

Dropping Out of Computer Science: A Phenomenological Study of Student Lived Experiences in Community College Computer Science

A Dissertation

Submitted to the Faculty

of

Drexel University

by

Daniel H. Gilbert-Valencia

in partial fulfillment of the

requirements for the degree

of

Doctor of Education

August 2014

UMI Number: 3641689

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 3641689

Published by ProQuest LLC (2014). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code



ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

© 2014

Daniel H. Gilbert-Valencia. All rights reserved.

Abstract

Dropping Out of Computer Science: A Phenomenological Study of Student Lived Experiences in Community College Computer Science

Daniel H. Gilbert-Valencia, Ed.D.

Drexel University, August 2014

Chairperson: Kathy D. Geller

California community colleges contribute alarmingly few computer science degree or certificate earners. While the literature shows clear K-12 impediments to CS matriculation in higher education, very little is known about the experiences of those who overcome initial impediments to CS yet do not persist through to program completion. This phenomenological study explores insights into that specific experience by interviewing underrepresented, low income, first-generation college students who began community college intending to transfer to 4-year institutions majoring in CS but switched to another field and remain enrolled or graduated. This study explores the lived experiences of students facing barriers, their avenues for developing interest in CS, and the persistence support systems they encountered, specifically looking at how students constructed their academic choice from these experiences. The growing diversity within California's population necessitates that experiences specific to underrepresented students be considered as part of this exploration. Ten semi-structured interviews and observations were conducted, transcribed and coded. Artifacts supporting student experiences were also collected. Data was analyzed through a social-constructivist lens to provide insight into experiences and how they can be navigated to create actionable

strategies for community college computer science departments wishing to increase student success.

Three major themes emerged from this research: (1) students shared pre-college characteristics; (2) faced similar challenges in college CS courses; and (3) shared similar reactions to the “work” of computer science. Results of the study included (1) CS interest development hinged on computer ownership in the home; (2) participants shared characteristics that were ideal for college success but not CS success; and (3) encounters in CS departments produced unique challenges for participants.

Though CS interest was and remains abundant, opportunities for learning programming skills before college were non-existent and there were few opportunities in college to build skills or establish a peer support networks. Recommendations for institutional leaders and further research are also provided.

This Ed.D. dissertation committee from the School of Education at Drexel University certifies that this is the approved version of the following dissertation:

Dropping Out of Computer Science: A Phenomenological Study of Student Lived Experiences in Community College Computer Science

Daniel H. Gilbert-Valencia

Committee:

Kathy D. Geller, Ph.D.

Holly Carpenter, Ph.D.

Elisa Orosco Anders, Ed.D.

Dedication

To my wife Kate who sacrificed so much. Without her support and encouragement this study would not have been.

To my sons: William, Ben, James, and Oscar.

To the folks who raised me: Amelia “Maw” Wray, Debra Fisher, Ed “Tata” Valencia, George “Paw” Wray, Irma Rister, Mary “Nana” Valencia, Randy Rister, Roland Fisher, and Wayne Gilbert.

To Rich “Smash” Cao. You’ve always answered my calls for help and I’m forever grateful to have served with you.

To all seekers.

Acknowledgments

A great many people were influential in this journey. First, many thanks to the participants of this study who gave their time and bravely spoke their truths. Their stories are the lifeblood of this study. They are the heroes of their communities and families who dared to set sail for the unknown.

A big thank you to Debbie Velasquez for providing the MESA statistics that sparked my curiosity about the topic. I'm also extremely grateful to all of the MESA directors who answered my many questions. I'm appreciative of the patience and support of Mai-Gemu Johnson, Michael Carney, Char Perlas, Connie Gonzalez, Jesse Ortiz, and especially Andy Newton who introduced me to Joe Welch. Joe took the time to point me towards *Stuck in the shallow end: Education, race, and computing*, which I read, highlighted, and re-read. Joe's work with the Computer Science and Information Technology Bachelor's Degree in 3 Years program at Hartnell College is in itself a best practice.

Thank you to my colleagues Drexel University: Bright Nichols-Stock, Catherine Kendall, Erica Wellington, David Inniss, Holly Whitworth, Kawami Evans, Marian Nichols, Linda Norman, Lottie Lynch, Lynn Martindale, Robert Shields, Todd Felts and Victor Dike. A special thanks to Phe Bach and Elena Heilman for checking in and making sure I was still moving forward. Many thanks to Drexel's outstanding faculty as well: Salvatore Falletta, Wendy Combs, Rebecca Clothey, Elizabeth Haslam, and especially José Luis Chávez.

My thanks too to my colleagues at Sacramento City College and the Los Rios Community College District. Rhonda Rios Kravitz gave much encouragement during this

year and helped me see the light at the end of the tunnel. I also wish to express gratitude for the encouragement and a willing ear to Jory Hadsell, Nicole Wooley, Norman Lorenz, Sheley Little, Tracey Valverde, and Zack Dowell.

I love my dissertation committee. I can't imagine a better group of academic guides. Thank you to Holly Carpenter for taking me on a second time and ensuring this study was something I could be proud of. Thank you Elisa Orosco Anders for listening to me chatter about this topic for nearly three years. You are a rock for many and I'm truly lucky to have you to lean on when I needed it most. Thank you Kathy Geller for your guidance as my chair. Your facilitation of this project was invaluable. You saved me from myself on numerous occasions and your wise and generous suggestions improved the drafts greatly.

And finally, it is Kate to whom I credit this study to. She endured so many brainstorming sessions and anxious moments. I owe every page of this study to her thoughtfulness, encouragement, and sacrifice.

Table of Contents

Acknowledgments.....	v
List of Tables	ix
List of Figures.....	x
Chapter 1. Introduction to the Research	1
Introduction to the Problem	1
Statement of the Problem to be Researched	5
Purpose and Significance of the Problem	5
Research Questions.....	7
Conceptual Framework.....	8
Definition of Terms	13
Assumptions and Limitations	16
Summary.....	18
Chapter 2. Literature Review.....	19
Introduction to Chapter 2.....	19
Literature Review.....	20
Summary of Chapter 2	46
Chapter 3. Research Methodology.....	48
Introduction.....	48
Research Design and Rationale	49
Site and Population	51
Research Methods	54
Ethical Considerations	61

Chapter 4. Findings, Results, and Interpretations	62
Introduction.....	62
Findings.....	64
Results and Interpretations.....	96
Summary of Findings, Results, and Interpretations.....	108
Chapter 5. Conclusions and Recommendations.....	109
Introduction.....	109
Conclusions.....	110
Recommendations.....	115
Summary.....	119
References.....	122
Appendix A: Interview Protocol.....	133
Appendix B: Observation Protocol.....	135
Appendix C: Invitation to Participate.....	136
Appendix D: Resume Template.....	138
Appendix E: Letter of Consent.....	139

List of Tables

1. Participant Demographics.....	63
----------------------------------	----

List of Figures

1. Research streams.....	11
2. Computer classes offered.....	29
3. Degrees and certificates earned.	51
4. Themes and subthemes of the study.	65
5. Participant mathematics tutoring and major selection.....	75

Chapter 1: Introduction to the Research

Introduction to the Problem

In the January 2012 State of the Union address, President Obama remarked, “Growing industries in science and technology have twice as many openings as we have workers who can do the job” (Executive Office of the President, 2012a). Though the speech suggested an overall STEM shortage, it is important to recognize that computer science (CS) positions are projected to account for 51% of all STEM job openings (Carnevale, Smith, & Melton, 2011).

Technology companies widely lament unfilled industry demand for computer science positions. Due to a lack of domestically available workers, technology companies in California stress dependence on importing foreign workers with H-1B visas. The current cap for H-1B visas is set at 85,000, though Microsoft, Facebook, and Google have recently endorsed legislation to increase the number to 300,000 (McCullagh, 2013). Offshoring CS positions, the moving of jobs and tasks from a high-cost country to lower-cost countries, has also been a widespread occurrence and a continuing threat to U.S. economic growth. Blinder (2007) suggested that computer programmers have been the number one offshored position.

Though CS jobs appear plentiful and difficult to fill, enrollments in CS courses have decreased and graduate numbers have fallen over the last 30-year period (Astin, King, & Richardson, 1981; Astin, Korn, & Berz, 1991; Pryor, Hurtado, DeAngelo, Palucki Blake, & Tran, 2011; Sax, Astin, Korn, & Mahoney, 2001). In 1980, 5.3% of college freshmen believed they would be in a computer programming related career after graduation. Though computing related technology has gained prominence in everyday

life, by 2010 only 1.7% of freshmen believed they would enter CS professions (Astin et al., 1981; Pryor et al., 2011). Female CS majors in particular have been in steady decline; 4.7% of female college freshmen chose to major in CS in 1980 while only 0.7% selected CS in 2010 (Astin et al., 1981; Pryor et al., 2011).

Community college student participation in CS majors has reflected a similar decline. Though California is a technology hub, in 2010-2011, California community colleges awarded just 1,171 computer science and information technology certificates and degrees, less than 1% of all degrees and certificates awarded statewide. Eighty-seven percent of California's 112 community colleges produce fewer than 20 CS graduates annually, with 20% accounting for the other 1151 CS graduates (California Community Colleges Chancellor's Office [CCCCO], 2012b).

A large proportion of community college students are underprepared for college-level academics, and one could infer that community college students are simply unable to tackle difficult mathematics and science courses that make up CS coursework. However, 48% of the University of California's science, technology, engineering, and mathematics (STEM) graduates began their educations at a community college (Community College League of California, 2013). This figure suggests that a substantial number of students are academically capable but may choose alternate academic subjects over CS before transferring to or after transferring to public universities.

The small numbers of computer science majors among STEM graduates is present even within specialized programs that seek to attract and support underserved STEM students in California community colleges. The Mathematics, Engineering, Science Achievement (MESA) program has a long, distinguished history of successfully

supporting and transferring underserved community college students to STEM majors at 4-year universities. Surprisingly, only 4.5% of MESA students are CS or computer engineering majors, while 21% are in the life sciences and 38% are engineering majors (CCCCO, 2011).

Little research exists to explain why community colleges do not produce more computer science graduates and certificate completers. Though robust data systems exist, data sets specific to this topic either are not collected, are not released, or may be complicated by FERPA regulations. Individual community colleges do not widely release specific statistics on students who matriculate into a major and it is difficult to identify the factors influencing the trickle of students graduating with CS associate's degrees or completing CS transfer requirements. Broad data sets contain students' plans to transfer or earn an associate's degree or certificate are available, and data are available on graduating students in specific majors, precisely how many students began their journey intending to study CS and did not persist in that program cannot be determined through public institutional reporting. This is problematic because a complete picture of how students progress through computer science or any other fields at community colleges is not provided. The goal of this study was to address this gap by focusing on students who initially matriculated as computer science majors and transferred to other fields of study; such students are not represented under the current graduation-focused data collection procedures.

The focus of this study was on students at community colleges for four reasons. The first was that community colleges were the least researched education segment. Studies targeting K-12 and 4-year colleges made up the majority of research on this topic

(Jones, 2010). The second reason was that though the most obvious factor affecting CS completions was perhaps a decline in overall CS enrollments, attrition from computer science studies presented a more actionable topic that community college administration and faculty could address (Astin et al., 1981; Pryor et al., 2011).

The third reason was the researcher's focus on the motivations behind decisions to change degree programs might have had additional significance in the context of California's community colleges. Recent recommendations by the California Community Colleges Student Success Task Force (2012) refocused community college administration on degree completion and transfer of students. Under the new guidelines, how quickly students proceeded through their programs towards completion became increasingly important. Changing majors could substantially increase time to degree, ultimately affecting the Student Success Scorecard, a California Community Colleges public performance measurement system. Understanding why students who embarked upon a course of study in computer sciences changed into other programs may help shed light on what factors could be addressed to ensure more timely program completion.

The fourth and final reason was that the U.S. computer science workforce recently was 66.7% White, 22% Asian, 4.8% Black, and 4.2% Latino (National Science Foundation, 2012). Community colleges served higher percentages of underrepresented students compared to baccalaureate institutions (CCCCO, 2012b; Green, 2006); therefore, their graduates might have a greater impact on the diversity of the CS workforce than any other segment of higher education. For California, a growing Latino population coupled with lower enrollment and performance rates of Latino students was a major concern for higher education administrators who were intent on increasing the success of this

population. Latinos represented 38.1% of the state population (U.S. Census Bureau, 2011), 33.8% of the California community college population (California Community Colleges Student Success Task Force, 2012), and 51.4% of the K-12 population (Ed-Data, 2011). The high percentage of Latino students in K-12 foreshadowed a significant increase in the Latino community college population over the next decade.

Understanding Latino student retention and persistence in CS may be key to boosting the numbers of successful computer science graduates.

Statement of the Problem to be Researched

While needs for graduates with CS degrees have escalated, the numbers of CS graduates have decreased.

Purpose and Significance of the Problem

This study was an exploration of the experiences that led underrepresented, low-income, first-generation students at California community colleges to transfer out of the computer science major into other areas of study. The purpose of this research was to study the reasons why so few students completed CS programs at community colleges and specifically to consider the experiences of the underrepresented population. Insights and identified areas of concern were gathered that may help to foster higher levels of degree completion among computer science students.

Understanding the essential experiences that students describe as germane to their decision to change from seeking degrees in CS may assist community college leadership in the identification of replicable approaches to increase student success. Administrators, faculty, and staff who are concerned about student success and retention may be able to use this work to help shape policy and practices. Students who follow the most direct

route to graduation by completing the program they initially enroll in may benefit by graduating on time. The flow-on effect has fiscal benefits for the students, the institution, and the government (Schneider & Yin, 2012).

Many studies have focused on students currently enrolled in a particular program or graduates who have successfully completed their course of study. While such studies provided rich data concerning what works, they did little to illuminate the factors that led to students failing to complete their studies in particular programs. This research differs by focusing on underrepresented, low-income, first-generation college students who transferred out of CS programs yet continued to study or to successfully complete another program at the community college or 4-year college level. Because such students either remained enrolled in the community college system or successfully transferred to four-year colleges and universities, the present study immediately excluded external factors such as access to adequate financial aid or other services and amenities at the institutional level.

A gap exists in the literature. Specifically, CS persistence and interest studies from 4-year colleges are abundant, yet few focused on community college students despite many researchers focusing on the differences between the two types of institutions (Ortiz, 2009). Ortiz surmised this to be due in part to the complex problems of studying students with a shorter program of study than those of students in traditional 4-year colleges. Two-year and 4-year institutions have different student populations; therefore, programs and services acclaimed by a study at one institution type may not necessarily be applicable for the other.

Nora (1987) noted that 2-year institutions are more likely to have a larger Latino student population, and Latino students “may have different response patterns to programs or support services” (p. 54). Ortiz (2009) noted that Latino students are more likely to require support services. California’s future is tied to its ability to educate an increasingly diverse population. Thus, the experiences of underrepresented, low income, first-generation college students were a focal point of this research.

Research Questions

The following research questions guided this study:

1. What are the experiences that lead underrepresented, low income, first-generation community college students to choose a CS major?
2. What are the experiences that lead these students to transfer out of community college CS programs?
3. What are the experiences that influence these students’ new choice of major?

Though the phenomenon of too few CS graduates is quantitatively discernable, quantitative methods cannot fully describe the complexities of the structure of the phenomenon. The use of a quantitative tool could reflect the researcher’s preconceived expectations regarding the phenomenon, and might therefore be unable to capture the essential features of the phenomenon. By using open-ended interview questions, the interview subject could introduce topics that may be outside of the researcher’s frame of reference.

Another beneficial aspect of qualitative research is the inclusiveness of outliers. Even though the phenomenon of changing majors is common, the reasons for changing

majors may not be universal. Phenomenology specifically supports the telling of a story that has yet to be told in its entirety (Moustakas, 1994). A phenomenological study best addresses the research questions by exploring the lived experiences and views of former CS students who transferred to a different course of study. Phenomenology assumes if individuals experience a phenomenon, similarities are present in the essential structures and core meanings derived from that experience (Patton, 2002). Using the qualitative approach allowed the researcher to catalog students' lived meanings and then, through interpretation, define the essential structure of the students' experience of the phenomenon (Merriam, 2002; Moustakas, 1994).

Conceptual Framework

Researcher Stance and Experiential Base

This study was approached with axiological, experiential, and ontological philosophical assumptions and a constructivist, epistemological lens. The foundation for this research rested on pragmatism. Pragmatism and the axiological assumption addressed the researcher's belief that the point of research centered on the expansion of knowledge for the improvement of society, in this case, an understanding of why California community colleges graduate so few CS students, with solutions as the quintessential goal of the research. The experiential assumption upheld the researcher's view that what is known is gained through experience. The epistemological stance acknowledged the researcher's current role in education; the researcher's profession is to serve the population studied here.

Ontologically, the researcher imagined multiple realities and believed reality to be socially constructed and therefore subject to continuous production (Berger & Luckman,

1966). The notion of what is CS is a typical example of social construction. No one thing “is” computer science. For the computer science professor, the coding and computer networking that form the basis of modern computing might be considered the foundational skills necessary to understand computers. To a young student who is more familiar with a smart phone, these elements might appear outmoded or antiquated.

The continual nature of social construction means that attitudes and perceptions of individuals and groups are ever changing. What was once cutting edge and exciting soon becomes commonplace. From this standpoint, the researcher noted that the findings of this study were influenced by factors defined in part by the socially constructed ideas and ideals prevalent in both education and computer science at the particular time of the research. The constructionist orientation assumed student knowing and decision-making in the case of major selection and persistence to be largely a product of a social process.

The computer science profession in the United States has been clearly split along racial and gender lines; therefore, various societal groups have different conversations about higher education and approaching it in different ways. Gergen (1985) reasoned, “The degree to which a given form of understanding prevails or is sustained across time is not fundamentally dependent on the empirical validity of the perspective in question, but on the vicissitudes of social processes” (p. 268). Identifying the social processes affecting CS persistence was essential to this research.

The researcher’s identity and experience contributed greatly to the conceptual framework for this study due to the “self-reflective nature” of qualitative research (Creswell, 2007, p. 3). Maxwell (2005) highlighted the value of researchers “using their own subjectivity and experience” to shape their research (p. 39).

The researcher served in a leadership capacity for 2 years implementing an American Recovery and Reinvestment Act (ARRA)-funded digital literacy program called California Connects that aimed to increase digital literacy in underrepresented populations throughout California. California Connects focused on two primary groups, one of which was underrepresented, low-income, and first-generation college students enrolled in science, technology, engineering and mathematics (STEM) majors at California community colleges. While doing this work, the researcher became interested in the educational opportunities and future prospects of this group and the expected resulting impact of STEM graduates on the increasingly technology-driven economy of California. The researcher noticed CS majors were a distinct minority among the underrepresented STEM population participating in the Mathematics, Engineering, and Science Achievement (MESA) program. This led the researcher to ponder why so few community college students majored in CS. The researcher later became a faculty member at a community college and began to cross paths with students who began their journey through higher education as CS majors, yet eventually chose an alternate path.

Research Streams

The literature review includes three research streams: (a) barriers to computer science matriculation; (b) interest development and persistence in introductory CS courses; and (c) underrepresented, low-income, first-generation college student populations in community college settings (see Figure 1). Community colleges serve higher percentages of underrepresented, low-income, and first-generation college student populations, thereby allowing research regarding groups under that umbrella.

