2006). Their paper guided all instrument creation for this study. The instruments received are included in Appendix A and B. The Engineering instrument (Appendix A) proved reliable with a cited Cronbach's Alpha from the original instrument ($\alpha = .89$). They did an 8-week test-retest correlation ($r=.89$), and it measured with a theory consistent set of measures of choice, performance, and persistence from their 1984 and 1986 studies (Lent et al., 2003). The Computer Science Instrument (Appendix B) was used in prior studies with a coefficient alpha estimate of .89 for persistence questions and .89 for coping efficacy. The study reflecting the instrument itself had alpha estimates of .88 for milestones and .87 for coping efficacy scales (Lent et al., 2008). These instruments are reliable templates for creating a domain-specific instrument for this study because they have strong validity, and support SCCT theory itself. Dr. Lent provided the researcher with exemplars and the guide with which to use them.

Instrument Item Revision Plan

The instrument derived from these exemplars was validity tested and reviewed by an expert peer-review panel. Experienced computer science educators helped shape the instrument. They evaluated questions submitted by the researcher, offered their own suggestions, rejected questions, modified them, and determined which questions represented the best set to properly measure the research questions of this study. All questions were formatted with Likert type scales to code them numerically. The two measurement variables were interest and self-efficacy. All questions were domain specific to computer science education: Learning sequencing, selection and iteration, and interests in remaining in the computer science pathways for their high school careers. Multiple questions tested each variable, and scores were totaled for each variable in analysis.

After an agreed upon and sufficiently large set of questions was developed, the instrument was pilot tested on a similar class at two other schools in the researcher's county. The results from this pilot were analyzed for reliability using Cronbach's alpha. Additional variation analysis was performed on the dual pretests.

Construct Validity

Construct validity is concerned with how well a measurement properly reflects a theoretical construct that is being measured. The difficulty comes from the fact that there is not a direct measurement of a theoretical construct (Andrews, 1984). Factor analysis provides researchers with a diagnostic tool that evaluates data connected to the theoretical model(s) being studied, or the target constructs of the study themselves (Zeynivandnezhad, 2019).

To demonstrate construct validity, results from the pilot test underwent factor analysis to determine if interest and self-efficacy factors are represented by loading appropriately based on instrument questions. Since interest and self-efficacy under the SCCT model are not independent, oblique rotation (oblimin) was used in the factor analysis seeking a determinant above 0.0001 so there is confidence in a lack of multicollinearity, and Keiser-Meyer-Olkin (KMO) value of at least 0.5 for confidence that the instrument's proportion of variance is low enough for factor analysis. Bartlett's test of sphericity was also used to verify statistical significance so data is separated from the identity matrix hypothesis of the test which would indicate variables too poorly suited for factor analysis (Field, 2018). Only two factors were expected to be identified in the results. Principle Components Analysis was the analysis method used as the factor analysis test for the instrument.

Statistical Model and Assumptions

Quasi-Experiments can often use parametric tests providing groups are normally distributed and there is homogeneity of variance. ANOVA can compare control and treatment groups for the Independent Variable (IVs) which are the robotics treatment and gender. It can also be used to test socioeconomic differences. Essential assumptions include Independence of Groups, Continuous Dependent Variable (DV), Normal distribution, Homogeneity of Variance.

However, as a quasi-experimental design, ANCOVA would have likely proven relevant to this study, by helping explain confounding covariances and remove their effects. ANCOVA can explain within-group variance, essentially taking unknown or unexplained variances from ANOVA and explaining their confounding variables. The cost is degrees of freedom (Creswell, 2018). Essential assumptions of ANCOVA include residuals that are: normally distributed, have a mean of zero, are independent, and as with many other statistics, have homogeneity of variance. ANCOVA is considered robust against minor violations of assumptions and is often used in quasi-experiments in lieu of matching (Shadish et al., 2002).

Pilot Study for Instrument Reliability & Construct Validity

Reliability for this instrument meant that each question means the same thing to each respondent. This involves ensuring that each question is understood by all respondents and wording is clear. It means that questions ask a single subject question, and response categories are unambiguous with a scale that is properly understood. The key was to provide unambiguous, understandable questions that provided consistent measures across respondents (Fowler, 2014).

To verify reliability, a pilot test was performed at two similar high schools within the same county, in similar classrooms of students enrolled in the same introductory programming course. Prior permission and coordination were performed to ensure minimal impact on the school and class used for the pilot study.

Cronbach's alpha was used to rate the reliability of the survey instrument after collection of instrument pilot-study responses. Cronbach's alpha is used to test internal consistency (Pelham & Blanton, 2007). Pilot study data was used for factor analysis to verify construct validity of the instrument.

Instrument Design Considerations

There is a well-established body of literature on the various effects that the order of questions can have on a survey's results, as in (Huang, 2016), (Lasorsa, 2003), and (Schwartz, 1999). Two specific experiments were performed by Strack et al. (1988) that specifically dealt with the priming effect of survey question order. To remove these potentially bias effects, sources are in open agreement that randomizing the order of presentation is one of the best solutions. As such, and because of the repeated-measure design of this study, the surveys were given electronically using software that automatically randomizes the order of the questions for the survey.

Instrument error can also be introduced when respondents to the instrument grow fatigued, are bored and no longer provide truthful answers, or do not provide any answers at all, as when they quit completing the instrument. They might also remember previous responses they gave and respond with the same answers if they have memorized the patterns of the survey. Specifically, results can be affected more by those with less education who answer a survey quickly. The survey can also teach the respondents what to expect, which can affect their responses (Egleston et al., 2011). Here again, randomization mitigated the bias potentialities of the instrument.

These questionnaires were electronic forms that were filled out by each student. The only categorizations collected for the study were gender, age, and grade level and the qualification for free or reduced fee lunches for socioeconomic status. These metrics were completely disassociated from students, who were given random identification numbers for survey response linkage. The researcher did not participate in identity blinding.

Prior to the pilot or actual study, proper permissions were obtained from parents or of-age respondent students as required. Piloting the instrument provide solid verification of reliability and an ability to uncover any confusion or instrument misunderstanding (Mertler, 2019). Initial construct validity analysis was performed on the pilot results to determine if further question modification was needed.

Procedure

Institutional Review Board Approval

For this study, minor students were completing surveys, however Institutional Review Board (IRB) exemptions from human research applied. Using the University of Georgia Human Research Protection Program instruction HRP 103 Investigator's Manual, the researcher provided this prospectus and all required forms to the IRB for research approval. Permissions forms were derived directly from IRB exemplars. No research commenced, including instrument pilot studies, until IRB approvals (waiver) had been granted. There was no financial support for this study, so there was no reporting for any financial compensations to provide. This study did not operate under any government or Department of Defense grants and was not clinical in nature.

The researcher completed required IRB training as part of the graduate coursework for the PhD program and was current for the entirety of the study period. Specific researcher training modules completed: Social & Behavioral Research – Children & International, Social & Behavioral Research, Conflict of Interest, and Export Compliance (unrelated to this study). The researcher coordinated with the senior committee chair, all IRB submissions for approval prior to submission to the IRB.

Participant Permissions

Students were provided consent forms for parental signature, or student signature if they were of age. These forms have been retained by the researcher as part of the records for this study. They were collected and verified before any data collection occurred.

Information Security and Privacy

Student identity was removed from all records used in the research, and students were given a randomized number with which to complete any surveys, so that the collected gender and socioeconomic status was able to be matched with the correct anonymized student. All IRB regulations for record storage and security, as well as state and county regulations for information security was followed.

Pilot Study Data Collection

Arrangements with a similar high school in the researcher's county were made to pilot the instrument. Surveys were filled out in a single class period, automatically reported via the survey software, and delivered to the researcher electronically for instrument analysis. From those results the peer-review committee had the chance to make any final instrument modifications necessary. A third school was contacted to perform the same operations to increase the available sample size for the pilot study. Factor analysis was performed to ascertain construct validity and determine if duplicates needed to be removed.

Study Procedures

One week prior to the study, the first survey was given to both experimental and control classes. The following week, on the first day of the study, the classes again completed a second round of the pretest. Students then received instructions for establishing their accounts in the ARC Integrated Development Environment (IDE) for Robot-Scratch (treatment) and the Scratch IDE (control)

The first week students received instruction on sequence programming. This programming in Blockly (treatment) or Scratch (control) was set to require less than 20 minutes to complete per program, so that there was sufficient time for each student to run their program on an actual robot, or in Scratch for the control group. The second week students received instruction on selection programming. Programming in Blockly (experiment) or Scratch (control) was again set to require less than 20 minutes per program, so there was sufficient time for each student to run their program on an actual robot, or in Scratch for the control group.

The third week students received instruction on iteration (looping) programming. This programming in Blockly (experiment) or Scratch (control) was again set to require less than 20 minutes to complete per program, allowing sufficient time for each student to run their program on an actual robot, or in Scratch for the control group. The final day of each week, students in both experimental and control groups took the post-test instrument. There was sufficient time to accomplish this each Friday.

Timeline of Events

Table 2 represents a timeline of events for the study. There was an additional week to account for any unanticipated delays for the commencement of the study, so that any portion or even an entire week delay could be absorbed without significantly impacting the schedule. This extra time was not needed. Posttests were given on Fridays so that there was no data impact from recollection loss over a weekend.

Table 2

Study Timeline Summary

Event	Target Dates
IRB Approval	Spring 2021
Study Permissions	Week of August 9th
Instrument Pilot Test	August 16th
Instrument Analysis	Week of August 16th
Pretest 1	August 23rd
Pretest 2	August 30th
Sequencing*	August 30th - September 3rd
Selection*	September 6th – September 10th
Iteration*	September 13th – September 17th
Over-Run Week	September 20th – September 24th

* Post-week survey instrument given each Friday.

Data Analysis

Statistical Software

IBM's SPSS software was be used for the statistical analysis of this study's results. This software is available under license for students at the University of Georgia.

Study Variables

The independent variables for this study were the experimental treatment for robotics, gender, and socio-economic status as defined by eligibility for free or reduced fee lunches. The treatment vs. control used a summative score of responses across the two continuous dependent variables of *self-efficacy* and *interest*. Gender and socio-economic status are both categorical (nominal) variables. For each variable, ANOVA statistics was used to compare scores.

Descriptive statistics were used to analyze each variable because they represent the most fundamental method of describing a set of observations (Pelham & Blanton, 2007). Specifically, measures of central tendency (mean, median, mode), standard deviation and variances of data are presented. The data was also analyzed to determine skewness or kurtosis of each item, as well as testing for normalcy. This provides a solid understanding of pre-treatment conditions. Graphical representations helping clarify the data are also provided. These same descriptive statistics are used to describe post-test responses.

It is important due to the quasi-experimental nature of this design to also perform t-test analysis of the pretest scores between the experimental and control groups to determine if they are statistically similar. This helps support the internal validity of this study. Groups should be as similar as possible. Additionally, variance analysis between each of the two pretests sought to find any possible maturation or other hidden variances. Following the study, analysis of covariance (ANCOVA) was slated to be performed on all significant results to see if there remained any hidden covariances affecting results - and to account for them.

It is important to understand the effect size of findings. Cohen's *d* would be used to guide interpretations of the practical effect size of any findings. The small (0.2), medium (0.5) and large (0.8) effect size recommendations are seen as just that – recommendations. Effect sizes can be contextually interpreted in addition to any statistically significant findings.

For all these tests, an alpha level of .05 was used as it is a recognized part of widespread Null Hypothesis Significance Testing (NHST) as an acceptable setting to reduce the chances of a Type I error. Type I errors involve making the wrong decision when the null hypothesis is true, which is belief that there was an effect when none existed (Field, 2018). For this study, if the null hypothesis could be rejected with an alpha of .05, it would be reported as statistically significant.

Type II errors are when a null hypothesis is not rejected when it should be. This would mean an incorrect conclusion that there was no difference between the treatment and control when an actual difference existed.

Avoiding Group Dynamics

Even though it is widely recognized that professional programmers and software developers typically work in teams, each student performed their assignments individually on the robots, even if coordination of robot use is required due to a limited number of the robots. This supports internal validity because control-group students worked on their coding individually as well. Placing students into teams or groups could introduce potential group-dynamic affects that would not be controlled. As such individual performance was used to prevent any potential confounding affects. Students took turns programming and using the robots.

Data Analysis and Reflection

Analysis of covariance would be used to help support any statistically significant results of the study by helping determine if any confounders or lurking variables exist that need to be accounted for within results. This would allow control for any covariates within results. It is important to be able to find any attenuation to provide confidence in results.

Data Analysis

Table 3 summarizes study variables and how they were analyzed. Statistical analysis methods are shown for each research question to understand their application. Socio-economic status was collected for analysis as a potential covariate. These data results are presented in charts and tables in such a manner as to convey the most accurate and applicable meaning to the findings of the study. All data and records are retained per IRB requirements for preservation.