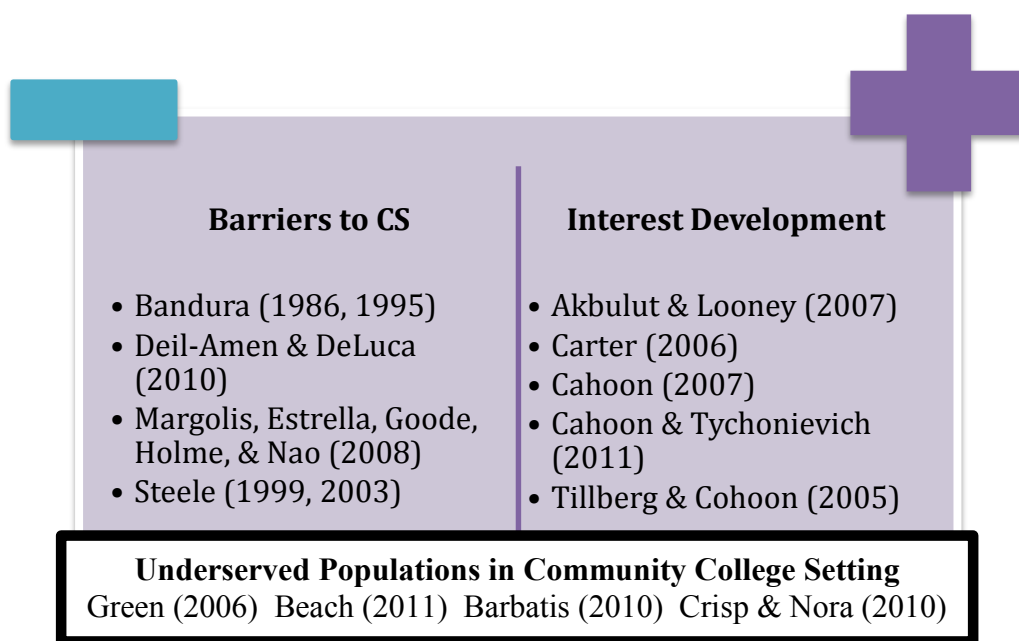


Figure 1. Research streams.

Barriers to computer science matriculation. The second stream identified the numerous socially constructed barriers to CS. The research of Margolis, Estrella, Goode, Holme, and Nao (2008) was featured prominently. Their discussion of barriers to CS for underrepresented groups revealed a denial of access to fundamental early CS experiences. The small percentages of students who were availed of CS experiences often faced other barriers if they did not fit the description of the typical CS student (White or Asian male). Steele (1999, 2003) explained how stereotype threat altered the experience of students who found themselves in the minority. The research painted a vivid picture of how the socially constructed environment worked to limit CS study for the majority of students, and specifically discouraged females and underrepresented minorities. Most importantly, the research highlighted that interest cannot appear in a vacuum; it is intrinsically related to barriers because students cannot be attracted to study a field to which they are never exposed.

Interest development and persistence in introductory CS courses. This stream explored student avenues for developing interest in CS by way of introductory courses.

Three distinct areas emerged from this stream:

1. Cahoon (2007), Cahoon and Tychonievich (2011), Cook (1997), and Akbulut and Looney (2007) found a supportive climate and format changes to be key drivers behind interest development and persistence in CS, especially in underrepresented populations.
2. Barker, McDowell, and Kalahar (2009) identified the connection between the perceived relevance to humanity and interest development. Again, this is especially relevant to underrepresented groups who were more interested in CS projects that benefited humanity.
3. Tillberg and Cahoon (2005) identified the great significance of peers in interest development and persistence. Throughout this research stream, persistence was continually interconnected to the very items that removed the barriers discussed previously. Themes involving peer support, learning communities, and collective responsibility for the learning environment emerged repeatedly, driving home the point that communities are the key to change.

Underrepresented, low-income, first-generation college student populations in community college settings. Though “80% of all underrepresented students who entered postsecondary education in the state did so through community colleges,” egregious student attrition rendered vast numbers of underrepresented students without a degree or certificate (Beach, 2011, p. 99). This final stream addressed the journey of

underrepresented students navigating community colleges and identified the student services available to support persistence. Student services played an important role (Radovic, 2010). McClenney and Waiwaiole (2005) identified many of the best practices currently institutionalized statewide through the California Community Colleges Student Success Task Force (2012), including (a) orientation courses; (b) learning communities; (c) effective advising; (d) collective responsibility among faculty, administrators, and counselors; (e) learning support beyond the classroom; and (f) hiring staff who truly cared about students.

Definition of Terms

Basic skills. Foundation skills in reading, writing, mathematics, and English as a second language (ESL), as well as learning skills and study skills necessary to complete college coursework (CCCCO, 2012a).

College prepared. Student's lowest course attempted in mathematics or English was at college level (CCCCO, 2012a). College algebra is considered a college level course. To be STEM prepared, students must be ready for a higher level of mathematics, namely Calculus.

Community colleges. Public, 2-year institutions offering post-secondary education.

Computer science (CS). Computer science coursework in higher education as well as coursework specific to the California community college, including associate's and certificate level study requiring 30 or more units. Coursework may include data processing, network architecture, programming, and hardware maintenance (CCCCO, 2012b).

Computer self-efficacy. An individual's judgment of his or her capabilities to use computers in diverse situations (Marakas, Yi, & Johnson, 1998).

CS0. CS0 is an orientation course to the CS field that introduces programming and creates a community for students (Cook, 1997; Hakimzadeh, Adaikkalavan, & Wolfer, 2011). It is offered as a general education course for non-CS majors and is sometimes used to recruit non-CS majors to the major. CS0 focuses on programming, which differentiates it from word processing or digital literacy computer courses.

CS1. CS1 is the first course CS majors take and focuses on programming. CS1 is not an appropriate introduction to the field for students without programming experience.

Faculty. Individuals who function as instructors in community college classrooms, to include both adjunct and tenure-track instructors.

Information and communications technology (ICT). ICT literacy and digital literacy are defined as “using digital technology, communications tools, and/or networks to access, manage, integrate, evaluate, and create information in order to function in a knowledge society” (International ICT Literacy Panel, 2002).

Matriculation. Student enrollment in higher education.

MESA. Mathematics, Engineering, Science Achievement (MESA) is a program in California that helps students pursue education in science, engineering, or mathematics. Most participants are low-income, first-generation college students. The program contains three branches: the MESA Schools Program for K-12 students, the MESA community college program, and the MESA engineering program at 4-year colleges (MESA, 2014).

Persistence. Students remaining enrolled in the academic institution and in good academic standing.

Remediation or developmental education sequence. “A process that begins with initial assessment and referral to remediation and ends with completion of the highest level developmental course—the course that in principle completes the student’s preparation for college-level studies” (Bailey, Jeong, & Cho, 2010, p. 2).

Pseudocode. Used in computer science textbooks to describe algorithms in natural language. Pseudocode must be translated into a specific programming language to become executable.

Socioeconomic status (SES). “A combination of social and economic factors that are used as an indicator of household income and/or opportunity” (National Center for Education Statistics, 2013).

STEM pipeline. Students who emerge from K-12 prepared to enter college without remediation in any subject (Carnevale et al., 2011).

Science, technology, engineering, and mathematics occupations (STEM). STEM occupations as discussed in this study included computer occupations (computer technicians, network technicians, programmers, computer analysts), engineering occupations, life and physical science occupations, and mathematical occupations. Medical occupations are considered part of a separate occupational group.

Stereotype threat. “The threat of being viewed through the lens of a negative stereotype, or the fear of doing something that would inadvertently confirm that stereotype” (Steele, 2003, p. 253).

Underrepresented students. Students who are racially underrepresented in higher education. The majority of underrepresented students in California are Latino. This study included a Black and a Native American participant.

Underserved students. Students who are the first in their families to attend college, are low-income, and are racially underrepresented in higher education (Green, 2006).

Visual technologies. Tools based on drag-and-drop graphical interfaces. They contain “visual representations of programming constructs, built-in UML modeling tools, graphics, the ability to create animations and programs” (Price, 2013, p. 16).

Assumptions and Limitations

Assumptions

This research was conducted from an inductive approach. The assumption was that the experiences of students participating in this study would reveal the factors that led to their decisions to enter and transfer out of computer sciences. The second assumption was that those factors were likely to be similar to those of other students who may have chosen to leave the field. The researcher focused on all racially underrepresented groups instead of a specific race, under the assumption that the findings would better serve the industry need to improve inclusiveness of the groups. The researcher also assumed the approach would lead to scalable findings.

California community college associate’s degrees and transfer level courses were generally made up of coursework focused on algorithms, discrete structures, programming languages, computer architecture, system fundamentals, and software development fundamentals. Certificates commonly available included networking,

security and information assurance, graphics design, web development, database administration, management information science, PC support, and programming. In some cases, business software skills and keyboarding were grouped under computer science as well.

Limitations

The researcher's background as a first-generation college graduate played an important role in the selection of this research topic and must be noted. During undergraduate study, the researcher was also classified as an underserved student, based his Latino ethnicity and socioeconomic level. An essential note was that the researcher began undergraduate study as a STEM major, yet instead finished an undergraduate degree in the social sciences. He later returned to community college and earned a CS certificate to further his technical knowledge and supplement his career in information and academic technology. The researcher currently served as faculty coordinator of instructional technology at a community college. Duties included the instruction of faculty with academic technology, the instruction of students with online learning and digital literacy, and participation in shared governance of information technology, educational technology, and student equity.

Each of these experiences might have rendered the researcher an insider or outsider and might have shaped his questions and data interpretation. It was essential that the researcher separate or bracket his assumptions prior to collecting data to ensure the assumptions did not cloud the researcher's ability to collect and analyze participant responses, artifacts, and field notes. The data collection plan allocated time for the researcher to do so.

This was a small qualitative study and the findings may not be generalizable to the broader population. However, research on this specific population is extremely limited and the size of this study was counterbalanced by the richness of the data.

Delimitations

This population of this study was delimited to students previously intending to transfer to 4-year colleges as computer science majors and excluded students pursuing certificates without an intention to transfer though previous statistics reported to include certificates requiring 30 or more units. Study participants additionally had to be in one of the following roles: current community college student, community college transfer student, recent graduate of a 4-year college, or community college graduate. This ensured that participants were progressing or had progressed academically and eliminated students who had left academia completely without earning a certificate or degree.

Summary

This research focused on understanding the socially constructed lived experiences that triggered underrepresented students to enter and then transfer out of CS programs in California community colleges. A better understanding of underrepresented student experiences that led to student attrition from CS programs may assist in producing best practices to alter the current underproduction of computer science associate degree graduates and transfer students at community colleges in California. Chapter 2 includes a review of the theory, research, and practice related to barriers to CS study, how students become attracted to CS study, and factors that increase affect access and persistence in underrepresented community college students.

Chapter 2: Literature Review

Introduction to Chapter 2

The Golden State (California) is often thought of as the heart of technological advancement. California's Silicon Valley attracts worldwide talent to its start-ups; "tech lords," flush with new fortunes, have celebrity status and impact worldwide business growth through angel investments and influence. Though the tech industry continues to grow and many career opportunities exist within it, the technical aspects of the work increasingly hinge on foreign labor: the United States simply does not produce enough computer science (CS) graduates to fill open positions (Carnevale et al., 2011; Executive Office of the President, 2012b).

Community colleges in particular produced very few computer science transfer students, although 48% of the University of California STEM graduates began their education at a community college (Community College League of California, 2013). This fact suggested that community college students who met the requirements for upper division STEM coursework were simply not choosing CS before their transfer to a public university. Available literature about community college computer science programs is minimal; however, research focused on K-12 and 4-year colleges and universities vividly portrayed a system that reduced the likelihood of CS as a major for most students and impacted underrepresented students to an even greater extent. The purpose of this research was to study the reasons why so few students completed CS programs at community colleges and specifically to consider the experiences of the underrepresented population. This literature review provided a lens through which to understand the story

of students in California, the barriers they might have faced, how they developed interest to matriculate into CS, and what community colleges did to help them persist.

This literature review specifically addresses community college student barriers to CS, student avenues of interest development to fields of study, and persistence of underrepresented students, first broadly, and then specifically in computer science. These three areas are most often referenced in the literature as key components of CS program completion (Bailey et al., 2010; Cheryan, Plaut, Davies, & Steele, 2009; Margolis et al., 2008); therefore, the need to study these areas as a foundation to the current research was vital.

Literature Review

Though this study specifically focused on community colleges, college students are largely influenced by their experiences in the K-12 system (Deil-Amen & DeLuca, 2010). This review begins with a discussion of the specific types of barriers students encounter before they reach the community college system. These barriers have been shown to influence a student's ability to study CS subjects without developmental education, a barrier in itself for many students (Bailey et al., 2010). This research stream was essential to understand the students who arrived at the doorsteps of community colleges with dreams of higher education and plans for a computer science major.

The second stream, interest development and persistence in introductory CS courses, addresses students who arrived at community colleges prepared to study CS or who successfully navigated developmental coursework, opening the door to CS study. As Carnevale et al. (2011) pointed out many of these individuals failed to enroll or finish STEM degrees. The avenues for developing interest in CS among academically prepared

students and the methods for retaining qualified students in specific coursework paths are explored. Finally, because California has a diverse student population and the researcher has a specific interest in the success of underrepresented students, the research considered the needs of underrepresented students at community colleges.

Barriers to Computer Science

This first stream addresses the numerous barriers influencing the California computer science pipeline: basic skills deficiencies in K-12, K-12 school environment, classroom environment, stereotype threat, home environment, early experiences, peer support, encouragement, self-efficacy, and knowledge of CS.

Basic skills. Basic skills are a prerequisite for a degree or certificate in computer science at a community college, yet historically, many students arrived at community colleges with basic skill deficiencies. More than half the college students in the United States require remedial or developmental coursework for college access (Bailey et al., 2010). In California, the situation is more severe: as many as 85% of community college students at entry have been identified as not prepared for transfer level courses (CCCCO, 2012a). This may be directly related to how the current K-12 system is arranged.

Deil-Amen and DeLuca (2010) suggested a system of three educational tracks exists in high school: (a) an academic track that provides college preparatory curriculum; (b) a vocational track that provides career and technical education (CTE) to ready participants for the workforce; and (c) a track of students that receive only general curriculum taught with lower-quality instruction, no CTE, and little if any guidance from school counselors. Deil-Amen and DeLuca (2010) placed 40% of U.S. high school students in the third track and reported it was “disproportionately composed of lower-

SES [socio-economic status], underrepresented minority, immigrant English language learners, and first-generation college students” (p. 28).

California’s population contains a larger proportion of students in the third track. In 2011, California’s K-12 schools served 1.4 million English language learners, 23.2% of the total enrollment, the majority of which (82.7%) were Spanish-speaking (California Department of Education, 2013). Over 57% of the students qualified for free or reduced lunch (Lucile Packard Foundation for Children’s Health, 2013) and over 62% of the students were underrepresented minorities (California Department of Education, 2013).

School environment. K-12 schools in California differ greatly. Low-SES schools focus on curriculum based on rote memorization of facts, and a dominant-dominated relationship exists between students and teachers. In contrast, high-SES schools offer “more unstructured and less-restricted environments where students have more autonomy and creative range, the teacher-student relationships are of mutual respect, and the curriculum is guided by problem solving and higher order thinking levels of engagement” (Gonzalez & Soltero, 2011, p. 268).

Margolis et al. (2008) identified differences while performing a mixed methods study investigating why so few Black and Latino students studied computer science at various sites within the Los Angeles Unified School District (LAUSD). The findings of the study illustrated that low-SES schools focused on basic digital literacy, desktop publishing/typing, and Internet publishing, and did not have computer science instructors. Computing courses presented a dominant-dominated teacher/student relationship, curriculum entailed step-by-step instructions for low-level thinking, and collaboration was discouraged. Computer labs were locked up or largely inaccessible outside of class.

Counselor-to-student ratios were as high as 545:1; therefore, caseload dictated that students had little interaction with counselors to plan their futures. Students interested in computer science electives were placed in alternate electives such as floristry or service due to overcrowding. Overall, faculty and administration evaluated Latino/and Black students as lacking interest in or lacking the qualities necessary to study computer science.

In contrast, the high-SES schools in this study offered a range of computing courses including AP computer science and 3-D animation and had open computer lab access and four computers in each classroom. The high-SES schools had knowledgeable CS faculty and new media internships and jobs were available via relationships with community partners. However, the advanced programming courses offered were still made up of basic assignments involving copying programs out of books, with little relevance to a career in computer science; predictably, students were disengaged by this subject matter (Margolis et al., 2008).

Such enormous distinctions in school environment may account for some of the CS pipeline issues, but not all. White and Asian females at high-SES high schools were also unlikely to take advanced computing courses. Researchers cited the classroom environment as the explanatory factor behind this situation.

Classroom environment. Classroom environment was a key factor for student success in introductory collegiate computer science courses. Wilson and Shrock (2001) determined comfort level in introductory courses as the number one predictive factor of success for undergraduate students. However, high school computer science courses discovered in the study were unwelcoming environments for females and non-White students (Margolis et al., 2008). Margolis et al. found that advanced placement (AP)

computer science courses were largely attended by “techie” White and Asian males. Conversations between the instructor and students who fit the techie description dominated class time. This was particularly irksome to female students; although they were socioeconomically on equal footing with male techies, they found the classroom environment to be intimidating. The researchers noted disrespect from tech-savvy males to other students, creating an environment in which others did not feel safe asking for help.

Fisher, Margolis, and Miller (1997) utilized an ethnographic methodology to study the experiences of undergraduate women majoring in CS and found similar incidents of female students feeling disrespected because of gender. Interviews with male students exposed strong beliefs about male and female strengths, weaknesses, and interests. West and Ross (2002) explored CS classroom practices and events; their findings indicated gender bias and male-dominated classrooms made CS appear “cold and unresponsive to female students” (p. 5).

Margolis et al. (2008) observed the “enormous amount of psychological risk” (p. 91) students experienced when taking courses outside of their support networks, as in a female attending a majority male-dominated course or a Latino student attending a majority White-dominated course. The minority participants experienced isolation and worried about being judged by classmates, and some ultimately moved out of AP/honors level courses or stayed away from advanced classes altogether. In classrooms, students received negative messages tied to race and gender from teachers, classmates, and counselors about their capabilities, and those messages became ingrained, further limiting student trajectory (Macaluso, 2010; Margolis et al., 2008).

Female students reported feeling unwelcome in stereotypical male geek spaces. Cheryan et al. (2009) examined the impact of stereotypical male geek objects (e.g., comic books, junk food, soda cans, Star Trek posters, video games, computer parts) on female and male interest in CS across four separate studies. The findings of the studies indicated stereotypical items significantly negatively influenced female student views of CS. Cheryan et al. noted that simply changing the stereotypical environment changed CS interest development.

Stereotype threat. Steele (2003) defined stereotype threat as “the threat of being viewed through the lens of a negative stereotype, or the fear of doing something that would inadvertently confirm that stereotype” (p. 253). If people cared about a specific domain, simply inserting something that reminded the subject of a known negative stereotype directed at their race affected their performance in that domain. Steele demonstrated that Black college students performed worse on a test when they thought it measured intellectual ability, with intellectual ability as a historical area of insecurity.

No ethnic group was free from stereotypes and the resultant impairment of stereotype threat (Aronson et al., 1999; Frantz, Cuddy, Burnett, Ray, & Hart, 2004; Steele, 2003). In an interview on PBS explaining his research findings relating to underrepresented students, Steele (1999) noted:

Almost every interaction can have that ambiguity to it and the threat to it, the threat that perhaps I’m being treated through that stereotype, so that students, even though they’re standing there on the same campus, in the same room with the same teacher, they’re really in very different environments. And that’s what’s been difficult for American educators to appreciate, the difference in those environments. (Steele, 1999, para. 26)

The threat was very real for underrepresented students studying computer related topics. Margolis et al. (2008) identified that in both the low- and high-SES schools, faculty believed CS interest and skill was inborn. Margolis et al. (2008) stated,

This belief in inborn qualities can have profound effects on the classroom environment. Here, it results in the propping up of students with preparatory privilege, often leaving other students riddled with insecurity and doubt, and limiting their ideas about what is possible for their own lives. (p. 85)

Computers in the home environment. A deeper dive into this gender and racial divide demonstrated that successful high school AP computer science students had commonalities. All had many computing resources at home, had the opportunity to spend most of their free time pursuing their computing interest, had parents employed in related fields or with the means to provide access to computing learning resources, and all lived near the school, thereby facilitating a peer technology learning network outside of school. The underrepresented student body was largely composed of commuter students from lower-SES areas. Margolis et al. reported,

Having insufficient technological resources at home, including out-of-date equipment that could not run necessary software or needed expensive repairs; insufficient access to a computer at home, usually because of the need to share with parents or siblings whose computing tasks might be more urgent; the inability to afford basic software like Microsoft Office Suite or peripheral equipment like a printer; and unreliable, inefficient, or slow access to the Internet. (Margolis et al., 2008, p. 81)

Low-SES populations demonstrated much lower levels of in-home computing. Only 53% of households with incomes under \$40K subscribed to broadband, compared to 84% of those with incomes between \$40K-\$80K. Latinos had the lowest adoption rates among ethnic groups (52%). Notably, only 43% of Latinos and 51% of Blacks owned and used a laptop computer with Internet access, compared to 70% of Asian and 64% of White Californians (Baldassare, Bonner, Petek, & Shrestha, 2013). These

statistics further described how many low-SES students lacked computing resources and experiences in the home.

Salinas (2008) noted additional barriers to Latino access to computing in particular, where connectivity or physical access was only the first hurdle. The Salinas study, encompassing undergraduate students, illustrated that many other items ultimately affected computer access, including the number of people sharing a computer, computer age and serviceability, and dial-up speed for the Internet. Twenty-three percent of participants reported their computer had been broken during the last 3 months, 32% had a computer that was over 5 years old, 74% shared their computer (25% had to share with 3 or more people), and 33% did not have the necessary software for schoolwork.

Lack of positive early experiences. Researchers found succeeding in computer science to be directly linked to early positive experiences with computers (Fisher et al., 1997; Taylor & Mounfield, 1994; Tillberg & Cohoon, 2005). Socioeconomic level often dictated early experience with computers in the home, as demonstrated by Baldassare et al. (2013), however, school access was often also limited (Valadez & Duran, 2007). Though digital high school legislation “provided \$1 billion over four years to supply computers and Internet access to California’s high schools” (Margolis et al., 2008, p. 29) in 1997, computer availability had not translated to access. A mixed methods study conducted by Castagnarao (2012) of 190 sixth-graders confirmed the finding; the author concluded that while computers were available in K-12 classrooms, access was limited and teachers did not use the computers very often.

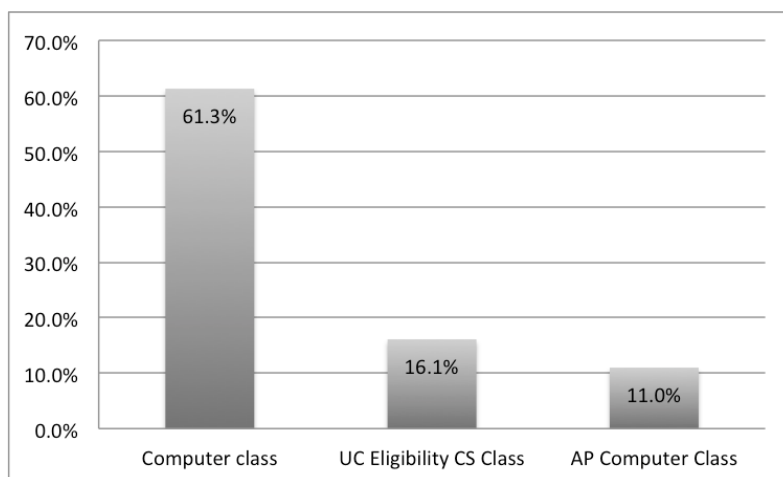
Multiple studies focused on K-12 outreach programs designed to provide precollege experiences. Such targeted programs utilized partnerships among K-12,

higher education, and often industry to deliver students hands-on experiences. The programs have been shown to positively impact student awareness and interest and to influence student major choice in higher education (Rursch, Luse, & Jacobson, 2010; Smaill, 2010). The program experiences helped students gain an understanding of the career field, develop self-efficacy in relation to the field, and develop interest to the practical applications and relevance of the field once they achieved understanding of the field (Gwinner, Prince, & Andrus, 2006; Smaill, 2010).

The majority of underrepresented students in California were not afforded such CS precollege experiences; Margolis et al. (2008) noted a lack of quantity and quality in computing courses. Statewide, the majority of high schools did not offer advanced computing courses, further limiting exposure for the majority of California's students and necessitating in-home experiences (see Figure 2).

Murphy et al. (2006) recognized the need for increased in-school computing experiences and noted females in particular benefitted from the experiences. Numerous researchers identified that female CS majors often arrived with less computing experience than males did (Barker & Garvin-Doxas, 2004; Carter, 2006; Fisher et al., 1997; Macaluso, 2010; Margolis & Fisher, 1997; Margolis, Fisher, & Miller, 2000). However, even high-SES females were less likely to report early experiences with computers than were male peers (Margolis et al., 2008). Research identified that females could catch up quickly and could succeed in CS if they had the opportunity. Findings from a study consisting of interviews with 73 graduating CS majors demonstrated that although females had significantly fewer early programming experiences compared to

males, they caught up and mastered concepts to complete on par with males (Murphy et al., 2006).



*Figure 2. Computer classes offered. Percent of California's high schools offering at least one section of CS courses by course type. Adapted from "The Computer Science Pipeline in Urban High Schools: Access to What? For Whom?" by J. Margolis, J. J. Holme, R. Estrella, J. Goode, K. Nao, & S. Stumme, 2003, *IEEE Technology and Society*, 22(3), p. 14. Copyright 2003 by IEEE.*

Lack of peer support. Peer support by way of social learning networks strongly influences self-efficacy (Rosson, Carroll, & Sinha, 2011) and has been identified as the single most important predictor of persistence in CS (Barker et al., 2009). Human learning primarily occurred in social environments (Schunk & Mullen, 2012) when peers validated learning milestones, shared information to speed up group learning, and offered observations to boost self-efficacy (Bandura, 1986). Students who lacked peer-learning opportunities related to computer science were at a greater disadvantage when positioned in a White and Asian male-dominated CS classroom, which often entailed a preformed peer support network, leaving others on the outside (Margolis et al., 2008). CS classrooms have been identified as a difficult place to establish a network out of thin air (Barker & Garvin-Doxas, 2004).

Barker and Garvin-Doxas (2004) conducted ethnographic research in 13 university level computer science courses over a 2-year span. Observations revealed an environment not conducive to peer network development; the courses contained a defensive and guarded impersonal climate and an informal student hierarchy, sorting those who belonged from those who did not by skill level. These elements in combination did not foster peer learning for students new to CS, limiting success for many.

Self-efficacy. Self-efficacy level was found to directly affect students' motivation or willingness to engage in learning a subject or performing a task (Bandura, 1995; Schunk & Mullen, 2012; Schunk & Pajares, 1997), suggesting dropout and underachievement as the result of low-self efficacy in academic tasks. Self-efficacy was also linked as a key factor behind individual career choice (Akbulut & Looney, 2007, 2009).

Females reported lower self-efficacy with computers than males, in general (Busch, 1995; Margolis & Fisher, 2001; Rosson et al., 2011; Rozell & Gardner, 2000; Shashaani, 1997); however, Beyer, Rynes, Perrault, Hay, and Haller (2003) notably found female CS majors also had lower computer self-efficacy than did male non-majors. Low-SES students and English language learners in particular also had low-levels of academic self-efficacy (Schunk & Mullen, 2012) as well as of computer self-efficacy (Vekiri, 2010). Some researchers discovered that self-efficacy could be raised with experience (Rosson et al., 2011; Rozell & Gardner, 2000); students taking just one computing course experienced an improved self-efficacy level (Shashaani, 1997).

Knowledge of CS field. Carter (2006) surveyed 836 high school calculus and pre-calculus students who would be qualified to study CS post-graduation to identify the students' interest or lack of interest in CS. Eighty percent of those surveyed had no concept of what CS majors studied; only 2% of respondents seemed to have a solid idea of the CS field. Margolis et al. (2008) identified a need to inform students as well as K-12 teachers about the CS field.

Section summary. Researchers identified numerous barriers to limit the CS pipeline: A central barrier to CS study in higher education has been successful completion of academic coursework in mathematics and science at the K-12 level. As many as 85% of California's community college students have been identified at entry as not prepared for transfer level courses (CCCCO, 2012a). Additional specific computer science barriers include school environment, classroom environment, and home environment that include lack of access to reliable technology, lack of computing courses relevant to advanced study, and low computing self-efficacy (Margolis et al., 2008).

The research exposed larger barriers to CS for underrepresented groups, specifically females and ethnic minorities. The current K-12 structure was found to deny access to fundamental early CS experiences for underrepresented students. Such barriers combined with prevalent "belief systems that rationalized this lack of access, translated—over the short and long term—into inequalities in knowledge, interest, and ultimately participation" (Margolis et al., 2008, p. 2).

Interest and Persistence in Introductory CS

The first stream identified the many barriers to CS faced by underrepresented students. While barriers to CS interest development are many, the next research stream

investigated what practices encourage student interest and help underrepresented groups overcome conventional barriers to CS. Literature discussed methods of attraction schools have employed to increase CS matriculation and persistence in introductory CS classes at 4-year colleges.

Class environment changes. While the first research stream identified CS class environment as a barrier for underrepresented students, changes to that environment positively influenced CS student interest (Barker & Garvin-Doxas, 2004). Interest and persistence were increased through the creation of a supportive climate. The foundation of the supportive climate included a reduction in anonymity through faculty use of student names, an increase in lecturer movement through the lecture hall, and increased focus on projects that included opportunities for student collaboration. Faculty efforts to increase student interactions also positively affected student persistence and improved the overall classroom environment.

Other research identified that class environment may be transformed positively by separating students by skill level. Cohoon and Tychonievich (2011) separated CS sections by prior experience level to encourage diversity of students and improve student participation among novices. The separation eliminated the ability of students with prior experience to dominate the class and make less-knowledgeable classmates feel like they did not belong.

Course format changes. The teaching pedagogy used to introduce CS has been shown to greatly impact interest development as well as persistence. Historically underrepresented groups and females responded positively to course format changes such as lowered curriculum difficulty in introductory courses, course pace, added collaboration,

and interest-based assignments (Akbulut & Loony, 2007; Cahoon, 2007; Maculoso, 2010).

Students in introductory CS courses enter with extremely different levels of previous experience (Cook, 1997; Cahoon, 2007). Akbulut and Loony (2007) identified the need for delivering content that was appropriate for an individual student. The findings indicated self-efficacy played a strong role in determining major selection. When students were first exposed to CS in academia, they needed to experience success while enjoying the learning experience. For this to occur, student perception of the content could be neither too difficult nor, conversely, too easy, to enable self-efficacy to grow and CS major selection to become more likely.

The first courses for CS majors at universities are generally two overview courses, CS1 and CS2, which exhibit high failure and low retention rates (Price, 2013). To address inexperienced groups, Cook (1997) suggested the addition of an introductory CS course (CS0) to the academic discipline. Computer course options for non-majors are generally limited to computer literacy courses. In 1997, no option existed to inform students about the field. CS0 provided an avenue for selling CS to non-majors and informing students about the possibilities for employment while introducing curriculum that is neither too difficult nor too easy (Reed, 2001). The purpose of the course was to foster peer support and provide a positive academic environment for students without programming experience. CS0 evolved over the last 15 years into different combinations of breadth (the overview of CS as described above) and depth (programming).

As technologies progressed and new scripting languages emerged, Reed (2001) called for a more balanced approach with CS0, using self-paced JavaScript tutorials to

teach novice programmers in conjunction with breadth topics to introduce the field. Findings of this combination indicated less intimidation among non-majors and increased interest in the field. Notably, Reed’s design lacked the peer support and team building opportunities utilized by Cook (1997).

Akingbade, Finley, Jackson, Patel, and Rodger (2003) continued to change CS0 by bringing in the element of fun potentially missing from the previous CS0 offerings. Their findings suggested that inclusion of student-built animations using JAWAA 2.0 scripting improved student learning outcomes. This CS0 class was specifically targeted at non-majors; hence, no mention of CS breadth appears; however, the research discussed previously highlights breadth as a necessary factor.

Attracting millennials by targeting desire for interest-focused courses is yet another way to present CS0. Research on the millennial generation highlighted a desire for personalization in coursework (Wheeler & Harris, 2008). A CS0 course with “different ‘tracks’ that students can choose from (e.g., robotics, gaming, music, mobile apps)” demonstrated increased academic performance and student retention (Haungs, Clark, Clements, & Janzen, 2012, p. 1).

To further investigate what course format changes may attract underrepresented students to CS, an introductory CS section—CS1X—was created at the University of Virginia (Cahoon, 2007). The course was designed for students with little to no prior programming experience to eliminate the possibility of intimidation by knowledgeable classmates. Student interest was gauged through surveys to determine what class examples would be used in the curriculum. The surveys indicated differing interests by gender and the course was designed to address the differences through the inclusion of a

mixture of possible projects that would be relevant for a diverse audience. The design emphasized social connections through many faculty and peer interactions, frequent hands-on mini-assignments, teaching assistant support in class, and laptops available for students who experienced technical difficulties or did not own a laptop.

Cahoon and Tychonievich (2011) noted that over a 5-year period, the university experienced increased participation in CS by underrepresented students who took the resulting CS1X course. Notably, the university experienced a 30% increase in the number of university students taking introductory course CS1. The department attracted Black students at a rate 1.2 times the national average, even though the University's percentage of Black students was only two-thirds the national average. Similarly, the department attracted female students at rate 1.6 times the national rate (18.8% versus 11.8%). "The department has seen an improvement in women, minority, and overall engineering [including CS] student retention, particularly in the first year" (Cahoon & Tychonievich, 2011, p. 1).

CS majors and non-majors alike lament the absence of creativity in introductory CS courses. Romeike (2007) provided an example of infusing creativity in an introductory high school CS course. SCRATCH, a visual technology, was used in an experimental class and traditional teacher-centered methods were used in a control class. Findings confirmed marked differences between the experimental and control classes. Students in the experimental class were 22% more likely to report that CS was fun, 64% were more likely to report CS was interesting, and 57% were more likely to view it as a creative endeavor.

Price (2013) noted the growing use of visual technologies within CS0 courses to scaffold fundamental programming concepts. The millennial generation of students currently attending college tended to be visual learners (Howles, 2007) and the technologies appealed to visual learners while enabling students to prototype and create functioning programs quickly, adding relevance to the experience. Price studied the use of visual technologies, namely ALICE and RAPTOR, in a CS1 introductory programming course. Findings indicated that visual technologies scaffolded student algorithm development, positively affecting retention. Other visual programming environments noted by Price (2013) include KODU, GREENFOOT, and SCRATCH. Mentioned as well are VISUAL LOGIC and ALGOTUTOR for algorithm development.

Ahmad (2012) analyzed an experimental course that utilized a visual technology, APP INVENTOR, a mobile development platform, to introduce programming concepts. Findings indicated students were attracted to the relevance of mobile application development so much that every student successfully passed the course. This was a key finding because CS courses historically had high failure rates (Haungs et al., 2012).

Pedagogical practices centered on active learning environments have been shown to impact persistence and retention in CS (Briggs, 2005). Peer instruction (PI), project-based learning (PBL), and team-based learning (TBL) have often been referenced to improve learning outcomes and peer networks. According to Porter, Garcia, Glick, Matusiewicz, and Taylor (2013), “PI centers on multiple-choice questions that students answer individually before discussing in small groups and answering again. This group vote is then followed by an instructor-led, class-wide discussion” (p. 1). PBL is credited with enabling students to become active learners through work on relevant projects

resulting in improved long-term retention (Kumar, 2003). TBL in CS involves two or more team members with shared objectives. This may also be referred to as collaborative-adversarial pair (CAP) programming, which involves two programmers and one computer, now commonly used for job training in software development. Briggs (2005) found that active learning helped students who were visual learners, indicating that millennials may find active learning particularly beneficial.

Relevance to humanity. Linking CS coursework to humanitarian centered problem solving positively influenced student interest (Cahoon & Tychonievich, 2011). Carter (2006) noted that among students surveyed, respondents overwhelmingly preferred to study fields that were more people-centered, and Ng and Sears (2010) found ethnically underrepresented groups were more drawn to careers that specifically served humanity. Cahoon (2007) likewise identified that underrepresented students in CS classes were often more motivated to learn topics they perceived as beneficial to society. Therefore, changing the curriculum to highlight the ways in which CS benefited humanity changed student perception of the field and increased interest.

Peers. The first stream identified a lack of peer support as a barrier, and developing peer support networks has been shown to strongly impact CS interest and persistence. Katz, Allbritton, Aronis, Wilson, and Soffa (2006) noted that females who had established a peer network in CS were more likely to persist in CS. The pedagogical changes that positively influenced peer network growth also supported student success in CS (Briggs, 2005). Barker et al. (2009) studied environmental and student factors to understand persistence in CS using a sample of 113 freshman and sophomore university students who had taken an introduction to programming course. The single strongest

predictor of persistence was student-student interaction. Specifically, students who successfully developed “peer networks within the major were more likely to remain in the major” (Barker et al., 2009, p. 4).

Peer group support, such as study groups, also increased student self-efficacy and improved persistence by helping students learn concepts they did not understand fully from the classes, grasp information that they did learn, and alleviate some of the pressures of exams by giving the students more confidence (Palmer, Maramba, & Dancy, 2011). Evidence suggests that underrepresented student persistence is impacted even more by peer relationships (Nora & Cabrera, 1996). Community college students specifically have been shown to benefit from peer learning communities (McClenney & Waiwaiole, 2005). Some community colleges have enlisted peer advisors to assist with basic needs such as preparing class schedules and finding classrooms, as well as navigating online systems.

Peers have been shown to play an extremely strong role in the recruitment of females to CS. Women identified “how coworkers, fiancés, and friends drew them to a computing major” (Tillberg & Cohoon, 2005, p. 131). Margolis and Fisher (2001) recognized that females were often extremely good at recruiting other females into CS study. Female focused CS classes and events showcasing female CS projects emerged in response to the findings.

Section summary. Though many students ultimately arrive at higher education institutions without exposure to or a basic understanding of CS, the research provided avenues for encouraging undergraduate interest in CS. Suggestions included changing the class environment to be more inviting and providing CS0 courses or CS1X courses

that were engaging, relevant, and approachable for non-majors or CS majors with limited early experiences. Additional techniques were linking curriculum to humanitarian-based projects; and creating community building and peer recruitment activities.

The research offered an encouraging representation of methods used to increase CS majors. However, it is important to point out that all available research focused on students attending 4-year universities, highlighting a need for research focused on community colleges to enable a complete picture of the CS pipeline.

Underrepresented, Low-income, First-generation College Student Populations in Community College Settings

Pinpointing the exact population of underrepresented, low-income, first-generation college students attending California's community colleges was difficult. Statistics illustrating ethnicity and income level were readily available and reflect growing majorities (Beach, 2011; CCCCO, 2012b); however, data regarding first-generation college students were not obtainable from publicly available databases. Though data were scarce on exact quantities of underrepresented, low-income, first-generation college students, the recent growth in low-income and ethnic minority students was notable. The numbers of students who qualified for the income-based Board of Governors Fee Waiver increased from 200,000 in 1992 to over 1,000,000 in 2011 (Bohn, Reyes, & Johnson, 2013). At 36%, Latino students represented the largest ethnic group attending California community colleges (CCCCO, 2012b). Low-income and ethnic minority students were less likely to have parents with college degrees (Paulsen & Griswold, 2010), and therefore, the majority of students who identified as low-income, ethnic minorities likely belonged to the underrepresented, low-income, first-

generation college student group. The next research stream first addresses the community college role in providing access to this group of college degree seekers, followed by discussion of the services that affect persistence among low-income, ethnic minority, and first-generation students.

Access. Community colleges have served the majority of underrepresented, low-income, first-generation college students: “80% of all underrepresented students who entered postsecondary education in the state did so through community colleges” (Beach, 2011, p. 99). Though such a high percentage of this student group began their journey through higher education at community colleges, most underrepresented, low-income, first-generation college students did not succeed there. The hope of open access was juxtaposed against the reality that student attrition had always been severe. In 2012, only 49% of all students completed a degree or certificate or successfully transferred to a 4-year institution within 6 years (CCCCO, 2012b). However, the averages for racial groups over the same period differed significantly—Latino and Black students fell well below the average. Though over 66% of Asian and 53% of White students succeeded, only 39.5% of Hispanic (Latino) and 39% of Black students achieved their goals (CCCCO, 2012b). Outcome differences prominently emerged along economic lines as well. Economically disadvantaged students were over 10% less likely to complete a degree or certificate in 6 years than were those with larger incomes (CCCCO, 2012b).