Table 3

Variables and Data Analysis

Research Question	IV	DV	Analysis
How does the use of robotics in programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?	Treatment	Self-Efficacy *Continuous*	Descriptive Statistics Mean, Median, Mode, Std. Dev., Variance, Range, Sample Distribution ANOVA/ANCOVA
How does the use of robotics in programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?	Treatment	Interest *Continuous*	Descriptive Statistics Mean, Median, Mode, Std. Dev., Variance, Range, Sample Distribution ANOVA/ANCOVA
Is there a difference between genders in how the use of robotics programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?	Gender	Self-Efficacy *Continuous*	Descriptive Statistics Mean, Median, Mode, Std. Dev., Variance, Range, Sample Distribution ANOVA/ANCOVA

Research Question	IV	DV	Analysis
Is there a difference between genders in how the use of robotics programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?	Gender	Interest *Continuous*	Descriptive Statistics Mean, Median, Mode, Std. Dev., Variance, Range, Sample Distribution ANOVA/ANCOVA

Assumptions of statistics

Table 4 summarizes the assumptions of statistics used in this study (Field, 2018).

Table 4

Statistics and Assumptions

Statistic	Assumptions
ANOVA	Continuous Dependent Variable (DV) Two categorical between-subject variables One categorical within-subjects variable No significant outliers in every cell of design DV approximately normally distributed Homogeneity of variance Sphericity is met
ANCOVA	Linear relation between Independent Variable (IV) and DV Homogeneity of regression slopes Covariate is independent of treatment effects (No interaction between the covariate and the IV)
Factor Analysis	For collection of observed variables, there exist underlying variables (factors) that explain relationships between the variables. Used to analyze structural validity of the survey.

Rationale for Variable Effect

This study used a design that should show a self-efficacy increase in each of the three weeks of the study period corresponding to the portion of the Böhm-Jacopini theorem introduced. Since skills are developed in specific increments, as each of these skills is mastered through specific focus, student confidence in their ability to use each construct should see an increase. Skills build sequentially and are naturally included within the next phases of training.

Self-efficacy should therefore continue to improve across the length of the study. Deviations would indicate struggle with a particular construct.

Interest was expected to be related both to the development of successful programming, as self-efficacy impacts interest (Lent et al., 2008), and the novelty of robotics. Despite mixed review of robotics use across the literature, these newer ready-to-use humanoid robots provided a novel approach with which to study their impact, particularly found in their ready-to-use setup.

CHAPTER 4

Results and Findings

This double pre-test, repeated measures quasi-experimental study sought to determine the influence of robotics on motivations of secondary students to complete a computer science pathway in high school. Participants consisted of high school students enrolled in an Introduction to Digital Technology (IDT) class at a suburban high school in the southeastern United States. The data collection instrument was a student survey taken twice before and three times across a three-week study period. Treatment used Revolution: JD humanoid robots programmed in the Blockly language. Observation group programming occurred in the Scratch programming language.

This chapter presents the analysis of the data and findings towards the objectives of the research. Data was collected and analyzed to understand the effects of interest and self-efficacy when a treatment of robotics is introduced in lieu of traditional software programming to teach the fundamentals of computer programming to new computer science students at the secondary level. The independent variables used were the study group (treatment vs. observation), and gender. The dependent variables were interest and self-efficacy, and an alpha level of .05 was used to determine the effect of each independent variable on the dependent variables. Data analysis included Analysis of Variance (ANOVA) statistics.

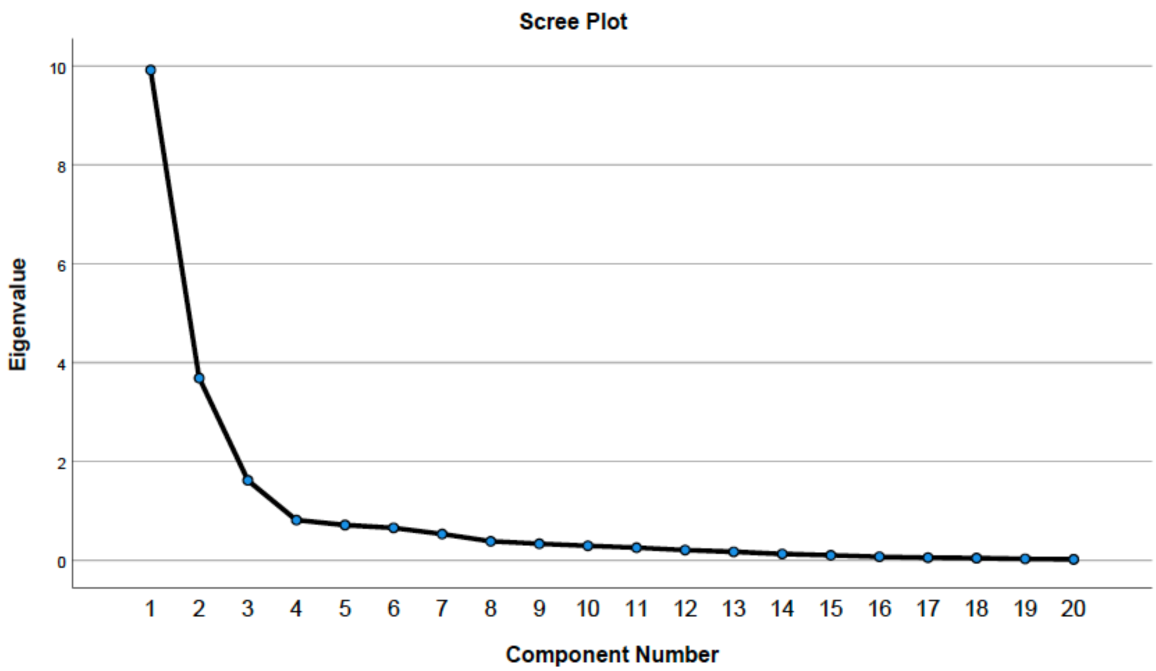
Instrumentation

Appendix C contains the research questions that made up the instrument used in this study. The first eight questions were scored using a 10-point Likert scale. Together with the next four questions scored on a 7-point Likert scale, they were used to measure self-efficacy. Eight questions scored on a 5-point Likert scale were used to measure interest. Scores for each factor measured were summed for each variable.

The instrument was piloted by two similar high schools in the same county as the study school, with a total of 40 responses between those schools. Despite a small sample size, an exploratory analysis revealed two factor loadings with an Eigenvalues significantly above 1, and a third marginally above, at 1.62. To verify the exclusion of the third factor, a parallel analysis was performed using the Monte Carlo PCA for Parallel Analysis program with a run of 100 simulations. A third factor loading of 1.82 was revealed above the 1.62 value from the factorial analysis. This supported rejection of the third Eigenvalue, leaving two factors for final consideration as the result. A two-factor extraction was then performed. The Kaiser-Meyer-Olkin test for sampling accuracy was .819 indicating a level adequate for analysis. Bartlett's $\chi^2(190) = 827.92$ ($p < .001$) indicated a correlation structure adequate for factor analysis. The two factors accounted for 68.02 percent of the variation. Figure 4 presents the scree plot from the factor analysis. This plot visualizes that the instrument did, in fact, load to two factors, indicating support for the design which centered on two variables: Interest and Self-Efficacy. Components 1 and 2 were clearly breakaway values in the chart, Eigenvalues well above 1.0.

Figure 4

Scree Plot for Instrumentation



Additional Constraints

Only two students were eligible for free or reduced fee lunches, making the sample size insufficient to perform any covariance analysis for that influence.

Additionally, this study occurred during the COVID-19 pandemic. Fortunately, the study was conducted with all students in person, both groups being present face-to-face at school for the entire duration of the study. Three students from the observation group and one from the control group, were lost to COVID-19 protocols that sent them to home isolation. These students were removed from class based on established contact tracing and other medical considerations outside the control of the researcher. Although they were removed from the study, they were allowed to follow the learning remotely.

Double Pretest Considerations

Statistical analysis for each question was performed using the first pretest, the second pretest, and the average of both pretests, to determine if there were any effects observable that might affect results. Results are provided for each run of the data. Tables 5 through 8 show the biographical makeup of both control and observation groups.

Table 5

Participant Gender

Group	Boys	Girls	Free/Reduced Lunches
Control	26	5	2
Observation	21	8	0

Table 6

Participant Age

Group	13	14	15	16	17
Control	0	15	8	7	1
Observation	5	12	8	3	1

Table 7

Participant Grade

Group	9	10	11	12
Control	17	10	3	1
Observation	16	7	5	1

Table 8

Participant Race

Group	White	Asian	Black	Hispanic
Control	12	15	2	2
Observation	15	9	1	4

Self-Efficacy Measurements

Self-Efficacy Descriptive Statistics

Table 9 presents descriptive statistics for all measurements including the double pretest and three weekly measurements for sequencing, selection, and iteration for the control group. This table shows relative stability between pretests and increases in measures of central tendency as the study progressed. These descriptive statistics also show relatively stable data standard deviations with values that tightened somewhat at the end of the study but remained consistent even across pretests.

Table 9

Control Group Self-Efficacy Descriptive Statistics

Control Group	Mean	Median	SD	Min	Max
Pretest 1	62.03	61.00	16.42	28	91
Pretest 2	64.42	65.00	16.46	29	89
Week 1 Sequencing	72.29	73.00	16.09	29	95
Week 2 Selection	79.48	82.00	16.79	33	103
Week 3 Iteration	90.81	94.00	11.73	57	107

n = 31

Table 10 provides the same for the observation group. Histogram analysis indicated slight left skewing on some of measurements, however most were closer to normal.

Table 10

Observation Group Self-Efficacy Descriptive Statistics

Observation Group	Mean	Median	SD	Min	Max
Pretest 1	56.00	53.00	19.33	20	97
Pretest 2	59.34	59.00	20.51	19	97
Week 1 Sequencing	71.24	67.00	18.04	40	100
Week 2 Selection	78.69	80.00	15.15	47	105
Week 3 Iteration	91.52	95.00	17.90	23	108

n = 29

Testing Assumptions for ANOVA

Schmider et al. (2010) make the case that not enough researchers test their statistics for violations of assumptions. This study tested assumptions for both interest and self-efficacy to be able to understand the effects on the results might be, and to understand how they might attenuate findings. The first assumption is that the dependent variable, self-efficacy is measured at the continuous level, which was met. The second assumption is that the two between-subject members (study group and gender) are categorical, which is also met. Third, there should be one within-subject variable which was time. Results did not show violations until the fourth assumption, which seeks to determine that there are no outliers found for any cell of the design. Outliers were identified based on the three pretest conditions considered.

Table 11 presents the outliers found for self-efficacy across both observation and control. Where outliers were found, the participant ID is presented to show where a specific student may have been a repeat or consistent outlier. This only occurred one time for self-efficacy.

Table 11

Self-Efficacy Outliers

Measurement	Cell	Outliers	Participant ID
Pretest 1	Control/Male	0	
Pretest 2	Control/Male	0	
Averaging Pretests	Control/Male	0	
W1/Sequencing	Control/Male	1	4115
W2/Selection	Control/Male	0	
W3/Iteration	Control/Male	1	4060
Pretest 1	Control/Female	2	4100, 4035
Pretest 2	Control/Female	0	4055
Averaging Pretests	Control/Female	1	4100
W1/Sequencing	Control/Female	0	
W2/Selection	Control/Female	0	
W3/Iteration	Control/Female	0	
Pretest 1	Treatment/Male	0	
Pretest 2	Treatment/Male	0	
Averaging Pretests	Treatment/Male	0	

Measurement	Cell	Outliers	Participant ID
W1/Sequencing	Treatment/Male	0	
W2/Selection	Treatment/Male	0	
W3/Iteration	Treatment/Male	0	
Pretest 1	Treatment/Female	0	
Pretest 2	Treatment/Female	0	
Averaging Pretests	Treatment/Female	0	
W1/Sequencing	Treatment/Female	0	
W2/Selection	Treatment/Female	0	
W3/Iteration	Treatment/Female	2	4230, 4335

Note. Bold Participant ID denotes extreme value

None of the outliers were excluded from analysis because of the small sample size, and the decision that these classes were an accurate representation of normal class participations in the experience of the researcher.

The fifth assumption seeks normality of dependent variables for each cell of the design. A Shapiro-Wilk test was performed for all three pretest conditions used (Pretest 1, Pretest 2, and average of pretests), and all three weeks of the study, for a total of 6 normality tests. Table 9 shows the violations recorded for each cell. All other tests were not violated.

Table 12

Normality Violations for Self-Efficacy

Group	Measure	Shapiro-Wilk
Male/Control	Week 3 Iteration	$W(26) = 0.91$, $p = 0.03$
Female/Control	Pretests Averages	$W(5) = 0.71$, $p = 0.01$
Male/Observation	Week 3 Iteration	$W(21) = 0.82$, $p < .01$
Female/Observation	Week 3 Iteration	$W(8) = 0.65$, $p < .01$

Understanding these violations, the decision was made to proceed understanding the small sample sizes of the two groups and the relative robustness of the ANOVA statistics against this specific violation (Cohen et al., 2007).