College preparedness or unpreparedness was a key basis for completion and outcome differences at California community colleges (CCCCO, 2012b). Though income level was associated with preparedness, ethnicity appeared to have a larger influence on student preparedness level at enrollment. The majority of community

college students arrived underprepared for college level work (Bailey et al., 2010) and low-income ethnic minorities were more likely to require remedial or developmental coursework, prolonging their time to degree (Deil-Amen & DeLuca, 2010; Horn, McCoy, Campbell, & Brock, 2009). Developmental coursework also increased services required to support persistence (Barbatis, 2010).

Though this underprepared student population required extensive services and academic support, community colleges received the lowest funding per student among state funded institutions: \$5,100 per full-time equivalent (FTE) student compared to \$6,741 at CSUs, \$6,770 at UCs, and \$7,500 for K-12 students (Bohn et al., 2013; California State University, 2012b; UC Office of the President, 2011). Community colleges served “high-need populations without the necessary resources—outcomes have been unsurprisingly low” (Beach, 2011, p. 103).

Over the last 4 years, community college state funding dipped greatly. The reductions in state funding directly affected the number of courses offered, ultimately reducing the number of students allowed to enroll. Between 2008 and 2012, the total enrollment of the California community colleges declined by almost 500,000 (Bohn et al., 2013). The budget, not preparedness or desire, ultimately curbed access as students were turned away. Underrepresented, low-income, first-generation college students without system-knowledge or social capital (Wells, 2008) are traditionally those most likely to be locked out (Wells, 2008).

Low funding coupled with high need has always been a part of California’s 1960 Master Plan for Higher Education. The plan established a tiered system using the community colleges as a cooling-out mechanism (Beach, 2011) and relegated community

colleges to the task of redirecting “aspiring students who wanted to transfer (but lacked the skills, money, or initiative to do so) into terminal students who achieved an alternative occupational credential” (Beach, 2011, p. 83). The plan relied on talented students to rise to the top; however, many underserved Californians still found themselves unable to do so and even alternative occupational credentials remained out of reach. Race continues to be the biggest predictor of college completion in California (CCCCO, 2012b).

Programs and services. Access alone has not historically enabled student success, nor has it ensured equity. Interventions are necessary to transform access into success for the growing underserved population (Barbatis, 2010; Crisp & Nora, 2010; McClenney & Waiwaiole, 2005; Radovic, 2010). Race in particular continues to emerge as an essential component affecting student success (Ortiz, 2009). Underrepresented, low-income, first-generation college students face additional barriers compared to students who are strictly low-income and first-generation students; social capital is related to race and ethnicity (Barbatis, 2010). The lack of social capital led Latino and Black students to experience “a more difficult time cultivating the relationships needed to advance their transition to college” (Ching, 2013, p. 9).

Student support services can help bridge the gap for students. Programs and services have emerged to recognize the connection between race and student persistence. Radovic (2010) discovered student services played an important role in underserved student persistence. The longitudinal study over a 3-year period in a Southern California community college indicated interaction with a community college counselor, financial

aid assistance, and EOP services to significantly improve persistence in Black and Latino students.

Ortiz (2009) discovered a significant number of students requiring special services in California were Latino/a. Ortiz (2009) explored the organizational and personal factors that assisted Latino students to persist to graduation by surveying college personnel and recent Latino graduates about their perceptions of the factors that contributed to successful completion of their courses of study. The findings suggested that developmental preparatory courses aided students in their credit-bearing academic work. Bettinger and Long (2009) performed a statistical analysis of 18- to 20-year-old students and found students who took developmental courses fared better than students of similar capability who did not take such courses. This finding contradicted studies suggesting that developmental coursework had a negative relationship to persistence and program completion (Deil-Amen & DeLuca, 2010).

Programs and services that encourage social integration positively affect underserved student persistence. Crisp and Nora (2010) studied Latino community college students specifically to determine the impact of predictor variables on persistence and transfer rates. Financial aid positively affected persistence, and greater levels of financial aid additionally improved persistence. This finding corresponded with Tinto's (1993) integration framework: when students are financially supported, they can become more socially integrated within the college. Barbatis (2010) emphasized the role of peer and family support in the social integration of students who persisted in community college. He concluded that learning communities, programs involving family members that last for the duration of enrollment, mentorship, faculty interaction, and social

activities all helped develop the social integration and increased the probability of persistence.

McClenney and Waiwaiole (2005) conducted a series of focus groups with current students as well as those who had withdrawn. The researchers found six strategies that appeared to yield improvements in student persistence. They included student success courses—intuitional data highlighted the importance of orientation and introduction courses that give students the tools and knowledge required within the institution. A compulsory orientation course offered in fall 2010 at Zane State College in Ohio, for example, equipped students “with appropriate expectations, procedural information, and heightened understanding of what is required for academic success (p. 39),” and allowed them to establish connections with faculty and other students. Other award winning institutions offer similar programs and the researchers proposed that student success courses of this nature were a valuable component for first time students.

Learning communities, student connections with each other and faculty, were the second strategy shown to be a strong factor in student success. The development of student communities to foster such connections is one way to achieve the outcome. McClenney and Waiwaiole (2005) showed an improvement in retention in all of the colleges that implemented learning communities.

The third strategy was effective advising: access to advisors and counselors that could help students navigate the sometimes daunting higher education environment is critical in helping students feel at ease and assisting them to develop achievable plans for academic success that combined their individual circumstances, goals, and priorities with knowledge about classes, course requirements, and career opportunities. Some

community colleges enlisted peer advisors to assist with basic needs such as preparing class schedules and finding classrooms as well as navigating online systems. Many of the highest achieving colleges in the McClenney and Waiwaiole (2005) study required students to attend support programs before attending academic classes.

Collective responsibility and team building formed the fourth strategy. Best practices included a comprehensive team of faculty, administrators, and counselors who maintained contact and involvement with students from enrollment past completion of their degrees. Factors identified as significant were the relationship between the above team members, an early alert system, efficient coordination of resources, a focus on outcomes related to collected data, and an awareness of students needs and concerns (McClenney & Waiwaiole, 2005).

The fifth technique was learning support. A key factor in supporting students was the awareness that learning support needed to extend beyond the classroom. Tutoring services both online and in person, computer labs, foreign language assistance, academic strategy support, and study groups are some examples. Outreach programs that connected with students who missed class and early alert and intervention programs for students who fell just short of passing courses were some of the services best practice institutions implemented to improve retention (McClenney & Waiwaiole, 2005).

Hiring the right people was the final strategy used. The relationship between staff and students was a critical factor in student retention (McClenney & Waiwaiole, 2005). Faculties whose staff demonstrated investment in their students improved the experience of the students, which in turn positively affected retention rates. The hiring practices of successful institutions took into account the fit between the institution's values and

principles and those of prospective employees. Final employment decisions were based on how well applicants appeared to fit the college culture rather than relying solely on traditional resume-based qualifications.

Section summary. Though community colleges suffer limited fiscal resources, the energized focus on accountability and efficiency shined a spotlight on underserved student persistence. Tinto's (1993) integration framework is valid for community college students (Crisp & Nora, 2010; Karp, 2010): social integration remains an essential element especially for underserved students. Services that connect the student to the academic community have been essential. Progress toward improving student outcomes cannot be achieved in isolation through various programs or initiatives, but must be an institution-wide focus if student retention is to be increased and maintained (McClenney & Waiwaiole, 2005).

Summary of Chapter 2

The research literature has shown close associations between barriers, student interest, and persistence in CS. The barriers previously identified included (a) a large basic skills gap exists for students statewide, shrinking the CS pipeline; (b) early experiences with CS have been largely non-existent for a majority of K-12 students due to school, classroom, and home environments; and (c) students who found themselves in programming classes experienced stereotype threat and lacked peer support if they did not fit the typical image of a CS student. All of these barriers worked in conjunction to limit potential student enrollment in CS. Student interest was thwarted by limited public knowledge of the CS field, paired with inaccurate perceptions. Though all of these things worked to limit CS matriculation, the research identified the fallacy of a commonly held

belief: that you must grow up dreaming in code to study CS. Students without experience can catch up quickly and promising practices to attract students have been noted, such as

- Providing CS trained K-12 teachers to increase public knowledge of the field;
- Bringing non-majors into the circle through non-intimidating CS0 and CS1X courses that introduce them to code with non-intimidating methods using visual technologies;
- Expanding student knowledge on how CS impacts other fields and can be used to help people; and last and perhaps most significantly,
- Creating community and peer-to-peer interactions that are respectful of individual diversity and acknowledge that all CS skill-levels are welcome.

Underserved students must receive services that increase their social integration on campus if persistence is to improve. With such knowledge, community colleges have a collective responsibility to create an environment that welcomes all to CS with the call, “Start where you are!”

Chapter 3: Research Methodology

Introduction

Few students complete CS degrees and certificates at community colleges. Insufficient evidence exists detailing the lived experiences of students in community college computer science programs, though the literature is full of studies focused on CS students at 4-year universities (Akbulut & Looney, 2007 & 2009; Cohoon, 2007; Cohoon & Tychonievich, 2011; Cook, 1997; Fisher et al., 1997; Frantz et al., 2004; Goode, 2010; Margolis & Fisher, 1997, 2001). In spite of the efforts of staff, faculty, and administrators at community colleges to facilitate graduation in many given fields, students make the final decision on their educational journeys. Therefore, research focused on the lived experiences and decisions of community college students who leave this academic major may be the best place to find solutions to increasing CS enrollment and completion. As noted by Locke, Spirduso, and Silverman (2013), “In any active area of inquiry, the current knowledge base is not in the library” (p. 47).

The researcher believed studying the experiences of students who did not persist in CS yet encountered success in other academic areas provided insight into why so few students completed CS degrees and certificates at community colleges. The knowledge this study revealed was evident in the described lived experiences of those who made the decision to move out of computer science. Their experiences yielded extensive insight into essential structures of the issue. Experiences specific to underserved students were a central part of this exploration due to the growing diversity within the California population.

The purpose of this research was to study the reasons why so few students completed CS programs at community colleges and specifically to consider the experiences of the underrepresented population. This study was guided by three research questions:

1. What are the experiences that lead underrepresented, low-income, first-generation community college students to choose a CS major?
2. What are the experiences that lead these students to transfer out of community college CS programs?
3. What are the experiences that influence these students' new choice of major?

This chapter contains explanations of the methodology used for this phenomenological study. Descriptions of the population, research design and rationale behind the approach, data collection methods, and data collection schedule are also in the chapter.

Research Design and Rationale

This study was an exploration of an aspect of the phenomenon of CS undergraduate underproduction at community colleges as CS students changed their majors out of the field. Transcendental phenomenological research methods were ideal for this study because they explored the lived experiences of a particular phenomenon, were interpretive, and depicted meaning for participants (Moustakas, 1994). Phenomenology readily allowed rich, descriptive data to emerge (Patton, 2002). The method facilitated an exploration of student experiences related to the environment, culture, and practices within computer science departments and brought forth participants'

socially constructed knowledge and ways of knowing about their experiences in primary, secondary, and higher education.

The researcher placed a parameter on this study to ensure participants shared the experiences of the phenomenon. This requirement included underserved students who began programs of study in CS at community colleges and attended at least one class in CS before choosing to transfer to a different major or program of study. The parameter helped ensure students who participated in the study met the formal requirement to attend community college CS classes and made their program change decision after attending one or more computer science courses at a community college.

The researcher had some CS student experience within community colleges and limited experience as a new faculty member at a community college. Before the researcher became immersed in the subject matter, every effort was made to suspend judgment and bracket any preconceived notions about CS environments and cultures in community college (Moustakas, 1994). The researcher's essence could not be removed in its entirety. Qualitative writing "is a reflection of our own interpretation based on the cultural, social, gender, class, and personal politics" (Creswell, 2007, p. 179). As a constructivist, the researcher acknowledged, "There are multiple, changing realities and that individuals have their own unique constructions of reality" (Merriam, 2009, p. 25). No single reality can be uncovered; multiple socially constructed realities emerge (Mertens, 2009).

This phenomenological study followed an inductive approach, repeated with each interview subject. The researcher began analysis with reduction or epoche, the

bracketing of his biases, experiences, perceptions, and mental models (Moustakas, 1994).

This aided the researcher to focus awareness on the phenomenon.

Site and Population

Population Description

In Fall 2012, California had 2.4 million community college students; 80% were low-income and over half were ethnic minorities (CCCCO, 2012b). The available data did not include the percentage of community college students who were first-generation college students nor did it provide information regarding student degree focus at matriculation. Still, graduation statistics were available for all public institutions of higher education (see Figure 3).

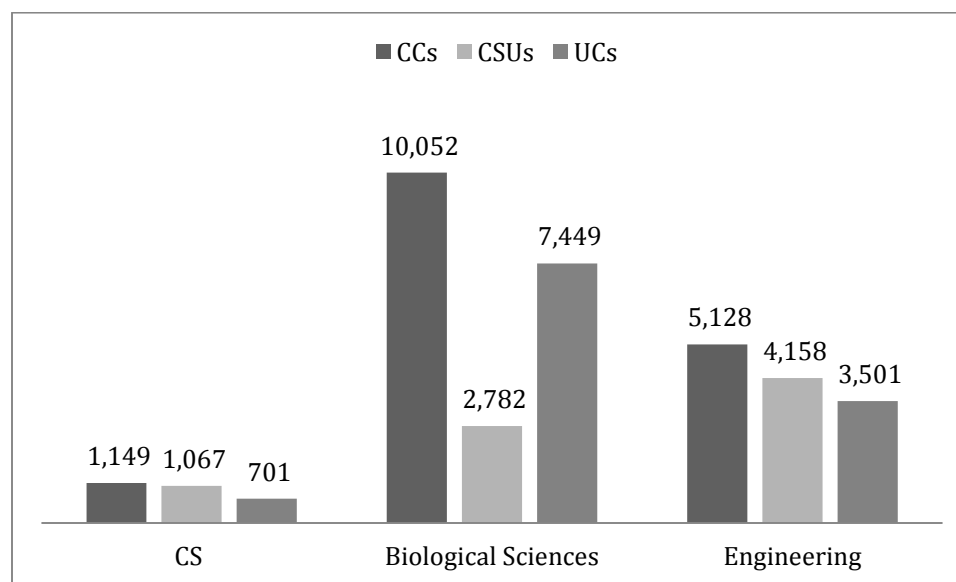


Figure 3. Degrees and certificates earned. This graph depicts graduates in computer science, biological sciences, and engineering in 2012 from California's public institutions of higher education. Community college statistics include degrees and certificates over 30 units. Biological sciences degrees earned under interdisciplinary studies are included. Adapted from "Datamart" by the California Community Colleges Chancellor's Office, 2012, "Undergraduate Degrees Granted by Campus and Discipline Division, 2011-12," by California State University, 2012a, and "Statistical Summary of Students and Staff" by University of California, 2012.

In 2012, 116,860 Californians earned community college degrees or certificates requiring more than 30 units; only 1,149—less than 1% of graduates—majored in CS. The CSU system fared only slightly better; of its 76,427 graduates, 1.4% majored in CS. The UC system performed similarly, with 1.43% of its 48,899 graduates completing a CS degree. The two most popular STEM fields at all three systems were biological sciences and engineering.

The study population was composed of 10 underserved students who originated their studies in community college CS programs before changing their programs of study. For the purposes of this study, underserved students were those who were the first in their families to attend college, were low-income, and were racially underrepresented in higher education (Green, 2006). The amount of higher education of the participants varied; participants were in one of the following roles: current community college student, community college transfer student, or recent graduate of a 4-year college. Given the sampling approach, no attempts were made in this study to control for age, gender, or ethnic variance among the participants. Among the 10 participants, 8 were Latino, 1 was Black, and 1 was Native American. Nine were male and 1 was female, and the participant age range was 20-28. All participants attended high school and college in California. Four attended college in Southern California and six attended in Northern California.

As mentioned previously, statewide data on student matriculation into CS were not available and CS graduation statistics, though available, did not report gender, race, or SES level. Furthermore, statewide data detailing community college transfer student success were incomplete. Many students transferred without obtaining a certificate or

associate's degree, and until recently, the student outcome dataset reflected such students as dropouts. The dataset is dependent on the colleges to report accurate student outcomes and completion of a certificate or degree creates an easy to identify data point.

In 2010, California Senate Bill 1440 instituted an associate's degree for transfer to simplify the transfer process for community college students transferring to the University of California and California State University systems. Transfer students who earned an associate's degree for transfer could be reflected in the dataset. However, though students were encouraged to complete an associate's degree prior to transfer, there was no guarantee that graduation would occur. Until the datasets of all academic institutions nationally at all levels are linked or combined, tracking student movement among the various institutions will remain difficult and produce an incomplete picture of student success.

Site Description

California community colleges are 2-year, publically funded institutions. Boards and administrative leadership teams locally govern each of California's 112 community colleges. Some colleges function as stand-alone entities while others belong to community college districts consisting of multiple colleges. The California Community College Chancellor's Office and Board of Governors act as sources of leadership, advocacy, and support for districts and campuses.

In Fall 2012, California had 30,442 full-time equivalent community college faculty positions made up of both tenure-track and temporary positions. Computer science faculty accounted for 833.9 of those positions (CCCCO, 2012c). Out of 112

campuses, 108 offered computer science degrees or certificates requiring 30 or more units. The remaining four campuses offered only basic computer literacy courses.

Computer science curriculum guides had been available since 1978 and were updated every 10 years through the Association for Computing Machinery (ACM) Two-Year College Education Committee, making standardization possible. But while a high-level of similarity may be present, substantial differences arise in terms of how local campuses interpret the courses.

Computer science departments in general are influenced by college leadership, and in some cases by district leadership, staff and faculty, the local environment, and the demographics of the students they serve. The variety of influences produces great differences among the individual community college districts, campuses, and satellite sites, which may influence students' experiences at their individual campus. This research did not address concerns about the sites themselves, but instead focused on student experience in relation to their studies in the field of computer science.

Specific site access was not sought and participants came from a range of campuses. Some had already graduated while others had transferred to other institutions. Interviews did not take place in community college locations. The researcher obtained approval from the Drexel University Institutional Review Board (IRB).

Research Methods

Description of Methods Used

This study included use of multiple methods to collect data. Methods used included (a) face-to-face, individual, semi-structured interviews, (b) field notes based on observations recorded by the researcher during interviews, and (c) artifacts.

Interviews with former CS students. Individual, face-to-face, semi-structured interviews using a 10-item protocol took place with a purposive sample of 10 community college students, transfer students, or graduates who had previously taken courses in computer science while majoring in computer science and later transferred into other subject areas while at a community college or after transfer to a 4-year college in California. The intent of the interviews was to discover student experiences and perceptions that led to the decisions to transfer to another program, thereby allowing researcher reflection on the meanings behind the perceptions of such experiences (Moustakas, 1994).

Through the interviews, the researcher identified barriers to student persistence that could be addressed at the institutional level, including the impact of policy and support services and learning environments. The researcher incorporated any experiences identified by the research participants that extended beyond the community college, such as student perceptions of career possibilities, educational preparedness before entering the community college system, or a general misalignment with their ideas of CS as a field. Structural statements, observations, and reflections of the researcher's account of the phenomenon were recorded in a researcher's journal and artifacts were collected.

Instrument. An interview protocol (Appendix A) containing standardized open-ended questions was used as the foundation for the semi-structured interviews. The protocol questions formed the basis of the interviews, and further probing questions helped to gain further insights and thick rich descriptions. Individual interviews with study participants were conducted at a mutually agreed upon time and location, not on a

community college campus, and scheduled for one hour. The goal of this approach was to elicit as much information as possible that the respondents found relevant without tainting the responses by the researcher's preconceptions or bias.

The researcher utilized the observation protocol form (see Appendix B) to take notes during interviews and to document observations such as non-verbal cues or specifics about the interview location, if relevant to the discourse. The researcher also utilized the observation protocol form post-interview for reflective notes when additional context was available upon reflection.