Assumption six tests that the dependent variable should be equal between groups of the between-subject factors. This homogeneity of variance is tested using Levene's Test of Equality of Error Variances. Assumption seven is the remaining assumption that asserts that the variances of the differences between the groups should be equal. For this assumption Mauchly's Test of Sphericity was used. These assumption results are reported with each ANOVA run.

Self-Efficacy Pretest 1

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 30.75$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.08,168) = .051$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

Self-Efficacy Pretest 2

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 51.83$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.02,168) = 1.12$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

Self-Efficacy Pretest Averages

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 41.65$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.28,168) = .22$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

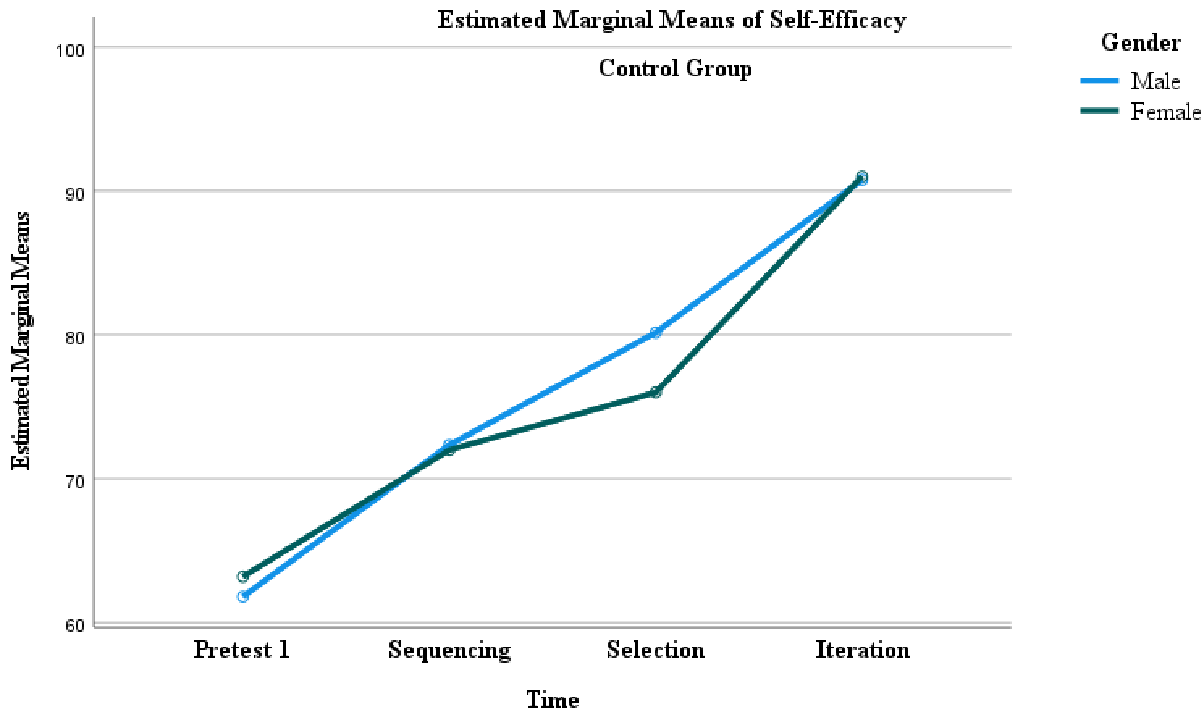
Data Visualization

Figures 5 shows the increases in self-efficacy over time based on gender for the control group. This chart visualizes a slightly more consistent growth increase for boys than girls, however both groups reached similar endpoints by the conclusion. This indicates self-efficacy growth consistent with expectations across the study/training period in the control group.

Figure 5

Control Group Self-Efficacy

Figure 6 shows the increases in self-efficacy over time based on gender for the



observation group. The chart indicates somewhat parallel growth patterns with boys holding slightly higher values throughout. This indicates self-efficacy growth consistent with expectations across the study/training period in the observation group.

Figure 6

Observation Group Self-Efficacy

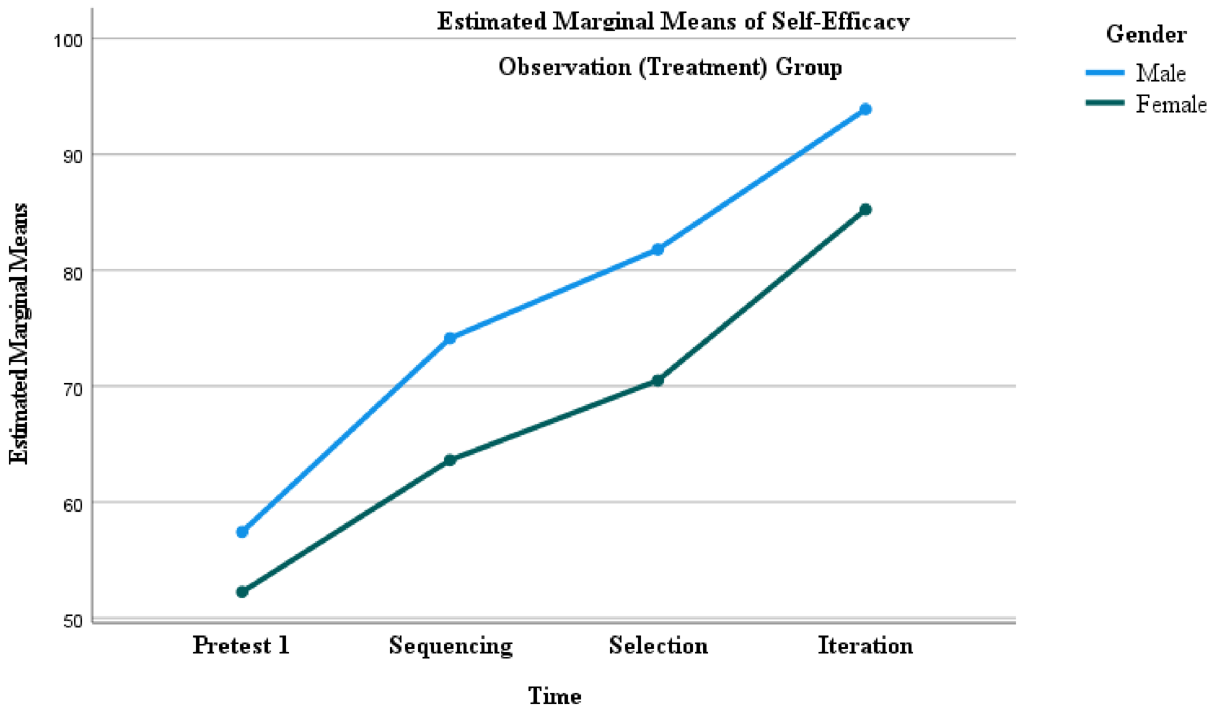


Table 13

Means Analysis for Self-Efficacy

Group	Pretest 1	Pretest 2	Measure 1	Measure 2	Week 3
Control	5.17	5.37	6.02	6.62	7.57
Observation	4.67	4.95	5.94	6.56	7.63

Adjusted Range (1: No Confidence, 9: Complete Confidence)

Table 14

Means Analysis for Self-Efficacy by Gender

Group	Pretest 1	Pretest 2	Measure 1	Measure 2	Week 3
Control Boys	5.15	5.27	6.03	6.68	7.56
Control Girls	5.27	5.87	6.00	6.33	7.58
Observation Boys	4.79	5.29	6.18	6.82	7.83
Observation Girls	4.35	4.05	5.30	5.88	7.10

Adjusted Range (1: No Confidence, 9: Complete Confidence)

Table 15

ANOVA Summary for Self-Efficacy

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Time * Group	Sphericity Assumed	240.123	3	80.041	.618	.604	.011
	Greenhouse-Geisser	240.123	2.081	115.403	.618	.547	.011
Time * Gender	Sphericity Assumed	173.092	3	57.697	.446	.721	.008
	Greenhouse-Geisser	173.092	2.081	83.188	.446	.649	.008
Time * Group * Gender	Sphericity Assumed	19.709	3	6.570	.051	.985	.001
	Greenhouse-Geisser	19.709	2.081	9.472	.051	.955	.001
Error (Time)	Sphericity Assumed	21752.658	168	129.480			
	Greenhouse-Geisser	21752.658	116.521	186.684			

Interest Measurements

Interest Descriptive Statistics

Table 16 presents descriptive statistics for all measurements including the double pretest and the three weekly measurements for sequencing, selection, and iteration for the control group.

Table 16

Control Group Interest Descriptive Statistics

Control Group	Mean	Median	SD	Min	Max
Pretest 1	34.55	37.00	6.43	9	40
Pretest 2	34.34	37.00	7.11	8	40
Week 1 Sequencing	34.55	38.00	6.87	11	40
Week 2 Selection	35.00	37.00	5.95	15	40
Week 3 Iteration	35.34	38.00	5.91	19	40

N=31

Table 17 provides the same for the observation group. Histogram analysis indicated at least some left skewing on most of the measurements. This group also showed fairly stable measures of central tendency with more variation near the beginning of the study. Like the control group, there was some standard deviation tightening by the end.

Table 17

Observation Group Interest Descriptive Statistics

Observation Group	Mean	Median	SD	Min	Max
Pretest 1	31.32	32.00	5.99	14	40
Pretest 2	31.06	32.00	5.99	16	40
Week 1 Sequencing	31.42	32.00	6.40	11	40
Week 2 Selection	32.16	33.00	7.12	8	40
Week 3 Iteration	32.42	32.00	6.32	12	40

N=29

Testing Assumptions for ANOVA

Here again assumptions were tested to understand how results might be mitigated based on any violations observed (Schmider et al., 2010). The first assumption is that the dependent variable, interest, is measured at the continuous level, which was met. The second assumption is that the two between-subject members (study group and gender) are categorical, which is also met. Third, there should be one within-subject variable which was time. Results did not show violations until the fourth assumption, which seeks to determine that there are no outliers found for any cell of the design. Outliers were identified based on the three pretest conditions considered. Table 12 presents the outliers found for interest across both observation and control. For interest, there were three students who were notably consistent outliers.

Table 18

Interest Outliers

Measurement	Cell	Outliers	Participant ID
Pretest 1	Control/Male	1	4110
Pretest 2	Control/Male	2	4020, 4115
Averaging Pretests	Control/Male	0	
W1/Sequencing	Control/Male	1	4115
W2/Selection	Control/Male	1	4115
W3/Iteration	Control/Male	0	
Pretest 1	Control/Female	1	4055
Pretest 2	Control/Female	1	4055
Averaging Pretests	Control/Female	1	4055
W1/Sequencing	Control/Female	1	4055
W2/Selection	Control/Female	2	4100, **4055**
W3/Iteration	Control/Female	1	4055
Pretest 1	Treatment/Male	0	
Pretest 2	Treatment/Male	0	
Averaging Pretests	Treatment/Male	0	
W1/Sequencing	Treatment/Male	0	
W2/Selection	Treatment/Male	0	
W3/Iteration	Treatment/Male	0	
Pretest 1	Treatment/Female	1	**4260**
Pretest 2	Treatment/Female	1	**4260**
Averaging Pretests	Treatment/Female	1	**4260**
W1/Sequencing	Treatment/Female	1	**4260**
W2/Selection	Treatment/Female	1	**4260**
W3/Iteration	Treatment/Female	1	4260

Note. Bold Participant ID denotes extreme value

None of the outliers were excluded from analysis because of the small sample size, and the decision that these classes were an accurate representation of normal class participations in the experience of the researcher.

The fifth assumption seeks normality of dependent variables for each cell of the design. A Shapiro-Wilk test was performed for all three pretest conditions used (Pretest 1, Pretest 2, and average of pretests), and all three weeks of the study, for a total of 6 normality tests. Table 13 shows the violations recorded for each cell. All other tests were not violated.

Table 19

Normality Violations for Interest

Group	Measure	Shapiro-Wilk
Male/Control	Week 2 Selection	W(26) = 0.90, p = .02
Male/Control	Week 3 Iteration	W(26) = 0.92, p = .048
Male/Observation	Pretest 1	W(21) = 0.90, p = .04
Male/Observation	Pretest 2	W(21) = 0.82, p < .01
Male/Observation	Pretest Average	W(21) = 0.87, p = .01
Male/Observation	Week 1 Sequencing	W(21) = 0.79, p < .01
Male/Observation	Week 2 Selection	W(21) = 0.81, p < .01
Male/Observation	Week 3 Iteration	W(21) = 0.78, p < .01
Female/Observation	Pretest 1	W(8) = 0.61, p < .01
Female/Observation	Pretest 2	W(8) = 0.69, p < .01
Female/Observation	Pretest Average	W(8) = 0.66, p < .01
Female/Observation	Week 1 Sequencing	W(8) = 0.66, p < .01
Female/Observation	Week 2 Selection	W(8) = 0.74, p < .01
Female/Observation	Week 3 Iteration	W(8) = 0.74, p < .01

Understanding these violations, the decision was made to proceed understanding the small sample sizes of the two groups. There exists strong support with ANOVA for *some* robustness against violations of normality (Schmider et al., 2010).