Participants. Participants were gathered using word-of-mouth and direct solicitation of colleagues serving former computer science students from some of the 108 California community college campuses offering CS degrees and certificates. Purposeful sampling was used to ensure the study participants were “individuals who have all experienced the same phenomenon in question” (Creswell, 2007, p. 62). The researcher contacted colleagues at community colleges and 4-year colleges or universities verbally in person or by telephone. Colleagues were asked to share the recruitment e-mail (Appendix C) with potential participants. All potential participants were first contacted through e-mail or phone, depending on the contact information available.

To participate in the study, participants had to have been enrolled in CS and attended at least one class before transferring to another academic program, had to be a member of the underserved student population, and had to meet one of the following criteria. The participant needed to be a current community college student in California, or needed to have graduated from a community college in California, or needed to have transferred from a community college in California to a 4-year college in California.

Participants were gathered using direct solicitation of colleagues serving former computer science students from some of the 108 California community college campuses offering CS degrees and certificates. Purposeful sampling was used to ensure the study participants were “individuals who have all experienced the same phenomenon in question” (Creswell, 2007, p. 62). The researcher contacted colleagues at community colleges and 4-year colleges or universities verbally, either in person or by telephone. Colleagues were asked to share the recruitment e-mail (Appendix C), which included the researcher’s contact information, with potential participants. Participants subsequently contacted the researcher to indicate their desire to participate in the study.

The researcher sent an invitation letter (Appendix C) to potential participants via their professional e-mail address. Respondents were asked to indicate their willingness to participate in an in-depth interview discussion and submit a resume. Respondents who self-identified as willing to participate were contacted directly by the researcher by phone to review the specifics of the study, including an overview of the consensus process. All participants were advised of the voluntary nature of the study, participant confidentiality, and the ability to withdraw from the study at any time. The elements of the consensus form were discussed, the participants were asked to confirm their continuing interest, and the interview was scheduled for a time and location.

Data collection. The semi-structured, face-to-face interviews were video recorded with Camtasia software and an external microphone or digitally recorded on a digital audio recording device. A secondary hand-held audio recording device was also utilized to ensure back-up to the collected data. The researcher transcribed the interviews and

field notes and reviewed them in full to ensure that vocal intonations were noted correctly and field notes matched appropriately.

All electronic data were downloaded to and maintained on a separate encrypted and password-protected drive without Internet access. During analysis, hardcopy data was maintained in a locked desk drawer by the co-investigator. Both electronic and hardcopy data collected will be retained by the principal investigator in a locked cabinet on Drexel premises, aligning with the IRB policy.

Artifacts from students and college public websites. The artifacts included resumes or curriculum vitae (CVs) freely provided by the student as well as information retrieved from publicly available college websites. These artifacts provided insight into the students' lived experiences. Respondents who willingly participated in face-to-face interviews were asked to contribute a resume or CV to further convey their experiences. Participants were provided with a template (Appendix D) to aid in consistency. The researcher analyzed publicly available college websites to cross-reference student course and program information. All electronic data were maintained on an external drive, encrypted, and password-protected. All hardcopy data were maintained in a locked desk drawer according to Drexel IRB instructions.

Researcher field notes and journal. The researcher maintained field notes and a journal to catalog his own structural statements and textural-structural statements of interviewee revelations pertaining to the specific phenomenon (Moustakas, 1994). All electronic data were maintained on an external drive, encrypted, and password-protected. All hardcopy data were maintained in a locked desk drawer. All documentation collected was retained until the conclusion of the study.

Data Analysis Procedures

Data analysis consisted of several steps. The first was a close reading of the textual narratives of raw interview data detailing participants' experiences, followed by analysis of journal notes. Open coding and gathering the reduction of the information into themes was next, and then returning the raw and interpreted data to the participants for their review (Merriam, 2002). The final step was the horizontalization of clusters of emerging themes by similarities (Moustakas, 1994).

This phenomenological study resulted in collection of data representing the subjective compilation of experiences and recollections of the subjects as well as the researcher. The researcher cannot be removed from the analysis in entirety. The researcher's role in interpretive qualitative research was essential. As Merriam (2009) stated, "Qualitative researchers are the primary instruments for data collection and analysis, and interpretations of reality are accessed directly through observations and interviews" (p. 29).

Data analysis followed Moustakas' (1994) transcendental phenomenological methods. Like the interview process, data analysis began with reduction or epoche, where the researcher bracketed his biases, experiences, perceptions, and mental models (Moustakas, 1994). The bracketing process helped the researcher open his mind and cast away what he believed. This process facilitated a new openness as the researcher focused his awareness onto the phenomenon. The next steps were a close reading of the textual narratives of raw interview data detailing participants' experiences, analysis of journal notes, open coding, and a reduction of the information gathered into themes. The data were cross-referenced to reflect commonalities across interview subjects.

The interview transcripts, observation field notes including non-verbal cues and reflective notes, and artifact protocol forms were “deductively analyzed to identify the recurring patterns or common themes that cut across the data” (Merriam, 2002, p. 6). This step was where the horizontalization of clusters of emerging themes by similarities occurred (Moustakas, 1994). Research software was used to assist the researcher with data management and theme identification. The software did not code the data, but provided a format to facilitate the coding process.

Stages of Data Collection

Data collection proceeded in stages after receiving approval from the Drexel University IRB. The researcher relied on colleagues within community colleges and 4-year colleges or universities to identify and contact appropriate potential participants. Participants were selected on a first-come, first-serve basis. To maximize internal validity, the researcher selected individuals with whom he had no prior history of conversations regarding CS. The subjects identified were contacted via e-mail or face-to-face and given a copy of the invitation (Appendix C) to study prior to conducting interviews.

Face-to-face, individual, semi-structured interviews conducted with former computer science students from a selection of California Community Colleges took place at a mutually agreed upon time and location. The interviews lasted up to one hour. Field notes were utilized to collect interview observations such as non-verbal communications. An observation protocol was used to record descriptive and reflective notes (see Appendix B). The multiple methods of data collection and analysis used in this study acted together to explore the lived experiences of students who had left the CS major.

Ethical Considerations

To ensure adherence to ethical procedures, the researcher sought permission to proceed from the Drexel University IRB. The researcher followed the guidelines of the IRB to ensure the rights of the participants were respected and that no participant was put at risk through participation in the research. All study participants received a consent form outlining their rights as voluntary participants, including their right to skip any question and to opt-out at any time. To protect privacy, participants remained anonymous and responses and information identifying their institutions were generalized to ensure confidentiality. Participants had assigned pseudonyms to further protect their identities. Findings were aggregated by themes for presentation to prevent identification of any individuals. Every effort was made to ensure findings could not be linked directly to individuals or specific colleges.

Chapter 4: Findings, Results, and Interpretations

Introduction

This study was an exploration of the experiences that led underrepresented, low-income, first-generation students at California community colleges to enter and then transfer out of a computer science (CS) major into other areas of study. This phenomenological study was designed to explore answers to the following research questions:

1. What are the experiences that lead underrepresented, low-income, first-generation college students to choose a CS major?
2. What are the experiences that lead these students to transfer out of CS programs?
3. What are the experiences that influence these students' new choice of major?

Chapter 4 contains the findings, results, and interpretations of the research. Data were gathered through an analysis of semi-structured personal interviews (Appendix A), a review of artifacts provided by participants (Appendix B), and integration of observations from the researcher's journal.

Participant Demographics

Purposeful sampling was used to ensure the study participants were “individuals who have all experienced the same phenomenon in question” (Creswell, 2007, p. 62). The participants had been enrolled in a community college computer science academic major and attended at least one class before transferring to another academic program. They were also members of an underserved student population (underrepresented, low-

income, first-generation college students). Last, they met one of the following criteria: (a) a current community college student in California, (b) a graduate from a community college in California, or (c) a transfer student from a community college in California to a 4-year college in California. Ten individuals were chosen to participate in the study (see Table 1).

Table 1

Participant Demographics

Participant Pseudonym	Status	Major	Ethnicity	Orgs.	No. of CS Courses	College GPA	HS GPA
George	Community College Student	Library Science AA in process	Native American	None	1	3.2	2.4
Sandra	Graduate 4-year College/University	BS Engineering	Latina	SHPE MESA SWE	8	3.5	3.9
Raul	Graduate 4-year College/University	BFA Art	Latino	MESA	2	3.2	3.9
Peter	Community College Graduate	Certificate Network Admin.	Latino	MESA	8	3.2	3.7
Elias	Graduate 4-year College/University	BS Biology	Latino	MESA	2	2.8	3.4
Shawn	Graduate 4-year College/University	BS Game Design	Black	NSBE	8	3.4	4.0
Lorenzo	Graduate 4-year College/University	BS Game Design	Latino	MESA	4	3.2	3.8
Manuel	Graduate 4-year College/University	BA Community Studies	Latino	MeChA	2	3.2	3.8
Sam	Community College Graduate	AS Science	Latino	None	3	3.6	3.8
Charlie	Graduate 4-year College/University	BA English	Latino	None	4	3.8	3.5

Among the 10 participants chosen, nine were male and one was female, and their ages were 20-28. Seven completed a bachelor's degree from a 4-year college or

university, one completed an associate's degree, one completed a certificate, and one participant was still attending a California community college. The average self-reported high school GPA for participants was 3.62. The average self-reported college GPA was 3.31.

Though the California community college CS student population might have had a male majority, statistics were not publically available to verify whether the study sample was similar to the actual population of students who select CS as a major at California community colleges. All 10 study participants received pseudonyms to ensure their privacy. Details of the participants' academic status, major, ethnicity, number of computer science courses completed, and college GPA are in Table 1.

Findings

Findings are demonstrated through a trail of evidence, using excerpts from interview transcripts supplemented with the researcher's observations, reflective notes, and artifact analysis (Bloomberg & Volpe, 2008). The data coding and the subsequent horizontalization of clusters by similarities produced three major themes: (a) pre-college characteristics; (b) challenges in college CS courses; and (c) reactions to the work of computer science. Within each theme, multiple sub-themes emerged (see Figure 4). The first major theme, pre-college characteristics, examined participants' relevant pre-college commonalities. Though participants originated from high schools and communities across the state, significant commonalities emerged. The second theme addressed the common challenges in CS as declared by participants. Many struggles were germane to most participants such as struggles in mathematics and a lack of CS tutoring. Finally, the third stream dives deeper into the specific reactions to the work of CS. Participants

overwhelmingly shared similar longings for items missing from CS. Many of these items were ultimately located in participants' new fields of study.

Pre-college Characteristics

Study participants appeared to share many common pre-college characteristics. Four key characteristics emerged: (a) an early love of computers fostered by access to a computer in the home; (b) self-description as the most highly skilled computer user amongst peers and family; (c) attendance at high schools where no programming coursework was offered; and (d) status as a high achievers in high school (nine of the 10 participants).

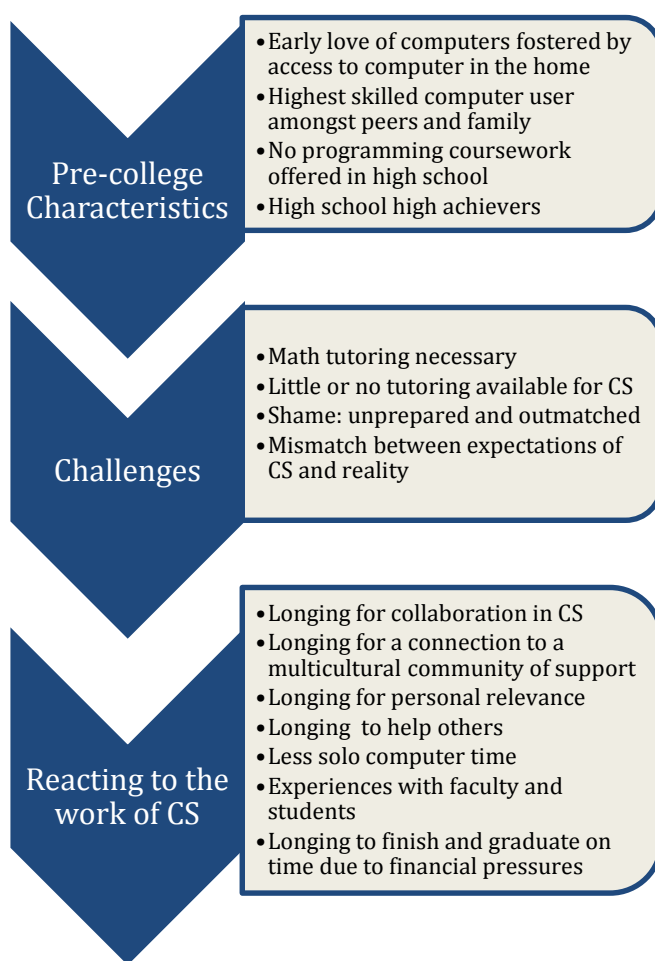


Figure 4. Themes and subthemes of the study.

Early love of computers fostered by access to a computer in the home. Study participants universally identified early access to computers in the home. Three participants had a computer in the home for “as long as they could remember.” Three acquired their first home computer in junior high school, and four participants acquired one in high school. Charlie explained, “We were pretty much the first people I knew to get a computer. It was super-expensive for my family—a Mac. I don’t even know how my mom afforded it.” Participants universally exhibited excitement when they discussed their early experiences in the home with computers. Raul noted,

I purchased a computer in 2001 [9th grade] . . . I assembled it myself and, you know, I really had this general interest in technology. And that sort of got me started and I just did these things on the side and learned on my own.

Peter confirmed, “What sparked my interest, was, I was a junior in high school and I just got a brand-new computer system.” When asked if he had a home computer while he was a child, Manuel raised his voice excitedly and leaned in.

Yeah, that’s what got me interested in it! There is so much, like, programs that are out there and I always kind of had a good time using LimeWire and playing video games and looking for other things to do with computers. Seeing that it was something fun for me to do—I like to take them apart too! I don’t know, it kind of just boggled my mind—how I started running the software to get programs going. And I had issues with my computer, so I would always have to like, restart it and fix it to work with it as well.

Early access to computers offered an experience that framed the initial excitement for all participants. This excitement led to a drive to experiment and explore the computer functions.

Most highly skilled computer user among peers and family. All 10 study participants self-identified as the go-to person for tech support for their extended families and peer groups. George commented,

I had always learned tech more easily because it was just something that we were able to grow up with . . . and so when my family had computer products or anything tech related, I was the always the one that they went to. I was the oldest child and naturally I was the first one to adopt everything and know how to use these devices . . . I had a younger sister and I have several cousins, so I always helped them out.

Sam described his role,

Anytime something would happen, my family or my cousins or friends would always come to me with their problems and I would be the one taking it apart, or if I didn't know anything, I would look it up and research how to do it, and with that I would fix it, take it apart, or buy parts if needed.

The researcher delved further into these phenomena with participants. All identified an extreme interest in computers they developed by their junior year of high school. Participants believed this interest was obvious to any who knew them. Sam explained it nicely,

Well, I would always express my love for computers. If someone came to our house, I would say, "Hey, this is what I learned today," and then they would see that I understand this kind of stuff. . . . So when they had problems, they would instantly say, "Oh, I will go to Sam."

The position as lead family and peer tech support often continued into the present for most program participants. Manuel spoke about his experience providing technical support to his family.

For the most part, I still am [tech support]. My mom recently just got a laptop, because they had the computer that we had since '98. It was like a dinosaur, and after I went to college in 2005, you know, they pretty much didn't use a computer or anything. And now they have a laptop that they recently got a year ago and they come to me about that. And my sister, too. She talks to me, when she has issues with it; she talks to me as well. Especially with technology stuff, they always ask me about it.

Most participants had early experiences taking apart the hardware components and troubleshooting their own computers, and three mentioned the reason was financial necessity. Sandra was the person among her peer group known for fixing problems. She

commented, “I knew how to figure out how to fix any problems that anybody had, so people would usually ask me to fix things.” George had similar experiences and he linked those directly to his decision to select computer science as his major.

I had built computers. I have done tech support, so it seemed like the right thing to do. . . . When my family had computer products or anything tech-related, I was always the one that they went to. I was the oldest child, and naturally, I was the first one to adopt everything and know how to use these devices.

Conversely, one study participant described how his interest in trouble-shooting computers led to his father’s disapproval.

I’d always fix my dad’s computer [when I was a kid], and then he would download free music programs and download viruses onto his computer, and I would fix it. He didn’t understand. I would get in trouble and he would yell at me and complain that I deleted all of his music. He just didn’t understand that he was downloading viruses. He forbids me from getting on his computer because he keeps saying that I deleted all his music. I just manually quarantined all the stuff that was bad and he cried that I deleted all of his music worth hundreds of dollars, which he got off the Internet for free! (Elias)

Raul found his love for computers was a good way to supplement his family’s income.

I was really interested with computers. I still [am], actually. In high school, I would fix computers. I would add RAM or take away RAM and the motherboards. Um, what else would I do? I would replace laptop broken screens. Um, so I was really into, like, fixing computers. You know, I remember, like, downloading hacking tools and messing around with programs.

Peter was able to expand his knowledge by working for his high school network support staff:

Everyone comes to me for computer work in my family. So I’m the first. Yeah, nobody in my family ever had any interest in computers. . . . During the summer, I started working for our school network technician and this guy taught me everything about fixing computers, running network cables, building small networks, and even some server stuff. And, like, from his knowledge and working with him and then taking the [computer repair] class through ROP, that was just my skills and my toolset that I took into college, and I built upon that as I went along.

The experience of being the computer expert among friends and family increased the computer self-efficacy for all participants. All exhibited a visible pride when they spoke about their expert status.

No programming coursework offered in high school. While high schools in some school districts offer computer-programming coursework, none of the high schools of the 10 study participants offered such courses. While some offered computer courses, the courses were described as word processing, introduction to the Internet, or general Microsoft Office courses. One participant reported the availability of an introduction to web design course based on HTML; this was the most advanced offering among the participants. Sandra stated, “They were pretty easy for me but we didn’t do any programming. They were all pretty basic.” Manuel had only a typing class at his high school. Sam was placed in the web design class where he excitedly learned how to build web pages. He cheerfully announced, “I got hooked on it!”

All participants reported a need for more advanced curriculum offerings. Sandra commented that she was excited to take an introduction to computers course at her high school but was disappointed that they “mostly did the Microsoft Office suite.”

Raul linked his difficulty in college level programming courses directly with this lack of computer science curriculum at the high school level. While he took all of the classes his high school offered, they were computer literacy courses. “The only class that there was, was like, learning how to surf the web and basic stuff like, that but never much more beyond basic PowerPoint. You know, I was all self-taught at that time.” He was able to gain HTML scripting experience through participation in the high school MESA program. Raul reasoned,

MESA was really targeting computer science and one of my steppingstones was web design, so I really like the idea of coding, you know, but when I went to college and I was taking classes, I found it really difficult and challenging taking programming.

Two participants were exposed to computer maintenance coursework in high school, which led to some confusion around the definition of computer science.

According to Peter,

We call it programming, but no, I never took the actual computer science courses. The only thing I took computer-related for sure would've been in ROP, the regional occupation program, and they offered computer maintenance and repair my senior year.

Charlie, likewise, took a yearlong, double class period ROP course during his senior year of high school, where he learned electronics troubleshooting and maintenance.

Two other participants found their niche in audiovisual (A/V) or graphic arts courses. Their love of technology drew them to where they could use video and image editing software with computers. Lorenzo had video editing courses that he enjoyed, which contributed to his desire to major in computer science. George's work as an A/V camera operator in high school contributed to his love for technology and interest in pursuing computer science. The lack of programming courses available at high school coupled with the availability of computer literacy courses, A/V, and computer or electronics maintenance courses left all participants with an enthusiasm for technology that influenced their choice in college majors.

High school high achievers. Among participants, nine attended low-SES public high schools, with eight graduating at the top of their class. One attended a private high

school on an academic scholarship, earned after exhibiting high academic potential in middle school. He, likewise, excelled in high school among his high-SES peers.