Assumption six tests that the dependent variable should be equal between groups of the between-subject factors. This homogeneity of variance is tested using Levene's Test of Equality of Error Variances. Assumption seven is the remaining assumption that asserts that the variances of the differences between the groups should be equal. For this assumption Mauchly's Test of Sphericity was used. These assumption results are reported with each ANOVA run.

Interest Pretest 1

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 30.75$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.20,168) = .310$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

Interest Pretest 2

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 16.74$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.45,168) = 1.965$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

Interest Pretest Averages

Levine's Test did not show any violations, ($p > .05$) for all measures. Assumption six was met. Mauchly's test of sphericity indicates that the assumption of sphericity has been violated, $\chi^2(5) = 41.65$, $p < .01$. Since sphericity is violated, Greenhouse-Geisser adjustments were used.

A three-way mixed ANOVA was run to understand the effects of study group, gender, and time on self-efficacy. The Greenhouse-Geisser adjusted three-way interaction between time, gender and study group was not statistically significant, $F(2.78,168) = 1.01$, $p > .05$. All two-way interactions were not statistically significant ($p > .05$).

Interest Data Visualization

Figures 7 shows increases in self-efficacy over time based on gender in the control group. This indicates interest growth consistent with expectations across the study/training period in the control group.

Figure 7

Control Group Interest

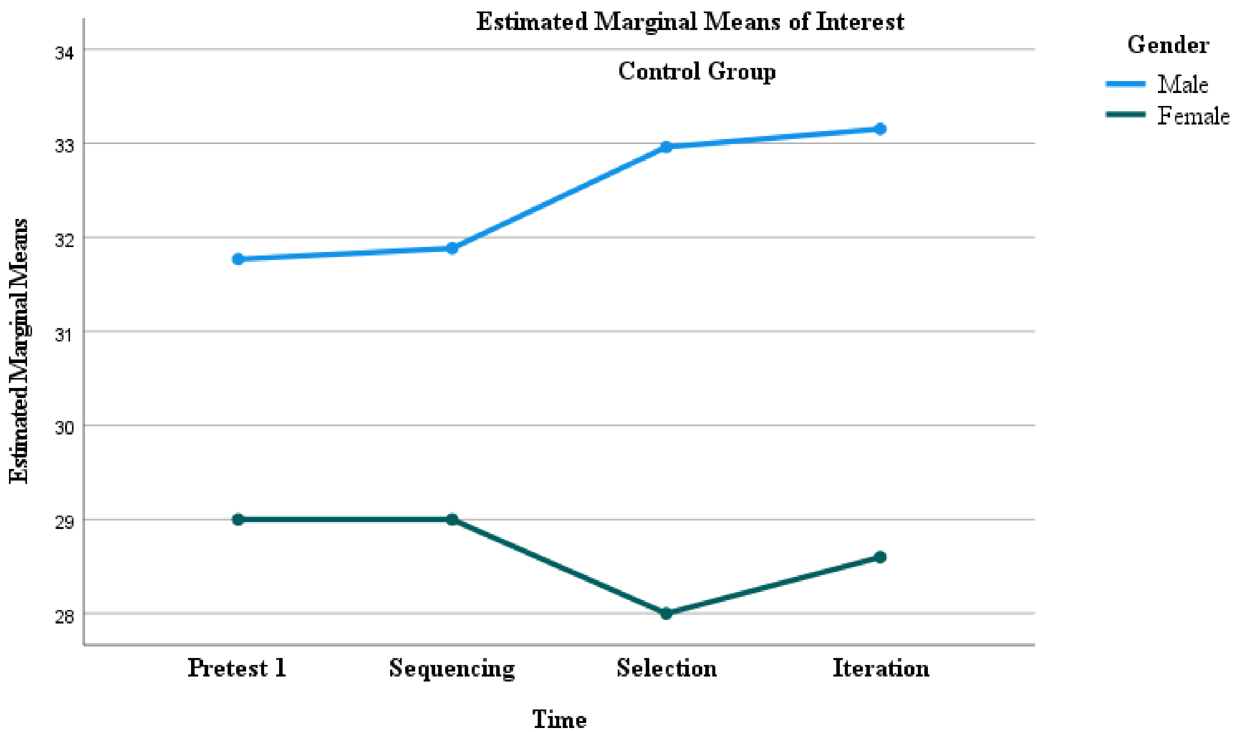


Figure 8 presents the same for the observation group that received the robotics treatment. This indicates interest growth consistent with expectations across the study/training period in the observation group.

Figure 8

Observation Group Interest

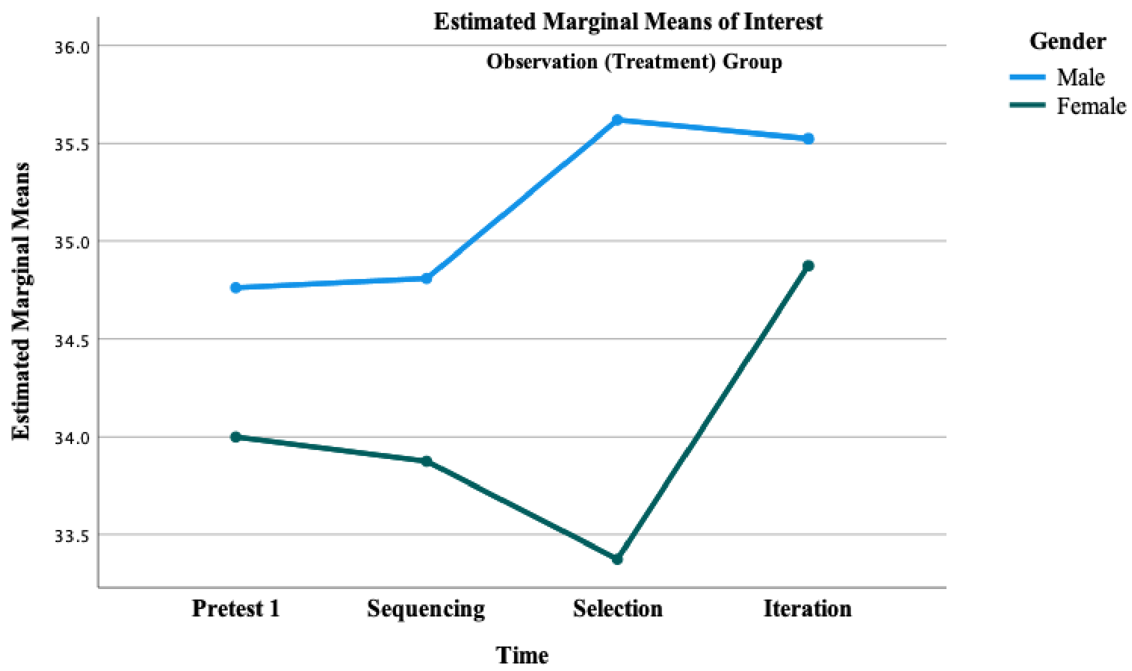


Table 20

Means Analysis for Interest

Group	Pretest 1	Pretest 2	Measure 1	Measure 2	Week 3
Control	3.92	3.88	3.93	4.02	4.05
Observation	4.32	4.29	4.32	4.38	4.42

Range (1: Very Low Interest, 5: Very High Interest)

Table 21

Means Analysis for Interest by Gender

Group	Pretest 1	Pretest 2	Measure 1	Measure 2	Week 3
Control Boys	3.97	3.92	3.99	4.12	4.14
Control Girls	3.63	3.70	3.63	3.50	3.58
Observation Boys	4.35	4.37	4.35	4.45	4.44
Observation Girls	4.25	4.09	4.23	4.17	4.36

Range (1: Very Low Interest, 5: Very High Interest)

Table 22

ANOVA Summary for Interest

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Time * Group	Sphericity Assumed	.978	3	3.792	.050	.604	.001
	Greenhouse-Geisser	.978	2.203	5.163	.050	.547	.001
Time * Gender	Sphericity Assumed	20.539	3	6.846	1.053	.721	.018
	Greenhouse-Geisser	20.539	2.203	9.323	1.053	.649	.018
Time * Group * Gender	Sphericity Assumed	6.039	3	2.013	.351	.985	.005
	Greenhouse-Geisser	6.039	2.203	2.741	.351	.955	.005
Error (Time)	Sphericity Assumed	1092.434	168	6.503			
	Greenhouse-Geisser	1092.434	123.376	8.855			

Variable Effect Results

Although expected rises in self-efficacy and interest occurred across the study period, even indicating a slight struggle by some students while learning selection (Week 2), the endpoints between control and observation groups were not statistically different. This was also true between genders.

CHAPTER 5

Results and Conclusions

Every fall semester, high school students begin their college applications. Not enough of them are pursuing computer science. Recruiting begins – at a minimum – in high school. Current computer science programs are not keeping pace with industry needs (Bowman, 2018). We must do more to attract and hold the interest of students selecting their career paths, or industry will continue to suffer.

Educators have been looking at the use of robotics in classrooms to drive interest for years. This study sought to add to this discovery process by introducing fully assembled humanoid robots, which represent a new breed of easy-to-use devices that are fully programmable in multiple languages. While results of this quasi-experimental treatment study did not show increases in interest or self-efficacy, they did point to reasonable *equivalency*.

Research Questions Analysis

Research Question 1

How does the use of robotics in programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?

For this question the study failed to reject the null hypothesis. There was no statistical difference detected between control and observation groups. Despite a double pretest and repeated measure design, there was no detected difference. Both groups increased self-efficacy across the training periods in similar manners.

Research Question 2

How does the use of robotics in programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?

For this question the study failed to reject the null hypothesis as well. There was no statistical difference detected between control and observation groups. Despite a double pretest and repeated measure design, there was no detected difference. Both groups increased interest across the training periods in similar manners.

Research Question 3

Is there a difference between genders in how the use of robotics programming activities influence the self-efficacy of students towards completion of advanced secondary computer science courses and pathway completion?

For this question the study failed to reject the null hypothesis as well. There was no statistical difference detected between boys and girls. Despite a double pretest and repeated measure design, there was no detected difference. Both groups increased self-efficacy across the training periods in similar manners. Visually the boys tend to show a slightly higher level of self-efficacy, but because of sample size constraints that difference should be taken as minor at best.

Research Question 4

Is there a difference between genders in how the use of robotics programming activities influence the interest of students towards completion of advanced secondary computer science courses and pathway completion?

For this question the study failed to reject the null hypothesis as well. There was no statistical difference detected between boys and girls. Despite a double pretest and repeated measure design, there was no detected difference. Visually, control group girls did show a slight decrease in interest. Boys tended to indicate higher interest across both groups, and earlier interest gains, with girls closing the gap by the end of the study. Again, however, because of sample size constraints these indications should be taken as minor at best. Interest does, however, seem to indicate the biggest chance for differences between the genders.

Research Conclusions

Prima Facia

Although this study failed to reject its null hypotheses, it had meaningful value. Showing robotics to increase interest and self-efficacy would have clearly been an outstanding result. Nevertheless, since both groups ended up at fundamentally the same place it does *tend* to indicate no *harm*. Robotics use became a viable *alternative*. Adding alternative tools to a teacher's toolkit is always of benefit to students.

Interest by students was voiced repeatedly to the researcher by students in the observation group during the study, and again by the control group after the study, when teaching was reversed to afford equal robotics experience to all students. These robots are continuing their use for this class, and as part of cross-curricular activities with engineering students as well.

No Effect

When considering a lack of effect from this study, it is important to look at teacher effect. It is entirely reasonable to believe that the researcher's experience led to a pedagogy in which both groups were able to master the concepts regardless of the differing tools used to achieve the learning. If this is the case, then the use of the robots demonstrated their utility as an alternative classroom tool.

Despite no statistical differences between endpoints between study groups or gender, the results did indicate growth of self-efficacy and interest in students. As students learned the fundamentals of programming, they showed growth across both variables over time.

Recommendations for Future Research

With careful design aimed at maximizing validity, there was either insufficient power to observe differences, or no differences truly existed, as would be the case in which teacher effect was a factor. Nevertheless, more power and longer designs will be important for future studies. Smaller study designs are simply not as extensible.

Future studies should include more longitudinal measurements, to better capture changes in self-efficacy and interest in students. These types of longer designs can be staged with several logical intervals. Semester-long studies, year-long entire-course designs, and even longitudinal designs that cover an entire high school experience for students could all prove beneficial to the literature. There will always be a place for smaller quasi-experimental studies, but longer studies might reveal more definitive shifts in attitudes and career decisions by students emerging from secondary education.

Another important consideration should be to find innovative ways to study specific courses or even programming languages across a given county or state. Information from a course or language specific study across multiple counties or institutions could find areas of strengths and weaknesses within the pedagogy itself, leading to improved coursework. As a rapidly evolving field of study, computer science may require researchers to rethink and even reinvent study designs. If done correctly, we might discover ways in which education can begin to close the industry gap.

Reflections Post-Analysis

Computer science is a rapidly evolving field of study – and sources agree that our educational system is not keeping pace. Curriculum decisions made 15 years ago may have been well-suited for the time, but they have not evolved at the pace of this exploding industry.

If we are going to compete with other nations, we need a more agile system for educating future programmers. Execution of this study reaffirmed the concerns of the researcher specifically when the discussions around power were being weighed. Classrooms across a *single county* in Georgia do not have sufficient similarities to conduct reliable large-sample research for *any* of the available courses or pathways in computer science. Pulling back to a statewide view brings even further disparity. Obtaining a sufficient sample size population to provide a pilot for this study's instrument was even considered a significant challenge.

Confounding this issue is a lack of computer science teachers trained *as programmers.* The researcher, in discussion with the state Department of Education (DOE), was recruited to help form semester-long professional development programming courses for over 175 current educators who have requested this type of training within the state.

As more educators become experienced programmers, the ability to expand research across schools and regions will be enhanced. This will significantly improve the ability for researchers to increase sample sizes in a meaningful way. This is not to say that studies like this one are unimportant. By no means is this the case. Individual classrooms will forever be limited in size for studies, but there is still a place for quasi-experimental studies to help educators make decisions for their students' best interests and success (Campbell, 1966).

We can, however, create more fertile ground for future research to provide increased extensibility of findings. This is a herculean task given the rapid outpacing of education by industry, but it is well worth the effort.

Situational Opportunities

Schools service local areas. While the researcher's school rapidly adopting embedded devices like robots and Raspberry Pi computing devices, other districts might be trying to rid themselves of these very machines. Parents and school boards can have a wide variety of expectations based on local industry and even regional employer demands. Rural areas may seek more embedded-programming device expertise for emerging farming technology, where other areas seek robotics expertise in graduates due to a specific industry performing manufacturing in the community. These dynamics should be considered an opportunity to experiment and learn. We should seek to accomplish broadly reaching standards while maintaining choice and voice for our local communities, to include everyone from our industry partners, parents, educators, and certainly our students themselves.

Final Thoughts

This quasi-experimentation classroom study was easily worth every effort it consumed. It is reasonable to say that despite the quasi-experimental nature of a classroom-borne study, much was gleaned, even if indirectly. At least through the lens of *this* school, and *these* students, robots have become an alternative way to engage students while remaining firmly grounded within the domain of theoretical computer science. Education in technology needs more options rooted in computer science that can bring content to life.

Good teaching should never be fully displaced by automation. It can, however, enhance the toolbox a teacher brings to bear. For this study, a small collection of humanoid robots sparked enough of a change that they are now being dressed (clothed) by students in a fashion design class and programmed to walk a modeling runway by the very classes that took part in this study. Engaging students has never been so fashionable.

References

Adrion, R., Fall, R., Ericson, B., & Guzdial, M. (2016). Broadening Access to Computing Education State by State. Communications of the ACM, 59(2), 32–34. https://doi.org/10.1145/2856455

Akbulut, A. Y. (2015). The Impact of the Introductory IS Course on Students' Perceptions of IS Professionals. *Journal of Information Systems Education*, *26*(4), 295–304. http://jise.org

Akbulut-Bailey, A. (2012). Improving IS Enrollment Choices: The Role of Social Support. *Journal of Information Systems Education*, *23*(3), 259–270. http://jise.org

Akbulut, A. Y., & Looney, C. A. (2007). Inspiring Students to Pursue Computing Degrees. *Communications of the ACM*, *50*(10), 67–71. https://doi.org/10.1145/1290958.1290964

Anderson, D. (2014). Tom Kilburn: A Tale of Five Computers. *Communications of the ACM*, *57*(5), 35–38. doi :10.1145/2594290

Andrews, F. (1984). Construct Validity and Error Components of Survey Measures: A Structural Modeling Approach. *The Public Opinion Quarterly*, *48*(2), 409–442.

Ash, R. A., Coder, L., Dupont, B., & Rosenboom, J. L. (2009). Examining the Obstacles to Broadening Participation in Computing: Evidence from a Survey of Professional Workers. *Contemporary Economic Policy*, (3), 413. https://doi.org/10.1111/j.1465-7287.2008.00135.x

Atchison, William F. (1973). The Impact of Computer Science Education on the Curriculum. *The mathematics teacher*, *66*(1), 7–83.

Auerbach, A. A., Eckert, J. P., Shaw, R. F., Weiner, J. R., & Wilson, L. D. (1952). The Binac. *Proceedings of the IRE, Proc. IRE*, *40*(1), 12–29. https://doi.org/ 10.1109/JRPROC.1952.273922

Bandura, A. (1986). *Social foundations of thought and action : a social cognitive theory*. Englewood Cliffs, NJ: Prentice-Hall.

Bandura, A. (1990). Some Reflections on Reflections. *Psychological inquiry*, *1*(1), 101. https://doi.org/10.1207/s15327965pli0101_26

Bandura, A. (2001). Social cognitive theory: an agentic perspective. *Annual review of psychology*, *52*, 1–26.

Bandura, A. (2002). Social Cognitive Theory in Cultural Context. *Applied psychology: An international review*, *51*(2), 269–290. https://doi.org/10.1111/1464-0597.00092

Barone, L. M., Ricci-Tersenghi, F., Organtini, G., & Marinari, E. (2014). *Scientific programming: C-language, algorithms and models in science.*

Betz, N. E. (1994). Self-concept theory in career development and counseling. *Career development quarterly*, *43*(1), 32. https://doi.org/10.1002/j.2161-0045.1994.tb00844.x

Blouin, J. S. (2011). *High school seniors' computer self-efficacy and interest in computer science careers*. (Doctoral dissertation). Retrieved from UGA GIL-Find Catalog. (uga.9938912043902959).

Bourke, J., Kirby, A., & Doran, J. (2016). *Survey and questionnaire design: Collecting primary data to answer research questions*. NuBooks.

Bowman, D. (2018). Declining Talent in Computer Related Careers. *Journal of Academic Administration in Higher Education*, *14*(1), 1–4.

Briddick, W. C. (2009). Frank Findings: Frank Parsons and the Parson Family. *Career* *development quarterly*, *57*(3), 207–214. https://doi.org/10.1002/j.2161-0045.2009.tb00106.x

Briddick, W. C. (2009). Frank Parsons on interests. *Journal of vocational behavior*, *74*(2), 230-233. https://doi.org/10.1016/j.jvb.2008.12.003

Brown, S. D., Lent, R. W., & Larkin, K. C. (1989). Self-efficacy as a moderator of scholastic aptitude: Academic performance relationships. *Journal of Vocational Behavior*, *35*(1), 64–75. https://doi-org.proxy-remote.galib.uga.edu/10.1016/0001-8791(89)90048-1

Brown, S. D., Tramayne, S., Hoxha, D., Telander, K., Fan, X., & Lent, R. W. (2008). Social cognitive predictors of college students' academic performance and persistence: A meta-analytic path analysis. *Journal of Vocational Behavior*. 72, 298-308.

Burgiel, H., Sadler, P. M., & Sonnert, G. (2020). The Association of High School Computer Science Content and Pedagogy with Students' Success in College Computer Science. *ACM transactions on computing education*, *20*(2).

Burnette, J. L., Hoyt, C. L., Russell, V. M., Lawson, B., Dweck, C. S., & Finkel, E. (n.d.). A Growth Mind-Set Intervention Improves Interest but Not Academic Performance in the Field of Computer Science. *Social Psychological and Personality Science*, *11*(1), 107-116. https://doi.org/10.1177/1948550619841631

Campbell, Donald T., Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Chicago: R McNally.

Campusano, M., Fabry, J., & Bergel, A. (2019). Live programming in practice: A controlled experiment on state machines for robotic behaviors. *Information & Software Technology*, *108*, 99–114. https://doi.org/10.1016/j.infsof.2018.12.008

Carter, L. (2006). Why Students with an Apparent Aptitude for Computer Science Don't Choose to Major in Computer Science. *SIGCSE Bulletin*, (NO 1), 27. https://doi.org/10.1145/1121341.1121352

Century, J., Ferris, K. A., & Zuo, H. (2020). Finding Time for Computer Science in the Elementary School Day: A Quasi-Experimental Study of a Transdisciplinary Problem Based Learning Approach. *International journal of STEM education*, *7*.

Cera, G., Mlouk, A., Cera, E., & Shumeli, A. (2020). The Impact of Entrepreneurship Education on Entrepreneurial Intention. A Quasi-Experimental Research Design. *Journal of Competitiveness*, *1*, 39–56. https://doi.org/joc.2020.01.03

Cliburn, D. C. (2006). A CS0 course for the liberal arts. *Proceedings of the 37th SIGCSE Technical Symposium: Computer Science Education*, 77. https://dl.acm.org

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences*. (2nd ed.) Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Cohen, L., Manion, L., & Morrison, K. (2007). *Research methods in education* (6th ed.). Routledge.

Computer Engineering Is Marriage of Computer Science, Electrical Engineering -Don. (2020, August 18). *Nigerian Tribune (Oyo State, Nigeria)*.

Computing Curricula 2020 Paradigms for Global Computing Education. (2020, December 31). https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2020.pdf

Cook D., & King J. (1968). A Study of the Hawthorne Effect in Educational Research. *Research in the Teaching of English*, *2*(2), 93–98.

Cook, T.D., & Campbell, D.T. (1979). *Quasi-Experimentation: Design and analysis issues for* *field settings*. Chicago: Rand-McNally.

Creswell, J. W., & Creswell, J. D. (2018). *Research design: Qualitative, quantitative, and mixed methods approaches*. Thousand Oaks, CA: SAGE Publications.

Dasgupta, S. (2016). *Computer science, a very short introduction*. Oxford: Oxford University Press.

Dillman, D. A., Smyth, J. D., & Christian, L. M. (2014). *Internet, phone, mail, and mixed-mode surveys: The tailored design method*. Hoboken, NJ: Wiley.

Dsouza, R., Quinonez, R., Hubbell, S., & Brame, J. (2019). Promoting oral health in nursing education through interprofessional collaborative practice: A quasi-experimental survey study design. *Nurse Education Today*, *82*, 93–98. https://doi.org/j.nedt.2019.07.010

Edwards, K. J., & Whitney, D. R. (1972). Structural analysis of Holland's personality types using factor and configural analysis. *Journal of counseling psychology*, *19*(2), 136–145. https://doi.org/ 10.1037/h0032637

Egleston, B. L., Miller, S. M., & Meropol, N. J. (2011). The impact of misclassification due to survey response fatigue on estimation and identifiability of treatment effects. *Statistics in medicine*, *30*(30), 3560–3572. https://doi.org/10.1002/sim.4377

Fagin, B., & Merkle, L. (2003). Measuring the Effectiveness of Robots in Teaching Computer Science. *SIGCSE BULLETIN*, (NO 1), 307. https://dl.acm.org.

Fathalla, S., Vahdati, S., Lange, C., & Auer, S. (2020). Scholarly event characteristics in four fields of science: a metrics-based analysis. *Scientometrics*, *123*(2), 677–705. https://doi.org/10.1007/s11192-020-03391-y

Field, A. P. (2018). *Discovering statistics using IBM SPSS statistics*. Thousand Oaks, CA: Sage Publications.

Fowler, F. J. (2014). *Survey research methods*. London: Sage Publication.

Freeman, S. C. (1993). Donald Super: A Perspective on Career Development. *Journal of career development (Springer Science & Business Media B.V.)*, *19*(4), 255–264. https://doi.org/10.1007/BF01354628

Frieze, C., & Quesenberry, J. L. (2019). How Computer Science at CMU Is Attracting and Retaining Women: Carnegie Mellon University's successful efforts enrolling, sustaining, and graduating women in computer science challenge the belief in a gender divide in CS education. *Communications of the ACM*, *62*(2), 23–26. https://doi.org/10.1145/3300226

Gee, E., Tran, K. M., & Parekh, P. (2020). Designing Analog Games That Engage Girls with Computer Science Concepts. *International Journal of Designs for Learning*, *11*(2), 17 26.

Georgia Department of Education (2021). Information Technology. https://www.gadoe.org/ Curriculum-Instruction-and-Assessment/CTAE/Pages/cluster-IT.aspx.

Georgia State Senate Bill (2019). https://gov.georgia.gov/sites/gov.georgia.gov/files/related _files/document/SB%20108.pdf

Goodman, J., & Anderson, M. L. (2012). Applying Schlossberg's 4-S Transition Model to Retirement. *Career planning & adult development journal*, *28*(2), 10–20.

Gottfredson, G. D., Jones, E. M., & Holland, J. L. (1993). Personality and vocational interests: the relation of Holland's six interest dimensions to five robust dimensions of personality. *Journal of counseling psychology*, *4*, 518.

Gray, J. *Men are from mars, women are from venus*, Harper Collins, New York. 1992.

Gray, G. T., & Smith, R. Q. (2004). Sperry Rand's First-Generation Computers, 1955-1960:

Hardware and Software. *IEEE Annals of the History of Computing*, *26*(4), 20–34. https://doi.org/10.1109/MAHC.2004.34

Griffin, K. A., & Gilbert, C. K. (2015). Better Transitions for Troops: An Application of Schlossberg's Transition Framework to Analyses of Barriers and Institutional support structures for student veterans. *Journal of higher education*, *86*(1), 71–97. https://doi.org/10.1080/00221546.2015.11777357

Grimminger-Seidensticker, E., & Möhwald, A. (2020). Enhancing social cohesion in PE classes within an intercultural learning program: results of a quasi-experimental intervention study. *Physical Education & Sport Pedagogy*, *25*(3), 316–329.