Though a large proportion of community college students are underprepared for college, of these 10 participants, all but one was immediately placed in college-level academics upon enrolling in community college. Nine of the 10 participants had taken at least one AP course and eight participants self-reported that they believed they were extremely well prepared for college-level academics. Sandra commented, “I felt pretty confident when I got to college. I took all of the college prep classes in high school. Most of the college courses were no problem for me.” Charlie echoed this experience.

Raul said, “Oh, definitely, [I] was well-prepared academically.” Sam felt his high school experience unquestionably prepared him for college level work. He explained, “Because all the stuff that I studied in high school, I had to retake it at college. I even kept some of the stuff [materials] from high school so that I could reference it in college.” Shawn said, “I felt like I had pretty good preparation for college. I was never too worried about that.” Conversely, Manuel had a high GPA and graduated at the top of his high school class, but did not feel prepared.

I didn't feel very well prepared. Because, um, I didn't know what it was. I was always good with asking teachers what I need to get done to get a better grade. So I don't know if that came into play as far as me getting all really good grades, and I'm not sure how the efforts of the other students at my school were. I felt like the teachers did the best they could, but I don't feel like it prepared me very well for, you know, getting a college education. I know when I went to college, I was like, whoa, what the heck is this!

Though nine of 10 participants graduated at the top of their classes, one did not. The student who performed lowest in high school shared experiences in college computer science courses similar to those of higher performing peers. However, he completed

fewer CS courses than peers did before selecting new majors. He also spoke of encountering more difficulty in college mathematics.

Theme summary. A computer in the home, paired with a genuine interest in technology, led the participants to achieve the designation of family and peer tech expert. Entry-level technology courses at high schools continued to foster and encourage the participants' interest in technology. Most strived academically and felt prepared for college, but a lack of programming coursework or programming experience foreshadowed the challenges that followed in college.

Challenges in College

As study participants described their challenges as CS majors, four issues appeared repeatedly. The participants reported (a) math tutoring was necessary to complete college-level mathematics, (b) little or no tutoring was available for CS, (c) feelings of shame about CS preparation compared to classmates, and (d) a mismatch between their expectations and the reality of CS curriculum.

Math tutoring necessary. All but two participants reported earning A's and B's in high school mathematics courses and feeling a high level of confidence in their mathematics skills. However, they noted that college mathematics was more difficult than they anticipated and seven of the participants required tutoring support in college. Sandra struggled in college math after doing well in high school calculus, but with tutoring support she says, "I did ok." Lorenzo said,

I used to love math in high school, in geometry and algebra and all that stuff, I loved it. I had A's, but when I got to pre-calc in college . . . it was just very hard. I hit it at the wrong time in my life. And it never, uh, it just, uh . . . never clicked.

Manuel took calculus in high school and earned an A, but he did not realize until his first year of college that his high school math preparation was insufficient.

I remember I took an AP calculus class in high school, but our teacher was just reading off the book when he was teaching. I guess he knew how to maybe teach it or not. I'm not even sure, because at that time, he was just doing that. All my friends got together and we would do the homework and help each other out. You know, we didn't really learn much. It was just, like, so confusing.

Fortunately, the tutoring offered at Manuel's college enabled him to make it through both calculus and multivariable calculus.

Lorenzo left computer science because of his math struggles. He participated in a math-tutoring lab, staffed by a math professor, specifically for underrepresented students. The math-tutoring labs for other students were staffed by student tutors and this one was considered superior because of the addition of a tenured professor. Lorenzo appeared to feel disturbed as he conveyed his experience. His entire disposition shifted as he shared.

I was doing really bad at math, and she said to me, "Maybe you should consider switching majors. It seems like it's just over your head." And I'll never forget that, when she said, "It seems like it's just over your head. Maybe you just should be somewhere else, you should think about that." She suggested that.

Raul's confidence was also high leaving high school.

I was pretty decent in math, you know, but yeah, once I hit college and I started taking math classes. It was a pre-rec to calculus and I remember earning a C in that pre-calc class. That made me a little scared of pursuing the computer science track.

Four participants exhibited surprise at the requirement of advanced mathematics for computer science. Peter thought that the math required for the major would be easier than it was, and Elias said with exasperation, "I go to the CS classes and I realized it was all math. Either math or logical equations, so word math." Shawn commented, "It was a lot more math than programming, is what I found."

Charlie said he made it to trigonometry in high school and did not realize that he would need advanced math for computer science.

I didn't make it up to calculus in high school because I didn't know that that was important. At the time, I thought that [trig] was pretty high. My mom had only made it out to algebra and my dad dropped out of high school, so he probably just did basic math, so I thought trig was pretty good. But then when I decided to be a computer science major, I decided I better take pre-calculus and I remember going to the [college] counselor and setting up my academic plan and telling her that I wanted to take pre-calc, and her discouraging me from taking it. She said it was going to be really hard and was "I" sure I wanted to take it. And this was after that first computer science class that I thought was a cakewalk.

And I guess I just thought I could do anything, since I never struggled in high school with anything. But she said, "If it gets too hard, come to me and drop it." And I remember just thinking, "That's not going to be necessary." And then pre-calculus hit me like a ton of bricks. I remember actually getting really depressed and working really hard and not getting very far in pre-calculus, and I got a D in it. I never made it past pre-calc and it still bothers me. (Charlie)

Seven of the study participants credited mathematics tutoring support as necessary for passing college-level mathematics. Notably, five of the participants who found it necessary also finished 4-year degrees in engineering or science-based majors. Only one participant currently enrolled in a science-based major had not needed mathematics tutoring. Two participants received mathematics tutoring but continued to struggle until they changed into humanities or art-based majors with less stringent mathematics requirements (see Figure 5).

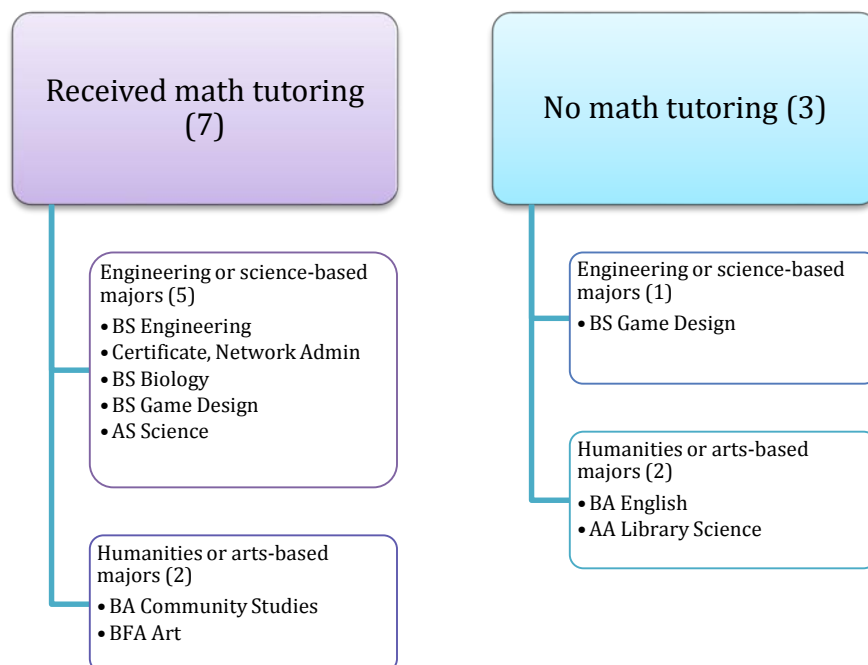


Figure 5. Participant mathematics tutoring and major selection.

Finally, two participants were not familiar with tutoring services offered at their campuses. George struggled in high school math and felt unprepared in college math, while Charlie did “ok” in high school math. Neither recalled seeking out or being aware of tutoring or peer study groups in high school. When they encountered trouble with mathematics in college, the idea of tutoring never occurred to either of them. Both admitted a belief that either you were intelligent enough to do well in something or you should not be doing it.

Charlie said, “I was just angry at myself. I was studying on my own every night while my friends hung out. I was sacrificing everything and it still wasn’t working.” Charlie found a green chalkboard and an unlocked classroom where he could work alone. He described being there late into the night to avoid distraction. When it came time to take his pre-calculus final, he said, “I froze. I just stared at the sheet of paper on the desk

for two hours and barely wrote my name. I've never felt so stupid." George admitted that he "should have studied more." When he spoke about math, his disposition changed from confident to unsure.

Mathematics tutoring was necessary for the majority of participants; nine needed tutoring and seven received tutoring. Nine mentioned struggles in mathematics as a contributing factor to leaving the CS major.

Little or no tutoring available for CS. While mathematics tutoring had a positive impact for half of the participants, all 10 participants identified a lack of CS tutoring at their colleges. One participant noted plenty of opportunities for mathematics tutoring and peer study groups, computer science had no options: "There was a sink-or-swim mentality" (Elias). Sandra did well in most of her college classes but struggled in programming courses.

Most of the college courses were no problem for me. . . . My other classes were okay. I got tutoring in math and did ok. But the computer science courses, there wasn't really any tutoring available . . . and I didn't get any help with the parts that I felt were extremely difficult.

Raul confirmed the need for tutoring.

I think I definitely would've needed a lot of one-on-one help to be in the computer science field. And I still haven't lost interest until this day. I want to learn coding because I want to create a couple apps, so right now I've been really doing my best to learn on my own.

Only two participants had any knowledge of available CS tutoring at their colleges, and those two students chose not to use it after initial bad experiences. Elias had a hard time communicating with the available tutors. He declared hotly,

I would've rather banged my head against rocks all day than try to decipher what the guy was trying to say. They were very socially awkward and just, like, they didn't answer direct questions. My tutor would do it [a code example] himself and then say, "Replicate." I don't learn that way.

Peter tried to work with the tutors but found only one he liked. He thought the others were not helpful. Peter concluded, “The problem was he [the good tutor] would [not be there] when I had class, so I had to skip a couple labs or classes just so I can go to him and do some programming outside of class.” Peter avoided the tutoring center at all other times after degrading experiences with other tutors. Peter’s body language when he spoke of these experiences demonstrated great frustration and resentment. Both Elias and Peter greatly stressed the need for approachable, helpful CS tutors.

Shame: Unprepared and outmatched. Shame was a recurrent topic among nine participants. As noted in previous findings, nine participants were high achievers in high school, accustomed to not only succeeding academically but also succeeding with ease. Participants conveyed the struggles to pass CS and math courses as a source of shame and conveyed this through words as well as body language. They indicated that interactions with classmates and professors deepened this shame. For example, Raul initially made friends with CS classmates but started to stay away from them. He explained, “I was becoming more of a hassle to them because I didn’t get it and they seemed to get it. So I ended up moving away from the group.”

Nine participants linked their struggles with CS classmates to their own high school CS exposure. These nine believed their classmates who successfully navigated CS were already familiar with programming languages. Elias explained professors assumed that those who were familiar with programming languages were innately suited for CS. Raul pointed out that he believed he would have made it in CS if he had high school preparation as he recounted his understanding of his CS peers.

[They] did take [CS] classes in high school. They took programming classes . . . that gave them an advantage. That little circle had programming classes in high school and they were even private programmers already; they had taken afterschool programs that taught them programming.

Lorenzo shared an expectation of what computer science would be and described encounters with peers.

I thought it would be fun being on the computer because I loved the computer anyways, but the lack of background did not help at all. And I think I really didn't know what was going on and my peers had an idea—the ones that had been exposed to programming before through their parents or their lifestyle—they crossed paths with programming before and that discouraged me. I was also worried, you know, I had to graduate.

Elias reported a similar experience

Most of the kids had prior coding experience with, like, C++. At least a lot of them had internships right out of high school, so they would go to software companies and code for free and learn it. Then they would come to class and the courses were extremely easy for them. But coming out of a “We play video games” background, we were no match.

Manuel also felt extremely unprepared and outmatched by his peers. His frustration is below.

I went to those [CS] classes and I was like, shit, these people already knew what Java programming was and like C++ and they worked with it. And I'm like, “Man, what the heck. I don't even know what half the stuff they are talking about is about.” And these people have this way upper hand, and I'm like, “Dang, you know. I don't even know if I'm going to be able to do this.” I started comparing myself to them and thought, “Am I going to be able to do good if these guys already got all this?” And this sucks for me because I haven't even experienced this and they already have that firsthand and, you know, so a lot of being unprepared kind of like pushed me away from it as well.

Though participants arrived at college with high computer self-efficacy, experiences in CS brought forth shame and degraded the self-efficacy of participants.

Mismatch between expectations of CS and reality. All participants reported a mismatch between what they imagined CS would be like and the reality of it. Raul reviewed this phenomenon:

I think jumping into it in college with really no understanding into what computer science really does and what it can do, I think that impeded me understanding it, and especially the options you can have from it. I remember in the classes, it was just, “This code makes this, this code and that stuff,” and so for me, that never made sense.

Charlie did not learn the difference between troubleshooting computer hardware and coding until he went to college as a CS major. Even though he left CS, Charlie continued to learn on his own.

I didn’t get away from computer science because I hated it. I think, at the time, it was just misinformation or a misunderstanding on my part. That first-generation college student knowledge had a lot to do with it. I didn’t have anyone in my family to say, “No, you should really stay in computer science.” I mean nobody really knew what the difference was between one college degree and the next, and most of my friends come from families without anyone who’d gone to college either, so I didn’t have any friends to tell me what the difference was either, or professors, for that matter. I mean, I didn’t really get close to any of my computer science professors and I didn’t really join any computer science organizations or anything in college. I’m sure that there was something there, but if there was, I didn’t know of it. If I can go back, I tell my younger self to stay with computer science. But you can’t go back, just go forward and do the best you can. And I’ve done that and I might still go back later, who knows?

George assumed he knew what to expect from the introductory CS course. He stated, “Because I had built computers, I had done tech support, I thought it would be more of that.” First, he was surprised to find that there were no computers in the classroom. “We actually didn’t have computers in the classroom,” he said with disbelief.

It was all lecture, basically. We worked out of a book. The professor did it on an overhead projector, or excuse me, the projector connected to his computer and then we would take a quiz after the class. We would have a whole huge homework assignment that we would go do at home. (George)

George commented that he would have greatly preferred short lectures paired with lengthy “hands-on time, and then you could [have a chance to] reflect back with the professor.” Sandra also determined she would have preferred a hands-on approach in the classroom. Sam learned how to create webpages in high school and enjoyed the largely project-based instruction. To his disappointment, his college CS classes “were broad and lecture-based.” All 10 participants mentioned a dislike for lecture-based CS curriculum.

Raul and Lorenzo emphasized a need for visual examples in the curriculum. They both realized that they had visual learning styles. In an introduction to programming course, Raul was given a coding project without visual examples of what outcome to expect.

I remember making a simple calculator. My expectation of programming was that I would actually see a physical calculator on my screen and it would do anything I wanted, and I think that’s what my struggle was with all along. I really wanted to visually see what I was doing and I couldn’t see that portion.

Raul continued, explaining that he approached his professor about this issue after discovering a tool that would allow him to visually see what his code was doing.

I realized it would help me learn in a visual manner so I had a conversation with him and he actually recommended that I choose another major. . . . So that just really disturbed me from actually wanting to take any more programming.

Both Raul and Lorenzo acknowledged the visual learning style was an advantage in their new majors. For example, Lorenzo majored in art for a while after struggling with programming and math. While an art major, he took a computer game design course that integrated some programming with art. He was excited to share about the labs and peer collaboration he found in game design. He reinforced that he was a visual learner in the art course and visual examples helped him understand programming theory.

Yeah, game design, they were more creative. For example, the professor gave an analogy that I'll never forget because it was a really cool analogy, that every time you create an instance of a class in computer science. He compared it to, like, a jellybean mold. So you have your mold and that's like a class, so every time you create an instance, it's like you created a jellybean with that mold. You can create a lot of jellybeans with that mold that you create. And it will be the same jellybean in all your instances in a program. It was a really artsy, cool analogy that I never will forget. It was really visual. It was on PowerPoint too. And that compared to a drawing of a triangle and a square connected with a line, these are cues, these are like data objects, they could definitely get more artsy with it, you know? That's me, and there are other people that passed the class, no problem. (Lorenzo)

Three participants mentioned pseudocode when they explained why they did not feel their CS courses were what they had imagined. Pseudocode is often used in CS textbooks instead of functional code in a programming language that can run an actual program. George just wanted to build something “real.” Elias agreed, pseudocode was not what he expected from his computer science curriculum and he had hoped “to be able to build applications that were actually useful right away.” Sam was also disgruntled by the “broad pseudocode” used in his classes and catalogs, which was one of the main reasons he left CS.

Participants were universally unaware of the contents of CS coursework upon entering college. They all lacked experience with programming and the lecture-based format did little to expand their knowledge of programming. Visual methods for exploring algorithms were desired by some participants, but these were not supported in coursework. Three participants had no understanding of the general purpose of pseudocode as a mapping and planning function for algorithm design. The participants who mentioned it did not recall receiving an explanation for how programmers used pseudocode and experienced it as an unnecessary hurdle in their education.

Theme summary. Though participants were eager to study CS at the onset of college, four essential challenges characterized their problematic experiences in the major. First, the need for mathematics tutoring eroded their academic confidence and placed an additional burden of time in their academic schedules. Second, the lack of CS tutoring reduced needed opportunities for social learning. The failure to thrive in CS degraded self-efficacy and brought forth shame formed the third challenge, and last, the mismatch between their expectations of CS and the reality of CS curriculum further led them to believe they had chosen CS mistakenly.

Reacting to the Work of Computer Science

Participants echoed reactions to the work of computer science. The coursework, course format, and the opportunities for peer and faculty interaction negatively impacted all 10 participants. Sub-themes included (a) longing for collaboration in CS, (b) longing for a connection to a multicultural community of support, (c) longing for personal relevance, (d) longing to help others, (e) less solo computer time, (f) experiences with faculty and students, and (g) a desire to finish and graduate on time due to financial pressures.

Longing for collaboration in CS. All 10 participants mentioned a desire to be more collaborative in their coursework within CS and noted the isolation within the CS program, an isolation fueled by numerous solo assignments and little group work. Sandra repeatedly highlighted the importance of collaboration and its definitive absence in CS. She declared with disappointment, “I was expecting more projects that were team-based.” Sandra explained why she left CS for engineering: “I enjoyed working with the other engineers and I made a lot of friends, where I didn’t make a lot of friends in computer

science.” Sandra longed for collaboration and she found that in her new major. She explained that in computer science, “We were expected to do a lot of the work on our own,” while in engineering, “There is no single contributor; there is more teamwork. I like that. I like working on something in a team.”

Raul wanted to replicate the collaboration experience he received in his high school MESA program. He expanded on that experience:

I came to the [high school] MESA program, where one of the competitions was making a website and working as a team. That was where I started sort of learning programming. That was with my friends, where we were all trying to, like, write different code pieces.

Raul and his friends learned from that collaboration experience and they each learned different coding skills and taught each other what they learned. He expected something similar in college but did not find it. Raul was unable to locate any CS clubs or organizations where he felt like he belonged. He explained, “The community didn’t feel very supportive at the time. I know there were a lot of clubs and organizations for it, but it didn’t call my attention.” He was able to find a strong connection through his work-study jobs. Those experiences as a tutor and peer advisor led him to a career in academic counseling. Similarly, Lorenzo did not feel welcomed by his CS peers.

I feel like there wasn’t a way that I could acquaint myself with [them]. My peers were very shy and the ones that were doing good, we didn’t have much of a similar interest, not enough for me to want to work with them. I wasn’t able to connect with them.

He did initially find a group of like-minded friends in CS. Lorenzo explained, “Yeah, man, but you know what? A lot of those friends switched majors as well.” Peter also longed for peer collaboration and support but found few opportunities in CS.

It’s frustrating because there’s no one to really sit there and say, “I know you’re in the same boat.” Especially as far as programmers and computer science majors

go, there is a sense of arrogance, a sense of ego for sure. These guys are thinking, “I’m smarter than you. I know more than you.” So if you show them any sign of weakness or any sign of needing help, that’s a sign of weakness to someone else. So, you know, I can’t ask for help. I can’t show them that I don’t know what I’m talking about.