Hannon, M. J. (2020). A Brief History of Memory Forensics, How Volatile Computer Memory Works, and Malware and Volatile Memory. *Computer & Internet Lawyer*, *37*(4), 8–19.

Happe, L., Buhnova, B., Koziolek, A., & Wagner, I. (2020). Effective measures to foster girls' interest in secondary computer science education A Literature Review. *EDUCATION AND INFORMATION TECHNOLOGIES*. https://doi.org/10.1007/s10639-020-10379-x

Herr, E. L. (1997). Super's life-span, life-space approach and its outlook for refinement. *Career development quarterly*, *45*(3), 238–246. https://doi.org/10.1002/j.2161-0045.1997.tb00468.x

Holland, J. L., Gottfredson, G. D., & Johnstun, M. L. (2009). Vocational Preference Inventory. *Career development quarterly*, *58*, 99–107.

Holland, J. L., Gottfredson, D. C., & Power, P. G. (1980). Some diagnostic scales for research in decision making and personality: Identity, information, and barriers. *Journal of personality and social psychology*, *39*(6), 1191–1200. https://doi.org/10.1037/h0077731

Holland, J. L., Toomey, K. D., Levinson, E. M., & Palmer, E. J. (2009). Self-Directed

Search. *Journal of employment counseling*, *46*, 82–93.

Hodges, B. D. (2018). Learning from Dorothy Vaughan: artificial intelligence and the health professions. *Medical education*, *52*(1), 11–13. https://doi.org/10.1111/medu.13350

Huang, F. L., & Cornell, D. G. (2016). Question Order Affects the Measurement of Bullying Victimization Among Middle School Students. *Educational & Psychological Measurement*, *76*(5), 724–740. https://doi.org/10.1177/0013164415622664

Johns Creek High School Profile (2020-21): Alpharetta, GA. (2020). Retrieved December 10, 2020. http://publicschoolreview.com/johns-creek-high-school-profile.

Jones, L. K. (1994). Frank Parsons' Contribution to Career Counseling. *Journal of career development (Springer Science & Business Media B.V.)*, *20*(4), 287–294. https://doi.org/10.1007/BF02106301

Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2013). The Effect of a Classroom-Based Intensive Robotics and Programming Workshop on Sequencing Ability in Early Childhood. *Early Childhood Education Journal*, *41*(4), 245–255. https://eric.ed.gov

Kemp, P. E. J., Wong, B., & Berry, M. G. (2020). Female Performance and Participation in Computer Science: A National Picture. *ACM Transactions on Computing Education*, *20*(1).

Khenner, E., Frieze, C. & Zane, O. (2020). IT education as a factor to influence gender imbalances in computing: Comparing Russian and American experience. *Obrazovanie i Nauka*, *22*(8), 189–206. https://doi.org/10.17853/1994-5639-2020-8-189-206

Korkmaz, Ö. (2016). The Effect of Lego Mindstorms Ev3 Based Design Activities on Students' Attitudes towards Learning Computer Programming, Self-Efficacy Beliefs and Levels of Academic Achievement. *Baltic J. Modern Computing*, *4*, 4. https://eric.ed.gov

Korkut Altuna, O., & Arslan, F. M. (2016). Impact of the Number of Scale Points on Data Characteristics and Respondents' Evaluations: An Experimental Design Approach Using 5-Point' and 7-Point Likert-Type Scales. *Journal of Faculty of Political Science*, *55*, 1–20. https://doi.org/10.17124/iusiyasal.320009

Krumboltz, J. D. (2009). The Happenstance Learning Theory. Journal of Career Assessment, 17(2), 135-154. https://doi.org/10.1177/1069072708328861

Krumboltz, J. D. (2011). Capitalizing on Happenstance. *Journal of employment counseling*, *48*(4), 156–158.

Krumboltz, J. D., & Duckham-Shoor, L. (1977). Reward Direction, Not Perfection. *Theory into practice*, *16*(4), 243.

Krumboltz, J. D., Foley, P. F., & Cotter, E. W. (2013). Applying the Happenstance Learning Theory to Involuntary Career Transitions. *Career development quarterly*, *61*(1), 15–26.

Krumboltz, J. D., & Worthington, R. L. (1999). The School-to-Work Transition from a Learning Theory Perspective. *Career development quarterly*, *47*(4), 312–325. https://doi.org/ j.2161-0045.1999.tb00740.x

Lasorsa, D. L. (n.d.). Question-Order Effects in Surveys: The Case of Political Interest, News Attention, and Knowledge. *Journalism & Mass Communication Quarterly.*, *80*(3), 499 512.

Ledgard, H. F., & Marcotty, M. (1975). *Communications of the ACM*, *18*(11), 629–639. https://doi.org/10.1145/361219.361222

Lent, R. W., & Brown, S. D. (1996). Social cognitive approach to career development: An overview. *Career Development Quarterly*, 44(4), 310. https://doi.org/10.1002/j.2161-0045.1996.tb00448.x

Lent, R. W., & Brown, S. D. (2006). On Conceptualizing and Assessing Social Cognitive Constructs in Career Research: A Measurement Guide. *Journal of Career Assessment*, 14(1), 12–35. https://doi.org/10.1177/1069072705281364

Lent, R. W., & Brown, S. D. (2013). Social cognitive model of career self-management: Toward a unifying view of adaptive career behavior across the life span. *Journal of Counseling Psychology*, *60*(4), 557–568. https://doi.org/10.1037/a0033446

Lent, R. W., & Brown, S. D. (2019). Social cognitive career theory at 25: Empirical status of the interest, choice, and performance models. *Journal of Vocational Behavior*, 115. https://doi.org/10.1016/j.jvb.2019.06.004

Lent, R. W., & Brown, S. D. (2020). Career decision making, fast and slow: Toward an integrative model of intervention for sustainable career choice. *Journal of Vocational Behavior*, *120*. https://doi.org/10.1016/j.jvb.2020.103448

Lent, R. W., Brown, S. D., & Hackett, G. (1994). Toward a Unifying Social Cognitive Theory of Career and Academic Interest, Choice, and Performance. *Journal of Vocational Behavior*, *1*, 79.

Lent, R. W., Brown, S. D., & Hackett, G. (2000). Contextual supports and barriers to career choice: A social cognitive analysis. *Journal of Counseling Psychology*, 47(1), 36–49. https://doi.org/10.1037/0022-0167.47.1.36

Lent, R. W., Hackett, G., & Brown, S. D. (1999). A Social Cognitive View of School-to-Work Transition. *Career Development Quarterly*, 47(4), 297–311. https://doi.org/10.1002/j.2161-0045.1999.tb00739.x

Lent, R. W., Ireland, G. W., Penn, L. T., Morris, T. R., & Sappington, R. (2017). Sources of self-efficacy and outcome expectations for career exploration and decision-making: A test of the social cognitive model of career self-management. *Journal of Vocational Behavior*, *99*, 107–117. https://www.journals.elsevier.com/journal-of-vocational-behavior

Lent, R. W., Lopez, J. A. M., Lopez, F. G., & Sheu, H. B. (2008). Social cognitive career theory and the prediction of interests and choice goals in the computing disciplines. *Journal of vocational behavior*, *73*(1), 52–62. https://doi.org/10.1016/j.jvb.2008.01.002

Lent, R. W., Miller, M. J., Smith, P. E., Watford, B. A., Hui, K., & Lim, R. H. (2015). Social cognitive model of adjustment to engineering majors: Longitudinal test across gender and race/ethnicity. *Journal of vocational behavior*, *86*, 77–85. https://doi.org/10.1016/j.jvb.2014.11.004.

Liu, A., Newsom, J., Schunn, C., & Shoop, R. (2013). Students Learn Programming Faster through Robotic Simulation. *Tech Directions*, *72*(8), 16–19. http://techdirections.com

Lynn, K.-M., Raphael, C., Olefsky, K., & Bachen, C. M. (2003). Bridging the Gender Gap in Computing: An Integrative Approach to Content Design for Girls. *Journal of Educational Computing Research*, *28*(2), 143–162.

Lykke, M., Coto, M., Jantzen, C., Mora, S., & Vandel, N. (2015). Motivating Students through Positive Learning Experiences: A Comparison of Three Learning Designs for Computer Programming Courses. *Journal of Problem Based Learning in Higher Education*, *3*(2), 80–108. https://eric.ed.gov

Majherová, J., & Králík, V. (2017). Innovative Methods in Teaching Programming for Future Informatics Teachers. *European Journal of Contemporary Education*, *6*(3), 390–400. https://eric.ed.gov

Menekse, M. (2015). Computer Science Teacher Professional Development in the United States:

A Review of Studies Published between 2004 and 2014. *Computer science education*, *25*(4), 325–350.

Mertler, C. A. (2019). *Introduction to educational research*, 2nd Edition. (2018, April).

Mohd Radzi, S. H., Tan, W. H., & Yusoff, A. (2019). Shipping Management Simulation Game for Teaching and Learning in Higher Education: A Quasi-Experimental Study. *Malaysian Journal of Learning and Instruction*, *16*(2), 155–186.

Nofre, D., Priestley, M., & Alberts, G. (2014). When technology became language: The origins of the linguistic conception of computer programming, 1950-1960. *Technology and Culture, 55*(1), 40-75.

O'Brien, K. M. (2001). The Legacy of Parsons: Career Counselors and Vocational Psychologists as Agents of Social Change. *Career development quarterly*, *50*(1), 66. https://doi.org/10.1002/j.2161-0045.2001.tb00891.x

Pan, Z., Polden, J., Larkin, N., Van Duin, S., & Norrish, J. (2012). Recent progress on programming methods for industrial robots. *Robotics and Computer Integrated Manufacturing*, *2*, 87.

Pappas, J. P. (1978). Pioneers in Guidance: Donald Super. *Personnel & guidance journal*, *56*(10), 585.

Pelham, B. W., Blanton, H. (2007). *Conducting research in psychology: measuring the weight of smoke.* Belmont, CA: Thomson, Wadsworth.

Phillips, P. P., Aaron, B. C., & Phillips, J. J. (2013). *Survey basics*. American Society for Training & Development.

Pintrich, P. R., & de Groot, E. V. (1990). Motivational and self-regulated learning components of classroom academic performance. *Journal of Educational Psychology*, *82*(1), 33–40. https://doi.org/10.1037/0022-0663.82.1.33

Pullan, M. (2013). Using Robotics to Improve Retention and Increase Comprehension in Introductory Programming Courses. *Journal of Educational Technology Systems*, *42*(2), 141–149.

Rosenbloom, P. S. (2004). A new framework for computer science and engineering. *Computer*, *37*(11), 23–28. https://doi.org/10.1109/MC.2004.186

Rudolph, C. W., Zacher, H., & Hirschi, A. (2019). Empirical developments in career construction theory. *Journal of vocational behavior*, *111*, 1–6.

Sadik, O., Ottenbreit-Leftwich, A. T., & Brush, T. A. (2020). Secondary Computer Science Teachers' Pedagogical Needs. *International journal of computer science education in schools*, *4*(1).

Saretsky, G. (1975). *The John Henry Effect: Potential Confounder of Experimental vs. Control Group Approaches to the Evaluation of Educational Innovations.* Paper presented at the annual meeting of the American Educational Research Association, Washington, D.C.

Savickas, M. L. (2008). David V. Tiedeman: Engineer of Career Construction. *Career development quarterly*, *56*(3), 217–224. https://doi.org/10.1002/j.2161-0045.2008.tb00035.x

Schlossberg, N. K. (2011). The challenge of change: The transition model and its applications. *Journal of employment counseling*, *48*(4), 159–162. https://doi.org/10.1002/j.2161-1920.2011.tb01102.x

Schmider, E., Ziegler, M., Danay, E., Beyer, L., & Bühner, M. (2010). Is it really robust? Reinvestigating the robustness of ANOVA against violations of the normal distribution assumption. Methodology: European Journal of Research Methods for the Behavioral and Social Sciences, 6(4), 147–151. https://doi.org/10.1027/1614-2241/a000016

Schwarz, N. (1999). *Self-reports: how the questions shape the answers*. *2*, 93.

Shadish, W. R., Cook, T. D., & Campbell, D. T. (2002). *Experimental and quasi-experimental designs for generalized causal inference*. Belmont, CA: Wadsworth Cengage Learning.

Singh, A., & Singh, S. (2016). A comparative study of early V/S modern computers. *International Journal of Advanced Research in Computer Science, 7*(6).

Smith Stephen W., Daunic Ann P., & Taylor Gregory G. (2007). Treatment Fidelity in Applied Educational Research: Expanding the Adoption and Application of Measures to Ensure Evidence-Based Practice. *Education and treatment of children*, *30*(4), 121–134.

Song, M., & Herman, R. (2010). A Practical Guide on Designing and Conducting Impact Studies in Education: Lessons Learned from the What Works Clearinghouse (Phase I). American Institutes for Research.

Spybrook, J. (2005). Optimal Design Plus Empirical Evidence (software), v3.01, HLM Software.

Strack, F., Martin, L. L., & Schwarz, N. (1988). Priming and communication: Social determinants of information use in judgments of life satisfaction. *European Journal of Social Psychology*, *18*(5), 429–442. https://doi.org/10.1002/ejsp.2420180505

Šverko, I., & Babarović, T. (2019). Applying career construction model of adaptation to career transition in adolescence: A two-study paper. *Journal of vocational behavior*, *111*, 59 73. https://doi.org/10.1016/j.jvb.2018.10.011

Swain, D. A. (1991). Withdrawal from Sport and Schlossberg's Model of Transitions. *Sociology of sport journal*, *8*(2), 152–160. https://doi.org/10.1123/ssj.8.2.152

Szumski, G., & Karwowski, M. (2019). Exploring the Pygmalion effect: The role of teacher expectations, academic self-concept, and class context in students' math achievement. *Contemporary educational psychology*, *59*. https://doi.org/10.1016/j.cedpsych.2019.101787

Tatnall, A., Blyth, T., & Johnson, R. (2013). *Making the history of computing relevant: IFIP WG 9.7 International Conference, HC 2013, London, UK, June 17-18, 2013, Revised Selected Papers*. Springer.

Thyer, B. (2012). *Quasi-experimental research designs (Pocket guides to social work research methods).* New York: Oxford University Press.

Tobin, S., Jayabalasingham, B., Huggett, S., de Kleijn, M., & Lawlor, B. (2019). A brief historical overview of artificial intelligence research. *Information Services & Use*, *39*(4), 291–296. https://doi.org/10.3233/ISU-190060

Tocháček, D., Lapeš, J., & Fuglík, V. (2016). Developing Technological Knowledge and Programming Skills of Secondary Schools Students through the Educational Robotics Projects. *Procedia - Social and Behavioral Sciences*, *217*, 377–381. https://doi.org/10.1016/j.sbspro.2016.02.107

Tsai, M.-J., Wang, C.-Y., & Hsu, P.F. (2019). Developing the Computer Programming Self Efficacy Scale for Computer Literacy Education. *Journal of Educational Computing Research*, *56*(8), 1345–1360.

Ümran, A. (2020). Effect of number of response options on the psychometric properties of Likert-type scales used with children. *Studies in Educational Evaluation: SEE., 66*, Studies in educational evaluation: SEE., 2020, Vol.66.

U.S. Bureau of Labor Statistics (2021). *Software developers: Occupational outlook handbook.* (2020, September 1). https://www.bls.gov/ooh/computer-and-information technology/software-developers.htm#tab-6

U.S. Department of Labor, Bureau of Labor Statistics. (2020). Employment Projections - 2019-2029. https://www.bls.gov/news.release/pdf/ecopro.pdf

Villamil, S., Hernández, C., & Tarazona, G. (2020). An overview of internet of things. *Telkomnika*, *18*(5), 2320–2327. https://doi.org/ 10.12928/TELKOMNIKA.v18i5.15911

Waddington, H., Aloe, A. M., Becker, B. J., Djimeu, E. W., Hombrados, J. G., Tugwell, P., Wells, G., & Reeves, B. (2017). Quasi-experimental study designs series—paper 6: risk of bias assessment. *Journal of Clinical Epidemiology*, 89, 43–52. https://doi.org/10.1016/j.jclinepi.2017.02.015

Wall, P., Fetherston, C., & Browne, C. (2018). Understanding the enrolled nurse to registered nurse journey through a model adapted from Schlossberg's transition theory. *Nurse education today*, *67*, 6–14. https://doi.org/10.1016/j.nedt.2018.04.01

Wang, D., Databytes: Growth in Computer Science and Engineering. (2018). *ASEE Prism*, *28*(3), 16.

Wang, J. (2017). Is the U.S. Education System Ready for CS for All? Insights from a recent Google-Gallup national research study seeking to better understand the context of K–12 CS education. *Communications of the ACM*, *60*(8), 26–28. https://doi.org/10.1145/3108928

Weinrach, S. G. (1996). The Psychological and Vocational Interest Patterns of Donald Super and John Holland. *Journal of counseling & development*, *75*(1), 5–16. https://doi.org/10.1002/j.1556-6676.1996.tb02309.x

Wilson, C., & Harsha, P. (2009). The Long Road to Computer Science Education

Reform. *Communications of the ACM*, *52*(9), 33–35. https://doi.org/10.1145/1562164.1562178

Witherspoon, E. B., Schunn, C. D., Higashi, R. M., & Shoop, R. (2018). Attending to Structural Programming Features Predicts Differences in Learning and Motivation. *Journal of Computer Assisted Learning*, *34*(2), 115–128. http://eric.ed.gov

Wood, R., & Bandura, A. (1989). Social Cognitive Theory of Organizational Management. *Academy of management review*, *14*(3), 361–384. https://doi.org/10.5465/AMR.1989.4279067

Xie, B., Xia, M., Xin, X., & Zhou, W. (2016). Linking calling to work engagement and subjective career success: The perspective of career construction theory. *Journal of vocational behavior*, *94*, 70–78. https://doi.org/10.1016/j.jvb.2016.02.011

Yatsko, V. A. (2017). The principles for the investigation of the historical development of computer science. *Scientific and Technical Information Processing, 44*(3), 207-214. https://doi.org/10.3103/S0147688217030108

Zhu, F., Cai, Z., Buchtel, E. E., & Guan, Y. (2019). Career construction in social exchange: a dual-path model linking career adaptability to turnover intention. *Journal of vocational behavior*, *112*, 282–293. https://doi.org/10.1016/j.jvb.2019.04.003

Zweben, S. & Bizot, B. (2016). Taulbee Survey. Computing Research Association 29, 5 (May 2017), 3–51; https://cra.org/crn/wp-content/uploads/sites/7/2017/05/2016-Taulbee-Survey.pdfA

Zytowski, D. G. (2001). Frank Parsons and the Progressive Movement. *Career development quarterly*, *50*(1), 57. https://doi.org/10.1002/j.2161-0045.2001.tb00890.x

Appendix A

Questionnaire Provided by Dr. Lent

Background Questionnaire

Student Number

0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

Year in School

- (F) Freshman
- (S) Sophomore
- (J) Junior
- (S) Senior
- (O) Other (please specify in box)

Race or ethnic group

- O Black or African American
- O Hispanic American
- O White or European American
- O Asian/Pacific Islander-American
- O Native American
- O Other (please specify in the box below)

Sex:

- (M) Male
- (F) Female

Please identify your intended major:

- O Aerospace
- O Biological Resources
- O Chemical
- O Civil/Environmental
- O Computer
- O Electrical
- O Fire Protection
- O Industrial
- O Materials
- O Mechanical
- O Naval Architecture
- O Nuclear
- O Ocean
- O Systems
- O BS-Engineering
- O Other (please specify below)

What was your Math SAT(or SAT-I) score?

0	0	0
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9

University:

- O Howard
- O University of Maryland
- O Morgan State
- O Naval Academy

(OVER PLEASE)

Engineering Fields Questionnaire

Part I. Instructions: The following is a list of major steps along the way to completing an engineering degree. Please indicate how much confidence you have in your ability to complete each of these steps in relation to the engineering major that you are most likely to pursue. Use the 0-9 scale below to indicate your degree of confidence.

How much confidence do you have in your ability to:	No Confidence at all 0	1	2	3	Some Confidence 4	5	6	Complete Confidence 7	8	9
1. Complete all of the "basic science" (i.e., math, physics, chemistry) requirements for your engineering major with grades of B or better	0	1	2	3	4	5	6	7	8	9
2. Excel in your engineering major over the next semester	0	1	2	3	4	5	6	7	8	9
3. Excel in your engineering major over the next two semesters	0	1	2	3	4	5	6	7	8	9
4. Complete the upper level required courses in your engineering major with an overall grade point average of B or better	0	1	2	3	4	5	6	7	8	9

Part II. Instructions: Here we are interested in knowing how well you believe you could cope with each of the following barriers, or problems, that students could possibly face in pursuing an engineering major. Please indicate your confidence in your ability to cope with, or solve, each of the following problem situations.

How confident are you that you could:	No Confidence at all 0	1	2	3	Some Confidence 4	5	6	Complete Confidence 7	8	9
1. Cope with a lack of support from professors or your advisor.	0	1	2	3	4	5	6	7	8	9
2. Complete a degree in engineering despite financial pressures.	0	1	2	3	4	5	6	7	8	9
3. Continue on in engineering even if you did not feel well-liked by your classmates or professors.	0	1	2	3	4	5	6	7	8	9
4. Find ways to overcome communication problems with professors or teaching assistants in engineering courses.	0	1	2	3	4	5	6	7	8	9
5. Balance the pressures of studying for engineering courses with the desire to have free time for fun and other activities.	0	1	2	3	4	5	6	7	8	9
6. Continue on in engineering even if you felt that, socially, the environment in these classes was not very welcoming to you.	0	1	2	3	4	5	6	7	8	9
7. Find ways to study effectively for engineering courses despite having competing demands for your time.	0	1	2	3	4	5	6	7	8	9

Part III. Instructions: Using the scale below, please indicate the extent to which you agree or disagree with each of the following statements.

Graduating with a BS degree in engineering will likely allow me to:	Strongly Disagree 0	1	Disagree 2	3	Unsure 4	5	Agree 6	7	Strongly Agree 8	9
1. ... receive a good job offer	0	1	2	3	4	5	6	7	8	9
2. ... earn an attractive salary	0	1	2	3	4	5	6	7	8	9
3. ... get respect from other people	0	1	2	3	4	5	6	7	8	9
4. ... do work that I would find satisfying	0	1	2	3	4	5	6	7	8	9
5. ... increase my sense of self-worth	0	1	2	3	4	5	6	7	8	9
6. ... have a career that is valued by my family	0	1	2	3	4	5	6	7	8	9
7. ... do work that can "make a difference" in people's lives	0	1	2	3	4	5	6	7	8	9
8. ... go into a field with high employment demand	0	1	2	3	4	5	6	7	8	9
9. ... do exciting work	0	1	2	3	4	5	6	7	8	9
10. ... have the right type and amount of contact with other people (i.e., "right" for me)	0	1	2	3	4	5	6	7	8	9

Part IV. Instructions: Now, please indicate your degree of interest in doing each of the following activities. Use the 1-5 scale to show how much interest you have in each activity.

How much interest do you have in...	Very Low Interest 1	Low Interest 2	Medium Interest 3	High Interest 4	Very High Interest 5
1. Solving practical math problems	1	2	3	4	5
2. Reading articles or books about engineering issues	1	2	3	4	5
3. Solving computer software problems	1	2	3	4	5
4. Working on a project involving engineering principles	1	2	3	4	5
5. Solving complicated technical problems	1	2	3	4	5
6. Learning new computer applications	1	2	3	4	5
7. Working on a project involving scientific concepts	1	2	3	4	5

(OVER PLEASE)

Part V. Instructions: Using the scale below, indicate your level of agreement with each of the following statements.

How much do you agree or disagree with the following statements:	Strongly Disagree 1	Disagree 2	Undecided 3	Agree 4	Strongly Agree 5
1. I intend to major in an engineering field	1	2	3	4	5
2. I plan to remain enrolled in an engineering major over the next semester	1	2	3	4	5
3. I think that earning a bachelors degree in engineering is a realistic goal for me	1	2	3	4	5
4. I am fully committed to getting my college degree in engineering	1	2	3	4	5

Part VI. Instructions: Many factors can either support or hinder students' college and career plans. We are interested in learning about the types of situations that could help or hinder your plans if you were to continue on in the College of Engineering. For the questions below, assume that you wanted to pursue an engineering major. Using the 1-5 scale, show how likely you believe you would be to experience each of the following situations.

If you were to major in an engineering field, how likely would you be to ...	Not at All Likely 1	A Little Likely 2	Moderately Likely 3	Quite Likely 4	Extremely Likely 5
1. Have access to a "role model" in this field (i.e., someone you can look up to and learn from by observing)	1	2	3	4	5
2. Feel support for this decision from important people in your life (e.g., teachers)	1	2	3	4	5
3. Feel that there are people "like you" in this field	1	2	3	4	5
4. Get helpful assistance from a tutor, if you felt you needed such help	1	2	3	4	5
5. Get encouragement from your friends for pursuing this major	1	2	3	4	5
6. Get helpful assistance from your advisor	1	2	3	4	5
7. Feel that your family members support this decision	1	2	3	4	5
8. Feel that close friends or relatives would be proud of you for making this decision	1	2	3	4	5
9. Have access to a "mentor" who could offer you advice and encouragement	1	2	3	4	5
10. Receive negative comments or discouragement about your major from family members	1	2	3	4	5
11. Worry that such a career path would require too much time or schooling	1	2	3	4	5
12. Feel that you don't fit in socially with other students in this major	1	2	3	4	5
13. Receive negative comments or discouragement about your major from your friends	1	2	3	4	5
14. Feel pressure from parents or other important people to change your major to some other field	1	2	3	4	5

Appendix B

SCCT Assessment Provided by Dr. Lent

Q. lentf11
The following is a list of major steps along the way to completing a degree. Please indicate how much *confidence* you have in your ability to complete each of these steps in relation to the discipline that you are most likely to pursue. Use the 0-9 scale below.