Maybe I can get some help from someone else. It’s like, I don’t know. They just want to hoard all of that information for themselves. I never like that. It’s like, if I know something, and maybe you know something that you can help me with, let’s share that information. Let’s not share each other’s work, but let’s just give each other a little help here. Maybe we can finish our own work faster.

Manuel also had trouble relating to his peers in CS. He listened to conversations among his CS peers, and the conversations were enough to make him feel like he did not belong. He shared, “They were technical or logical, like, ‘There is this new game,’ where I’m more interested in things that are directly affecting myself or my community.”

Elias mentioned the reason he switched majors: “I was trying to find my niche and people that were willing to sit aside and teach me.” Fortunately, he was able to find this a few years after graduating from a 4-year university as a biology major. He now worked as a coder and technical writer. He laughed as he explained how he learned on the job. “The guy that taught me has his master’s degree [in CS] and he said that he could teach a monkey how to type, so it’s kind of messed up. I do a lot of XML coding.”

Sam was excited to share that right before he left CS, one of his professors had founded a programming club. He said, at each meeting,

We got together and put our heads together and decided what to build. We made a single Mario-type game in a programming language that some people knew and some people didn’t know. And it turned out that the people that didn’t know the programming language learned the programming language, so it helped them out and it was fun.

Sam’s involvement with the CS programming club project might have influenced his current trajectory. He had joined an organization that fostered entrepreneurs and community creativity from a co-working space. There, Sam began collaborating with

other learn-it-yourselfers to learn how to code without formal higher education. The membership fee for desk space was \$99 a month; however, Sam's fee was waived if he volunteered time towards running the co-working space. Sam was bubbling with excitement about this arrangement and he was more than thrilled to collaborate in this way.

Participants experienced a lack of collaboration opportunities driven by the absence of collaborative coursework and a pervasive feeling of "other" when interacting with classmates. These two things inhibited participants from development social learning mechanisms within the CS major. This led to a longing for collaborative spaces and some participants were able to find collaborative CS opportunities outside of their colleges.

Longing for a connection to a multicultural community of support. The inability to relate to peers and the lack of collaboration opportunities in CS led participants to seek out spaces where they felt accepted. Various multicultural student support organizations fulfilled this need.

Manuel was drawn into a community studies major after finding a place for himself working with MeCha (Movimiento Estudiantil Chican @ de Aztlán), a student group that promoted Chicano/a education and political consciousness. Sandra noted that the multicultural organizations supported her entry into and completion of her engineering degree. "I met a lot of people in engineering in MESA and SHPE [Society of Hispanic Professional Engineers] and SWE, which is the Society for Women Engineers. I got really involved in those organizations and I went to lots of conferences." Sandra lit up with excitement when she mentioned all of the opportunities for networking

in these groups. Although CS is a STEM major, it was not well represented in these organizations, according to Sandra.

Shawn sought out collaboration during his fourth year in college and, like Sandra, found it in a multicultural group. Shawn's 4-year college had a multicultural engineering support program and many of his peers frequented a study room.

You just go in there and study and there are always people in there, studying. People who have taken the class before you, and we will talk and they will say, "Here's an old quiz. This'll help you study," and so that was kind of the support I got at the end.

Lorenzo found support in MESA. He asserted, "I could definitely connect more with people in MESA because it was multicultural then, you know?" On the other hand, he mentioned, "The advanced computer science nerds, they were mostly White and very, um, their lifestyle was just very different." Like Sandra, Lorenzo did not know of any CS majors in MESA.

Raul participated in MESA as well. The researcher asked, "Did MESA expose you to any professionals in CS or CS faculty?" Raul answered, "Uh, not at all, no." Notably, MESA, a group that provides collaborative support for first-generation college students who pursue STEM degrees, did not influence participants to remain in CS. However, students who actively participated in MESA did remain in STEM majors.

Participants all experienced isolation as CS majors and longed for collaborative learning opportunities. Remarkably, seven participants found welcoming multicultural groups and peers who offered ample collaboration. Further, though participant experiences in MESA influenced STEM major selection, they did not provide opportunities for CS support.

Longing for personal relevance. All 10 of the participants mentioned a lack of relevance in the curriculum and most participants discussed the idea that project work added relevance. George began by identifying a longing for particular projects.

If projects could be a large part of computer science . . . you know, just saying that they could create things, maybe develop apps or focus on smart devices developing for android and iOS platform, I think that would be a huge point to capture the younger audience or my generation.

Charlie struggled with solo projects and found the most relevance for CS while participating in a mandatory class project. He found the class project highly beneficial, but it was the only group project he encountered. He explained,

We worked on a project for a non-profit child legal advocate group. We had to go out and meet with people from the non-profit, interview them, write up a proposal, and design them a complete website with, like, a backend database, you know, with the ability for their clients to log in. And the database had to store information in a way that the client could pull the information back out. But my group had to go out as a group and interview to put together a proposal for this client. And I really liked that project because I learned so much about how to handle the business end, I guess, of computer science. And the times that I got to get together with my group were really great. But I guess what was also super cool was one of my group members was really good at back-end programming, and I learned a lot from him because he had been coding for a while and the class was really easy for him. So I was helping him with the backend and he was able to point out any problems in my code and he was super helpful. It was so much better than sitting by myself and not knowing where I was going wrong.

Sandra and Elias also perceived relevance through collaborative project work.

Elias lamented, “I didn't learn relevant stuff until I got my job and I met a super-nerd who is willing to sit down and share.” Both Sandra and Elias found it easier to learn programming skills through project work with collaborative peers and colleagues. Raul gave a good example of how he encountered personal relevance in project-based work in an elective course:

So in this dance class, and I mean I have no background in dancing, and with being shy, I was kind of awkward, but there was one class session with a bunch of

graduate students, and they came with the computer and cameras and all sorts of equipment and they just set up. We were instructed to dance in front of the camera, and there was a backdrop, and what was pretty cool was, every time someone would perform different body movements, it would create different sounds, from drums to guitars to bells. You know, having a group of people creating music. And I found that really amazing, and so I kept my conversations going with that professor throughout that first year, and you know, he was the one that I would go talk to and visit every once in a while, and I would ask him about his projects, you know, tell him hi.

Because, I was still trying to be part of programming that first year, but I was trying to find other avenues because I didn't have a clue as to what I wanted to do with programming. So I really liked the fact that he had mentioned that a lot of the software was actually computer science but with an artistic point of view. So I really liked that idea, so that year I spent time going from the theater department to the art department, to the computer science department, to the film department, to the music department, trying to figure out which of those majors would allow me to do something like that; [figure out] which projects the teachers brought into the course. Um, so I spent that first year really going to different departments, looking at the class description, really trying to take classes that sounded interesting, especially with technology.

Each participant gave an example of a CS project they individually thought would enhance their attraction to the CS major. While most remained focused on project work that resulted in the creation of something personally usable by the participant, many also mentioned a desire to make something that would benefit their communities.

Longing to help others. Nine participants shared an attraction to a career that contributed to humanity. Sandra could not imagine a life of sitting in front of a computer, writing code all day long, because she “wanted to work with people” and “work on something that was relevant and helped others.” She said, “I just didn't see that in the computer science classrooms. There were a couple great instructors but I just couldn't tie my life to it at that time.” Peter mentioned a strong desire to help, noting that one of the reasons he left CS was because he missed human interactions and the opportunity to help people.

I just sat behind a computer screen all the time, writing code. My love and passion was more about helping people. Co-workers coming to me when they needed help: I would fix them up and they would be working again. With programming, I didn't feel like I was helping anybody. I honestly just felt like I was basically helping myself, you know? So that's why I kind of just switched too.

Peter explained this phenomenon further:

I just have more of a desire to help people now. I guess I still can help with programming, but it's a lot easier with computer technician or desktop support kind of stuff. It gives you more sense of purpose and more self-fulfillment. Because for myself, you know, I like to be more well-rounded. I like to know a little of everything about computers, you know, programming, software, hardware, you name it. But my emphasis, you know, was more of the one-on-one, the helping.

George was most attracted to his new major, library science, because he believed librarians “get to work with people in that regard and that they are helping and they help out a huge variety of people.” Manuel took a Latin American studies course that sparked his interest in helping his community. He was not interested in studying computer science after his focus shifted to youth empowerment.

I was, like, hardware and software is really cool, but you know, as I started getting involved, I felt like there was more of a need for myself to be implemented in that, versus in this other field that wasn't too kind to folks who wanted to create a change.

Helping others and the lack of the perceived benefit of CS in their communities emerged profoundly for participants. Remarkably, one participant went so far as to say the study of computer science felt “selfish” compared to the many other academic paths he could take.

Less solo computer time. Five participants mentioned a desire to spend less time in front of a computer than they imagined a programmer would need to do in a professional capacity. George clarified, “I didn't want to really be behind the computer

the whole time.” He believed this would be the case after conversations with friends who had graduated with CS degrees.

I trust their opinion. . . . They always say it’s 90% programming and then 10% any other activity, and I just don’t think that I want to be coding all the time. It’s very tedious work and it’s a very hard job too. It’s not an easy job, but the other factor is that I don’t always want to be in the cubicle or office space. I like to work and not be stuck behind a computer screen all the time.

Paul described his experience as a CS major, “I would lock myself in a tiny little room or my bedroom for hours upon hours, just by myself, coding and stuff, and that just kind of gets to a person after a while.” Elias spoke about his experience working on solo coding assignments.

To write straight elegant code, it takes a lot of patience and to figure out what’s wrong in the code, because products don’t work if there is a screw-up in the code. That takes a different type of brain, as far as I’m concerned. I don’t believe I have the concentration to write elegant intensive code. I tend to overlook stuff on my own, so I’m probably better off in my current occupation.

Sandra and Charlie also shared a disdain for working on their own, writing code.

While both were fine writing code with peers, yet they found the isolation of the CS coursework difficult. Charlie emphasized,

I’ve always loved computers so I really didn’t think that I would mind sitting in front of a computer all the time, but in the end, I really didn’t mind. I felt isolated. I mean, we did do one group project, but I didn’t really talk to my group members much, and it’s not like they were all nerds either. I had quite a few folks that were into graphic design, pretty cool people. But, it didn’t seem like the classes encouraged interaction. And I’m to blame for not seeking out people that were excited about computers, I guess, too. It’s just, I could do it when it was just HTML/CSS stuff because I could usually solve problems myself, but as I got into PERL/CGI, it was soul-crushing on my own. That’s not what I want to be doing day in, day out.

Five participants quoted in this sub-theme currently worked in a technology-based profession and did coding in their daily duties; however, coding was only one aspect of what they did. Elias explained why this worked for him: “I don’t like being in the office

24/7. I like traveling. I can be in front of the computer for a couple hours, but anything more than that, you know, you get eyestrain. You get bored.”

The reality of CS coursework and the desire for less solo computer time emerged in half of the participants. Though it potentially illustrated a poor match between the participant and the major, many participants currently worked in technology-based professions where they could explore their interests both in front of a computer and away from it.

Experiences with faculty and students. Only two study participants shared positive experiences with CS professors or TAs, and eight participants shared a recollection of dry, lecture-based classroom experiences and specifically mentioned a desire for approachable CS faculty. Notably, all eight participants found professors in other departments who were approachable and who utilized instructional methods that both engaged and motivated them.

Peter commented that he failed to receive the instructional support he needed in the introductory CS courses. He commented, “They didn’t really teach me as well as I thought. I was trying to play catch-up the whole time.” Peter approached his professors for help. When the researcher delved deeper into these interactions, Peter looked disgusted and said,

No compassion! I mean honestly, you would go to them, struggling. I mean, of course, you’re not going to go to them and ask them to write your code out. They are not going to do that. But let’s say you’re sitting there for 2 to 3 hours. You’re struggling on this one little part. And you’ve been busting your ass to figure it out. I’d go up to some of them and say, “Could you just give me like a little pointer, you know? Point me in the right direction.” And then they would look at me like I was either stupid, retarded, or I didn’t know what the hell I was talking about. They just had no compassion. They didn’t care. You know, why am I going to come back here and waste my time?

Lorenzo did not have as strong an emotional response to the question; however, he described a definite separation between himself and the professors.

You know what? I felt disconnected from them. I just felt like computer science, at least the professors that I was dealing with, they just felt like robots. I mean, they were not very social, their teaching skills weren't all that good, very monotone. They were just logical, you know. Their way about teaching the class was just very procedural. I mean, they are not going to crack a joke, not the ones I had. I mean, there might have been a couple younger ones, but the older professors were definitely very monotone and it was very hard to understand, and I felt disconnected from them.

Manuel also experienced a disconnection between himself and the CS professor.

There was one professor that didn't know how to teach our class at all, so that kind of kept me away from it. We would try to ask him questions and he wasn't really clear on it. It kind of turned me off trying to learn what he was teaching. It's just the support overall, like for myself, that I felt was not there.

Elias believed the CS professor he encountered after transferring to a 4-year college was more approachable than a CS professor had been at his 2-year community college.

The difference was the professor at the community college was super math-based, he had no personality. And at least inside of Cal State, they tried to make it fun. But the community college guy, it was just kind of, get them in, get them out. He didn't want to help.

Manuel described a lack of support based on racial differences. He reasoned,

A lot of it has to do with professors of color within the faculty and understanding the struggle. Overall, for myself as a student of color, the support was not there. And you know, going out alone, college was already a super culture shock for me. Coming from an all-Latino community, it was very different culturally.

Manuel believed he would have needed more internal support from the CS faculty to have remained in the major, but the faculty were unable to understand and meet his needs. Additionally, college exposed Manuel to needs within his community of which he

had previously been unaware. This knowledge led to a desire to participate in youth empowerment. He recalled,

We had a big conference and I wanted to go and help out because they're bringing kids from the Raza community. We did a program where we brought them up from a high school to get them interested in the college, and I asked her [a CS professor] if I could take just that day off and she totally said, "No, that's not possible." It wasn't going to be flexible at all, so that kind of scared me off [from CS as a major].

Not all experiences between CS professors and the participants were negative.

Sandra did not find her computer science professors less approachable than the professors in her chosen major, engineering. She remarked, "Computer science professors didn't influence me. They were boring, but so were my engineering professors." Additionally, the participants gave a few positive examples of faculty interaction. Charlie mentioned that he enjoyed his early CS courses and professors.

I liked the web design courses because they were mostly project-based. I really liked creating webpages and I liked being able to show my friends and family my homework, even though they didn't seem that interested in it. Later I started taking more programming—JavaScript first, which is really just a scripting language, and then Perl CGI. The projects stopped and we mainly listened to a lecture and then were sent home to do exercises out of the book. I really missed the team projects, though, and I think that is where I lost interest.

Sam had a mentor-type relationship with a CS instructor. He recalled the instructor encouraging him to continue in CS. Sam explained, "He said, and I'm exactly quoting him, 'I see what you can do and I think that you will have a bright future if you continue on this path.'" When discussing the faculty in his new major, George said,

[They] had always seemed to me to be easy, very easy people to go and ask for any sort of help. They'll help you with a research question, or if it's a personal question, they'll still help you out. They are very approachable people.

Before leaving CS, Manuel had a great early experience in a CS course that helped him define his learning needs. He believed one CS professor to be passionate about the subject matter and teaching.

[He] took his time to talk to the students about machine programming and C++. After I took that class, I thought, “Oh yeah, this doesn’t seem too bad.” But then we got into these other ones and I thought, “I don’t think this is going to work out after all.”

Manuel found many faculty in his new major who were similar to the first CS professor, which encouraged him and made him feel like he “was in the right place.” Significantly, he completed a mandatory 6-month field study where he met an education professor. That professor became an essential mentor that Manuel returned to over the years for advice and professional contacts.

Raul encountered positive experiences in other places; he did not know what to do professionally after graduation and had never been informed about the different graduate or professional degrees possible for college graduates. Fortunately, through work-study employment in college counseling, he found individuals willing to share their experiences as well as professional advice. He believed this mentorship was essential to his current professional trajectory because it led him to graduate school and a position as a college counselor.

Though positive experiences with CS faculty favorably influenced participants in their CS studies, the negative experiences outnumbered the positive ones. The negative experiences contributed to a sense of not belonging in the major and eroded self-efficacy.

Longing to finish and graduate on time due to financial pressures.

Participants were first-generation college students with limited financial means. College attendance has an opportunity cost and to lessen that cost, nine study participants worked

part-time, 15-20 hours a week, while attending classes. The financial burden of college attendance appeared to weigh heavily on participants. Raul did not feel able to take a risk on CS. Raul explained that failing CS

Was a little bit of a wake-up call, and I started to assess what was going on for myself. I was afraid of being on academic probation. I really didn't know anything about college, so I realized that maybe I just needed to take other classes that were easier at the time for me. I was disappointed because it was really something I have a strong interest in. It might've just been something I couldn't learn on the first try.

Peter found CS to be more time consuming than other majors were. Cutting back on work hours was a burden Peter did not feel able to maintain.

You really have to invest a lot of outside time, especially with the programs. Well, for the most part, I had the programs at home, but it's mostly just going there and being there at the same time as the professor. I noticed I had to cut back on work hours because I needed more time to go to the labs.

Manuel was worried about taking longer than 5 years to finish his bachelor's degree. He said, "I didn't want to have to stay in college for longer than 5 years because I was worried about paying the money back." Charlie and Lorenzo shared similar sentiments.

I was feeling that pressure to keep my grades up for financial aid and also just for pride, and struggling to get good grades scared me. And there wasn't any more financial aid for bad computer science majors than really good English majors. I mean, I probably could get better money if I was a good English major compared to a struggling CS major. (Charlie)

I was also worried; you know, I had to graduate. Everyone tells you, you need to figure out what you want to do first, but I was in college, so trying to figure out what I wanted to do until the end. I feel like I was lucky and graduated. (Lorenzo)

Sam believed the time and energy required to get to the relevant CS classes was too large a burden.

I thought it's just too much time and energy when I can just go home and learn on my own at no expense. There are so many resources that you can get to online. There are so many free courses the other universities are offering online, like at Stanford and MIT computer science, and I'm actually learning a lot on my own time. I come here [to the hacker lab] and I hang out with people. If I have any questions, people, I ask around, even if it's people here or online forums.

Timely college graduation driven by financial and internal pressures to achieve weighed heavily on study participants. For many, those concerns aided their decision to leave the CS major.

Theme summary. Though the work of computer science drove participants away, it also helped them define the items they personally found essential: peer collaboration, multicultural collaboration, relevance, helping others, getting away from the computer, experiences with faculty and peers, and graduation in 5 years or less. That knowledge resulted in well-defined educational and professional paths. Four participants found a welcome space in other baccalaureate STEM programs with robust multicultural student support groups. Three joined social science or liberal arts baccalaureate programs after taking electives in those subjects. One finished community college as a general science major but did not transfer to a 4-year institution because he chose to pursue computer science without higher education. One transferred to a 4-year university and returned to community college to complete a certificate in network administration. One was currently attending community college and intended to transfer to a liberal arts baccalaureate program.

Results and Interpretations

This section contains study results derived from the themes and paired with an interpretive discussion. The three themes described in Chapter 4 illustrated findings such as students shared pre-college characteristics, students faced similar challenges in college

CS courses, and students shared reactions to the work of computer science. Further analysis of the three themes compared to the relevant literature led to three findings: CS interest development hinged on computer ownership in the home, participants shared characteristics that were ideal for college success but not for CS success, and encounters in CS departments produced unique challenges for participants.

Finding 1: CS Interest Development Hinged on Home Computer Ownership

The literature review found a direct link between student success in computer science and early positive experiences with computers (Fisher et al., 1997; Taylor & Mounfield, 1994; Tillberg & Cohoon, 2005). Though California students have access to computing in public schools, computer availability had not translated to early positive experiences (Margolis et al., 2008). The present study found positive experiences with computers originated in the home. Having a computer in the home was essential for developing interest in computer science and increasing computing self-efficacy before college major selection.