How much confidence do you have in your ability to:

No Confidence At All — Complete Confidence
1 2 3 4 5 6 7 8 9 10

1. Complete all of the "basic science" (e.g., math, biology, chemistry) requirements for your discipline with grades of B or better.
2. Excel in your discipline over the next semester.
3. Excel in your discipline over the next two semesters.
4. Complete the upper level required courses in your discipline with an overall grade point average of B or better.

Q. lentf2

Here we are interested in knowing how well you believe you could cope with each of the following barriers, or problems, that students could possibly face in pursuing a degree in your discipline. Please indicate your *<u>confidence in your ability to cope with, or solve, each of the following problem situations</u>*.

How confident are you that you could:

No Confidence At All									Complete Confidence
1	2	3	4	5	6	7	8	9	10

1. Cope with a lack of support from professors or your advisor.
2. Complete a degree despite financial pressures.
3. Continue on in your discipline even if you did not feel well-liked by your classmates or professors.
4. Find ways to overcome communication problems with professors or teaching assistants in your discipline.
5. Balance the pressures of studying for discipline courses with the desire to have free time for fun and other activities.
6. Continue on in your discipline even if you felt that, socially, the environment in your classes was not very welcoming to you.
7. Find ways to study effectively for discipline courses despite having competing demands for your time.

Q. lentf3

Using the scale below, please indicate the extent to which you agree or disagree with each of the following statements.

Graduating with a degree in your discipline will most likely allow you to:

Strongly Disagree									Strongly Agree
1	2	3	4	5	6	7	8	9	10

1. ...receive a good job offer.
2. ...earn an attractive salary.
3. ...get respect from other people.
4. ...do work that you would find satisfying.
5. ...increase your sense of self-worth.
6. ...have a career that is valued by your family.
7. ...do work that can "make a difference" in people's lives.
8. ...go into a field with high employment demand.
9. ...do exciting work.
10. ...have the right type and amount of contact with other people (i.e., "right" for you).
11. ...supervise others.

Q. lentf4

Now please indicate your degree of interest in doing each of the following activities. Use the scale to show how much interest you have in each activity.

How much interest do you have in:

Very Low Interest	Low Interest	Medium Interest	High Interest	Very High Interest
1	2	3	4	5

1. Solving practical math problems.
2. Solving computer software problems.
3. Working on a project involving principles related to your discipline.
4. Solving complicated technical problems.
5. Learning new computer applications.
6. Working on a project involving scientific concepts.

Q. lentf5

Using the scale below, indicate your level of agreement with each of the following statements.

How much do you agree or disagree with the following statements:

Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
1	2	3	4	5

1. I plan to remain enrolled in my discipline over the next semester.
2. I think that earning a bachelors degree in my discipline is a realistic goal for me.
3. I am fully committed to getting my college degree in my discipline.
4. I plan to work in my discipline after graduation.

Q. lentf6

Many factors can either support or hinder students' college and career plans. We are interested in learning about the types of situations that could help or hinder your plans. Using the scale below, show how likely you believe that you would be to experience the following situations.

In your discipline, how likely would you be to:

Not likely at all				Extremely likely
1	2	3	4	5

1. Have access to a "role model" in the discipline (i.e., someone you can look up to and learn from by observing).
2. Feel support for this decision from important people in your life.
3. Feel that there are people "like you" in the discipline.
4. Get helpful assistance from a tutor, if you felt you needed such help.
5. Get encouragement from your friends for pursuing your discipline.
6. Get helpful assistance from your advisor.
7. Feel that your family members support this decision.
8. Feel that close friends or relatives would be proud of you for making this decision.
9. Have access to a "mentor" who could offer you advice and encouragement.
10. Receive negative comments or discouragement about your discipline from family members.
11. Worry that such a career path would require too much time or schooling.
12. Feel that you don't fit in socially with other students in this discipline.
13. Receive negative comments or discouragement about your discipline from your friends.
14. Feel pressure from parents or other important people to change your discipline to some other one.

Appendix C

Survey Instrument

Computer Science Survey

Please answer each survey question as you feel TODAY.

* Required

1. Please type in your anonymous study code below: *

____

Confidence

In this section you will be rating your confidence level for each of the following statements.

2. Rate your confidence level in your ability to: Complete a pathway in Computer Science *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Complete Confidence

3. Rate your confidence level in your ability to: Complete all tasks and assignments in Introduction to Digital Technology at a high level *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Complete Confidence

4. Rate your confidence level in your ability to: Earn an average grade of B or better in a computer science pathway *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Complete Confidence

5. Rate your confidence level in your ability to: Write a computer program that uses sequencing *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All											Complete Confidence

6. Rate your confidence level in your ability to: Write a computer program that uses selection *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All											Complete Confidence

7. Rate your confidence level in your ability to: Write a computer program that uses iteration *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All											Complete Confidence

8. Rate your confidence level in your ability to: Write a computer program combining sequencing, selection, and iteration together into a single algorithm *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All											Complete Confidence

9. Rate your confidence level in your ability to: Write a computer program that correctly does what I want it to do *

Mark only one oval.

	1	2	3	4	5	6	7	8	9	10	
No Confidence At All	◯	◯	◯	◯	◯	◯	◯	◯	◯	◯	Complete Confidence

Confidence (Part 2)

In this section you will be rating your confidence level for each of the following statements.

10. Rate your level of agreement with the following statement: I am good at writing computer programs *

Mark only one oval.

	1	2	3	4	5	6	7	
Very Untrue	◯	◯	◯	◯	◯	◯	◯	Very True

11. Rate your level of agreement with the following statement: I am a good computer science student *

Mark only one oval.

	1	2	3	4	5	6	7	
Very Untrue	◯	◯	◯	◯	◯	◯	◯	Very True

12. Rate your level of agreement with the following statement: I find computer science absorbing *

Mark only one oval.

	1	2	3	4	5	6	7	
Very Untrue	◯	◯	◯	◯	◯	◯	◯	Very True

13. Rate your level of agreement with the following statement: I am very capable of becoming a programmer *

Mark only one oval.

	1	2	3	4	5	6	7	
Very Untrue	◯	◯	◯	◯	◯	◯	◯	Very True

Interest

In this section you will rate your current interest level for each of the following statements.

14. Rate your interest level in: Writing computer programs *

Mark only one oval.

	1	2	3	4	5	
Very Low Interest	◯	◯	◯	◯	◯	Very High Interest

15. Rate your interest level in: Learning to program in Python *

Mark only one oval.

	1	2	3	4	5	
Very Low Interest	◯	◯	◯	◯	◯	Very High Interest

16. Rate your interest level in: Learning to program in Java *

Mark only one oval.

	1	2	3	4	5	
Very Low Interest	◯	◯	◯	◯	◯	Very High Interest

17. Rate your interest level in: Learning more about skills you gain from a Computer Science Pathway *

Mark only one oval.

	1	2	3	4	5	
Very Low Interest	⬭	⬭	⬭	⬭	⬭	Very High Interest

18. Rate your interest level in: Completing a Computer Science pathway in high school *

Mark only one oval.

	1	2	3	4	5	
Very Low Interest	⬭	⬭	⬭	⬭	⬭	Very High Interest

Interest (Part 2)

In this section you will rate your level of agreement with the following statements.

19. I plan to take the next course in a Computer Science pathway *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	⬭	⬭	⬭	⬭	⬭	Strongly Agree

20. I plan to take all three courses in a Computer Science pathway *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	⬭	⬭	⬭	⬭	⬭	Strongly Agree

21. I plan to continue studying Computer Science beyond graduation *

Mark only one oval.

	1	2	3	4	5	
Strongly Disagree	◯	◯	◯	◯	◯	Strongly Agree

Appendix D

IRB Approval/Exemption

UNIVERSITY OF GEORGIA

Tucker Hall, Room 212
310 E. Campus Rd.
Athens, Georgia 30602
TEL 706-542-3199 | FAX 706-542-5638
IRB@uga.edu
http://research.uga.edu/hso/irb/

Human Research Protection Program

EXEMPT DETERMINATION

July 7, 2021

Dear John Mativo:

On 7/7/2021, the Human Subjects Office reviewed the following submission:

Title of Study:	Influence of Robotics on Self-Efficacy and Interest in Computer Science
Investigator:	John Mativo
Co-Investigator:	Raymond Schenk
IRB ID:	PROJECT00004137
Funding:	None
Review Category:	Exempt 1

We have determined that the proposed research is Exempt. Johns Creek High School has provided authorization to conduct research confirming that the research activities are normal educational practices, are not likely to adversely impact students' opportunity to learn required educational content, and the project does not involve disclosure of educational records. The research activities may begin 7/7/2021.

Since this study was determined to be exempt, please be aware that not all future modifications will require review by the IRB. For more information please see Appendix C of the Exempt Research Policy (https://research.uga.edu/docs/policies/compliance/hso/IRB-Exempt-Review.pdf). As noted in Section C.2., you can simply notify us of modifications that will not require review via the "Add Public Comment" activity.

A progress report will be requested prior to 7/7/2026. Before or within 30 days of the progress report due date, please submit a progress report or study closure request. Submit a progress report by navigating to the active study and selecting Progress Report. The study may be closed by selecting Create Version and choosing Close Study as the submission purpose.

In conducting this study, you are required to follow the requirements listed in the Investigator Manual (HRP-103).

Sincerely,

Kate Pavich, IRB Analyst
Human Subjects Office, University of Georgia

Appendix E

Fulton County Schools Approval

Fulton County Schools
Where Students Come First

August 9, 2021

Dear Raymond Schenk:

Your request to conduct the research study "Influence of Robotics on Self-Efficacy and Interest in Computer Science" has been approved. Enclosed is a copy of the Research Agreement. Please note that while this approval permits you to approach individual schools and/or teachers within the Fulton County School system, the final decision regarding participation is a local option and rests with each school principal and teacher. A copy of this letter must be provided to schools along with any correspondence requesting participation in this study.

No identification of Fulton County Schools (students' names, teachers' names, administrators' names, etc.) is to be included in data collected as a part of this study. Also, complete confidentiality of records must be maintained. Please remember to send a summary report once the study is complete to the address below. If any additional information or assistance is needed, please feel free to reach us at programevaluation@fultonschools.org.

If data collection continues for more than one year, you will need to complete and submit the "Research Modification / Continuation Form" (available on the DPE web page) before each additional year. This form can also be completed to request approval for changes to your data collection procedures.

We appreciate your interest in conducting research with Fulton County Schools.

Sincerely,

George Ryan Moore
Ryan Moore, Ed.D.
Executive Director – Strategy and Governance

Appendix F

Parental Consent Form

UNIVERSITY OF GEORGIA

PARENTAL CONSENT FORM

Influence on Self-Efficacy and Interest in Computer Science

You are being asked to allow your child take part in a research study. The information in this form will help you decide if you want your child to be in the study. Please ask the researcher(s) below if there is anything that is not clear or if you need more information.

Principal Investigator: *John Mativo*
Johns Creek High School

Your child is invited to be in this research study because he/she is in a class where the teacher will be evaluating methods to improve computer science participation. We want to learn more about how different activities affect interest and confidence in computer science. We are very interested in how students make decisions towards learning more about technology and how to become programmers.

During the fall semester, for one five-week period, students will be learning about the structured programming theorem, the basic building blocks of any computer program. This is part of normal Introduction to Digital Technology (IDT) content. If you agree to allow your child to be in the research study, your child will take a survey once a week for five weeks. The survey asks questions about their interest in continuing to study computer science and how confident they are about writing computer programs. Each survey will take less than 10 minutes to complete and will be completed during class at the end of each week.

Participation is voluntary. Anyone can stop at any time without penalty. While the class activities will continue, we will not collect information from or about your child to use in our research if you or your child want us to stop. The decision to take part or not to take part in the study will not affect your child's grades.

We will take steps to protect your child's privacy by replacing your child's name with a code. We will keep the list that links the code to your child's name in a separate place. We may publish articles and present the research at conferences, but we will not publicly identify your child. We do not plan to share identifiable information with anyone who is not connected to this research study. We will keep the list with names long enough to make sure we have all of the right records. Once the list with names is destroyed, we will not use or share the de-identified data for future research.

If you have any questions about the study, contact Raymond Schenk, at *schenkr@fultonschools.org*. If you have any complaints or questions about your rights as a research volunteer, contact the Institutional Review Board (IRB) at IRB@uga.edu or 706-542-3199.

If you agree to allow your child to participate in this research study, please sign below:

Raymond Schenk		
Name of Researcher	Signature	Date

Name of Child

Name of Parent/Guardian	Signature	Date

Please keep one copy and return the signed copy to the researcher.