Digital divide. Research addressing differences in computer ownership and Internet access across socioeconomic levels and ethnic classes has produced the term *digital divide*. This term is used to separate those who have computer and Internet access from those who do not. Though this divide has narrowed, demographic differences persist (Baldassare et al., 2013). Latino home computer and Internet adoption rates continue to trail all major ethnic groups (Zickuhr & Smith, 2012). Based on the participant age range of 20-28, interviewees were in a category where computer ownership was atypical (Baldassare, Bonner, Paluch, & Petek, 2008).

Remarkably, all participants had their own computer in the home during adolescence and gave numerous examples to illustrate high computer self-efficacy. Participants shared intensified memories of confident computer use and demonstrated powerful, joy-filled emotional responses when discussing their first computers. These early experiences motivated them to seek out computer courses in high school and CS study in higher education. In contrast, Goode (2010) found limited exposure to in-home computing resulted in a weak technological identity and limited access to computing-related education and career options. To increase computer science graduates, with its large and growing Latino and low-income population, California faces significant challenges in bridging the digital divide and the basic digital skills gap.

Mobile rift. While findings of this study showed an association between having a home computer and CS interest development, no similar association emerged between smartphone ownership and CS interest development. Latinos in California adopt smartphones over home computers at greater rates than all other groups and are more likely to access the Internet through a mobile device (Baldassare et al., 2013). Baldassare et al. (2013) also found lower income Californians were more likely to access the Internet through a smartphone than through a laptop or desktop computer.

The increase in mobile use among Latinos and low-income Californians is an emerging phenomenon without ample research. However, while most computing hardware had decreased in price over the last decade, smartphones and tablet sales might have been supported by telecom contracts, which reduced the entry price for consumers. The reduced entry price might influence the device selection of low-income Californians.

Mobile devices offer simplified user interface, which may also be more attractive for new users.

Though tablets and smartphones continue to increase in capability each year, significant differences persist in comparison with laptop or desktop computers. The hardware and software necessary for computer programming primarily exists on desktop and laptop computers. If the trend of Latino and low-income smartphone and tablet adoption continues, fewer Latinos and low-income Californians will enjoy the experiences necessary for the development of CS interest and strong technological identities. The trend could result in the creation of a mobile rift based on socioeconomic class and could limit who goes on to study CS. The result could be an extension of the current CS workforce demographic composition of generally fixed ethnicities and socioeconomic backgrounds.

Finding 1 summary. This finding revealed families who were able to provide computers in the home gave their children the experiences required for initially selecting computer-intensive higher education majors. Conversely, the lack of a computer in the home might be enough to exclude CS as a career option. The growing adoption of smartphones over home computing options could create a mobile rift, and without intervention, may lead to decreased Latino and low-income participation in CS majors.

Finding 2: Participants Shared Characteristics Ideal for College Success but Not CS Success

The majority of participants shared characteristics that were ideal for college students to possess: internal motivation and high academic achievement. Though participants were first-generation college students, they were academically college-

prepared and internally motivated to complete a degree. While a large proportion of community college students in California are underprepared for college (CCCCO, 2012a), among the participants, all but one were immediately placed in college-level academics after excelling in high school and graduating in the top 20% of their classes. Most had taken at least one AP course, although nine of the 10 participants had attended low-SES public high schools. However, this level of academic preparation did not result in the participants feeling CS prepared.

College-prepared. Although atypical for students of this demographic to have such high readiness, the participants were not exceptional in any marked ways relevant to this study and did not constitute a sufficient challenge to previous research mentioned above. Many participants belonged to MESA in high school or college and some MESA programs exclusively recruit students with GPAs above 3.2, although participants did not participate in programs with known firm GPA cut-offs. Though a number of factors were identified that might have contributed to the participants' success, to explain why these students excelled remained outside the scope of this study. This research did not conclusively identify why these participants were different from the general population of community college students. The sole difference identified between participants and their demographic group was a computer in the home during high school.

Interestingly, computer ownership was correlated to high academic achievement in high school. Salinas (2008) linked owning one's own computer in the home to academic achievement in college. The present study also found a relationship between in-home computers and an assignment in Track A, an academic track that provides college preparatory curriculum in high school. Typically, underserved students are more

likely to be in Track C, a track of students who receive only general curriculum taught with lower-quality instruction, no CTE, and little if any guidance from school counselors (Deil-Amen & DeLuca, 2010). Remarkably, nine out of the 10 participants were placed in Track A and one in Track B, a vocational track that provides career and technical education (CTE) to ready participants for the workforce. As underrepresented, low-income, first-generation college students without familial guidance, such placement alone set the participants apart from the majority of underrepresented, low-income, first-generation college students.

STEM-prepared. The quality and availability of higher level mathematics courses at low-SES high schools differs in comparison to high-SES high schools (Deil-Amen & DeLuca, 2010). Chaney, Burgdorf, and Atash (1997) identified a connection between the difficulty of mathematics courses in high school and student achievement in STEM. Likewise, the present study found that although nine participants were college-prepared, only six were truly STEM-prepared or eligible to enroll in calculus. The participants identified a low level of difficulty in their high school mathematics courses, but five of the six STEM-prepared participants found their mathematics preparation at low-SES high schools had not truly prepared them for college-level calculus. For the remaining four participants, a lack of advisement in high school left them unaware of the mathematics required for a degree in CS.

Problematic mathematics preparations further increased the number of courses required for a CS or STEM degree and correspondingly increased the time necessary to earn a CS or STEM degree. Though Bettinger and Long (2009) found students who took developmental courses fared better than did students of similar capability who did not

take such courses, all four participants who took additional mathematics coursework switched into non-STEM majors.

Tutoring helped bridge the mathematics preparedness gap in the current study. McClenney and Waiwaiole (2005) likewise found tutoring and learning support improved retention in all student groups and additionally proposed a focus on effective advising to help students navigate the higher education environment. As noted previously, effective advising was missing for participants in high school; however, McClenney and Waiwaiole (2005) found it to be essential for underserved students in community college.

The next section highlights the importance of mathematics and advisement in high school as well as a difference in course or instructional quality at low-SES high schools. A high-performing student at a low-SES high school is likely to emerge without being fully STEM-prepared, producing an additional burden for the student. The result of the lack of preparation was a class divide that placed additional hurdles in front of the students with the fewest resources.

Lack of opportunities for exposure to programming. Most importantly, participants universally lacked exposure to programming experiences in high school. Their high school computer science courses were general computer literacy courses or hardware technician courses. While in-home experiences with computers increased computing self-efficacy and high computer self-efficacy played a role in college success (Salinas, 2008), findings from the present study clearly indicated that computer skills by themselves were not enough to overcome the other barriers of CS.

This finding aligned with the findings of Margolis et al. (2008) that low-SES schools offered CS courses focused on basic digital literacy, desktop publishing/typing,

and Internet publishing. Such alternative computer technology offerings did much to encourage interest in CS, but the lack of programming availability contributed to a general misunderstanding between “computer science” as it was known to the participants and the “computer science” curriculum they faced in college. The misunderstanding essentially set up the students for a challenging experience at best, and at worst, a devastating experience that left them questioning their intelligence when compared to peers with adequate CS preparation and exposure in high school.

Finding 2 summary. Whereas for the majority of participants, preparation for college was distinguished, regardless of the level of mathematics reached in high school mathematics, preparation was not adequate. A significant number of participants were not advised on the importance of mathematics and STEM. The absence of programming exposure further set them apart from their classmates, where differences were already great. The underexposure to CS in high school caused many of the unique challenges discussed in Finding 3.

Finding 3: Encounters in CS Departments Produced Unique Challenges for Participants

Participants noted numerous encounters with peers, faculty, and tutors that left them questioning their major choice. Encounters became increasingly negative for participants after the first CS course.

Course format in introductory courses and peers with experience. Wilson and Shrock (2001) determined comfort level in introductory courses to be the most important predictive factor of success for undergraduate students. Participants in this study all successfully completed the introduction to CS course, a lecture-based course

taught in a traditional format. In-class tests focused on memorization of facts were standard. Though students were not encouraged to collaborate in their introductory CS courses, the format was familiar to the participants.

Most students found the first introductory course to neither encourage nor discourage their desire to study CS. The introductory course was followed by a course focused on algorithms, and for nine participants, this is where their academic struggles began. As soon as the focus shifted to coding, participants felt like they were at a disadvantage. They struggled with and detested the solo coding assignments, and peers with previous coding experience were reported either to stick together or to work alone.

Though this potentially illustrated a poor match between the participant and the major, pair-programming, the collaboration of two or more students on a programming assignment, has emerged in the CS field as a best practice and is now being used in some CS classrooms. Werner, Hanks, and McDowell (2005) found pair-programming helped female students perform better on exams and increased persistence in the students. Interventions aimed at enhancing student collaboration increased persistence across all ethnicities and genders (Briggs, 2005; Kumar, 2003; Porter et al., 2013).

The unwelcome environment and lack of collaboration described by the participants mirrored the findings of Margolis et al. (2008) that courses were attended primarily by “techie” White and Asian males, and conversations between the instructor and students who fit the techie description dominated class time. The participants’ words echoed other studies, which found White and Asian male students in CS to be unwelcoming to other groups (Cheryan et al., 2009). They experienced Steele’s (1999) stereotype threat as minority members of CS classes.

Stereotype threat was very real for underrepresented students, and according to Margolis et al. (2008), was especially threatening among students who studied computer-related topics. Participants in the current study found the classroom intimidating and retreated. Similar to the participants of Margolis et al. (2008), they experienced isolation and worried about being judged as unintelligent by classmates. Participants received negative messages about their capabilities, and although they were aware of an experience mismatch, the negative messages led to fear, stress, and poor academic results. Margolis et al. concluded that in part, the negative experience for students was due to CS faculty behavior driven by a belief that CS interest and skill was inborn.

This belief in inborn qualities can have profound effects on the classroom environment. Here, it results in the propping up of students with preparatory privilege, often leaving other students riddled with insecurity and doubt, and limiting their ideas about what is possible for their own lives. (Margolis et al., 2008, p. 85)

While the emotional power of participant CS experiences was difficult to quantify, it was important to highlight. Though all participants expressed negative emotions in relation to CS classrooms and coursework in college, four participants exhibited extreme emotional responses when discussing their experiences and their feelings of “other” both in and outside of CS classrooms. Eyes watered and voice octaves rose as they discussed the humiliation they felt when they were deemed by classmates, instructors, or themselves to be less intelligent than CS required. They exhibited signs of stereotype threat and their self-efficacy crumbled. Though the study methods limited what could be known from the faculty and peer perspective, participants recognized their preparation was not the same as that of peers and participants experienced insecurity and self-doubt around their CS pathway.

Peer support weak or absent. The participants noted the extreme lack of peer support in CS classrooms. Barker and Garvin-Doxas (2004) identified the link between defensive communication and attrition in CS among underrepresented students. Defensive communication reduces opportunities for students to talk openly with classmates about classwork. Rosson et al. (2011) concluded that peer support by way of social learning networks strongly influenced self-efficacy, and Barker et al. (2009) identified self-efficacy as the single most important predictor of persistence in CS. None of the 10 participants established permanent peer support groups among CS peers, although one participant noted that while he had a group of friends during the first semester of college, the entire group of friends switched into other degree paths. Another participant was directly influenced away from CS by acquaintances farther along in their CS studies. Although outside the scope of this study, such examples may point to implications for the influence of peer networks on academic choices, in addition to their roles of supporting academic success.

Schunk and Mullen (2012) affirmed that human learning primarily occurred in social environments. Tillberg and Cahoon (2005) identified the great significance of peers in interest development and persistence. As mentioned previously, all 10 of the participants identified a lack of social learning environments both in and out of the classroom. Without social learning opportunities, they lacked the essential human connections needed to validate learning milestones and share information to speed up group learning. One participant noted the emergence of a club that promoted social learning, but he had already made the decision to leave CS and was unable to reap the possible benefits or to provide insights for this study.

Cultural mismatch. Numerous researchers identified the existence of a culture within CS classrooms that was unwelcoming to outsiders (Barker & Garvin-Doxas, 2004; Barker et al., 2009; Margolis & Fisher, 1997, 2001; Rosson et al., 2011). Participants likewise experienced an unwelcoming culture in CS and a mismatch between the “techie” White and Asian male cultures and their own cultures. Participants revealed a belief that they were culturally unlike their classmates and professors, which created constant dissonance in their CS courses.

The cultural value of community service repeatedly emerged as a key difference for participants. A desire to serve others set them apart from their classmates who remained focused on video games and movies. Numerous participants also mentioned the importance of family and community in their daily lives. They longed for the aspects of their neighborhood that stood in contrast to the culture of the classroom. Participants identified experiences missing from CS study in and out of the classroom including: socializing with groups, serving their communities, and relating to others over shared interests. The lack of these things discouraged their participation in CS courses and encouraged them to locate spaces where they felt welcomed and valued.

Notably, positive experiences with faculty and peers in other departments influenced their decisions to switch majors. Nine of the participants credited faculty and peers in their new major for a portion of their major change. The cultures in multicultural STEM support groups such as MESA, SHPE, and SWE were also found to be welcoming and supportive of the values participants missed in CS.

Finding 3 summary. As previously mentioned, all of the participants entered college with high computer self-efficacy or a confidence in their ability to use a computer

and learn new computing skills with ease. Yet, nine experienced significant self-doubt about their computer skills in relation to CS during their undergraduate years.

Opportunities for social learning did not emerge for participants. This aligned with the Margolis et al. (2008) conclusion that students who lacked peer learning opportunities related to programming were at a greater disadvantage when positioned in a White and Asian male dominated CS classroom, which often entailed a preformed peer-supported network, leaving others on the outside. Finally, the cultural mismatch experienced by participants caused a large amount of dissonance and propelled them to locate other fields of study with similarly socialized peers and faculty.

Summary of Findings, Results, and Interpretations

Three major themes emerged from the analysis of the triangulation of interviews, artifacts, and observations: participants shared pre-college characteristics, faced similar challenges in college CS courses, and echoed similar reactions to the work of computer science. The findings that emerged from the research suggested (a) CS interest development hinged on computer ownership in the home, (b) participants shared characteristics that were ideal for college success but not for CS success, and (c) encounters in CS departments produced unique challenges for participants. The interpretations from these findings and results formed the basis for the conclusions and recommendations in Chapter 5.

Chapter 5: Conclusions and Recommendations

Introduction

The purpose of this research was to study the reasons why so few students completed CS programs at community colleges and specifically to consider the experiences of the underrepresented population. A careful analysis was conducted utilizing interview transcripts, observation field notes, and artifact protocol forms to identify recurring themes. This study was guided by three research questions:

1. What are the experiences that lead underrepresented, low-income, first-generation community college students to choose a CS major?
2. What are the experiences that lead these students to transfer out of community college CS programs?
3. What are the experiences that influence these students' new choice of major?

Through the interwoven voices of the participants, field notes, and artifacts, three major themes emerged and formed the findings of the study: students shared pre-college characteristics, faced similar challenges in college CS courses, and echoed similar reactions to the work of computer science. The literature review provided a foundation for the research. With the addition of the findings, the following results emerged: CS interest development hinged on computer ownership in the home, participants shared characteristics that were ideal for college success but not CS success, and encounters in CS departments produced unique challenges for participants.

Chapter 5 contains an exploration of the conclusions of the research formed jointly with the research questions and the findings. The presentation of the conclusions

is made alongside the researcher's interpretations through a constructionist lens. Conclusions are followed by a discussion to answer the three research questions. Recommendations for professional practice and future research are next, and the chapter ends with the researcher's final reflections.

Conclusions

This study focused on experiences and perceptions of underserved students concerning community college computer science. Research Question 1 embodied an important but ancillary focus of the study: What are the experiences that lead underrepresented, low-income, first-generation community college students to choose a CS major? Understanding the experiences of why these students first chose CS helped to frame a particular phenomenon. The participants came from an underserved group and they made an improbable choice to study CS. Understanding what set these participants apart from the greater underserved group was important because the same factors may affect their experiences in CS and in other majors as well. Research Questions 2 and 3 represented the core of the study. They sought to ascertain the intrinsic motivations for leaving CS and to explore the experiences that attracted the participants to their new fields of study.

Research Question 1

Research Question 1 was, "What are the experiences that lead underrepresented, low-income, first-generation community college students to choose a CS major?" The singular act of computer ownership began to set the participants apart from their peers. During the period when participants attended K-12, the majority of low-SES households functioned without a home computer. The experience of having a computer in the home

coupled with an innate curiosity about using and fixing the computer led the participants to seek other opportunities to interact with and learn about computers, expanding the resources they had available.

Interest and a lack of funds to maintain the initial computing resources led to increased involvement in learning computer maintenance, experiences that helped build high computer self-efficacy among the students. The high self-efficacy was further amplified as family, extended family, and friends routinely approached with technology issues. The participants all cherished the feelings they derived from helping others and the recognition they received for their knowledge and skill. This drove them to seek additional computing experiences in their junior highs and high schools and through employment.

Though CS interest was abundant, opportunities for learning programming skills were non-existent in their available academic and professional spaces, yet they had opportunities to take alternative technology courses such as audiovisual, graphic design, and computer/electronics repair courses. Such opportunities facilitated their continued interest in computers. This interest was paired with high academic performance and a strong desire to finish a 4-year degree, with the result as matriculation as CS majors.

The experiences with computers and computer technology courses that resulted in high computer self-efficacy and a corresponding position of authority and expertise among peers, family, and community networks was in direct contrast to the impotence and alienation participants experienced when they entered CS as a field of study. The world in which the students developed interest and expertise through practical interaction with their home computers and the courses available to them in the secondary school

system stood in stark contrast to the academic pursuit of CS, at least at the community college transfer level. Given this error in understanding, CS as a field of study must be better defined to facilitate the inclusion of people who come to the field with a passion and practical pursuit of computing from their everyday lives.

Research Question 2

Research Question 2 was, “What are the experiences that led these students to transfer out of community college CS programs?” The first CS course was in a format the participants had no trouble navigating. It consisted of attending lectures, reading textbooks, and taking multiple-choice tests. The course did little to discourage the participants from pursuing CS, though many reported struggling in the first-year mathematics courses required for the CS major and credited math tutoring for their success in math. Participants who struggled in math and did not utilize math tutoring did not complete the mathematics requirements for the CS degree.

Subsequent CS courses delved into programming languages and pseudocode. The alternative technology offerings to which participants had been exposed in high school did much to encourage continued interest in CS. However, the lack of programming experience and exposure to professionals in a CS field provided the participants with a false sense of the reality of the computer science field, a point noted in greater detail above. Though subsequent CS courses were introductory by design, participants were surprised by the math involved, were put off by interactions with peers and professors, and ultimately disheartened by a distinct absence of opportunity to shore up slim programming skills. Participants viewed their peers with prior programming experience as largely unhelpful, unapproachable, and distinctly culturally different from themselves.

Many participants agreed the CS faculty shared the attitudes of their more experienced peers, alienating them further from the field and limiting their resources to bridge the knowledge gap.

Participants wished for approachable CS tutors and professors who could help them not just understand how to code, but provide an environment in which learning could happen to meet the needs and interest of unprepared students as well as those of the more experienced members of the class. For participants, this environment included visual learning, project-based assignments, increased collaboration, and connections between CS projects and community service. Participants were clear in their desires and most developed their goals through direct experiences in other departments. This shift indicated best practices for engaging underserved students existed and could be duplicated in CS.

Research Question 3

Research Question 3 was, “What are the experiences that influence these students’ new choice of major?” Participants were drawn to their new majors while looking for elements they found missing in CS. For all participants, collaboration was a central requirement. None desired the solo coding and computer time required by their CS courses. They worried that if they completed a CS degree, they would relegate themselves to a professional existence of isolation. They went in search of people and communities with whom they could share, learn, and build.

Students sought alternatives to their CS studies and searched for areas with more communal activities. The alienating experiences of their time in CS and their inability to navigate the work led participants to conclude they needed a more collaborative work

environment. Some found collaboration in STEM support organizations. Others found it in project work in their new departments. Many stayed within STEM majors and still found more opportunities for collaboration. Collaboration was closely related to peer support, and many found thriving opportunities for engagement with peers in other departments.

The other fundamentals participants looked for included personal relevance and a drive to assist in the communities they came from. Many of the participants provided computer support to friends and family prior to attending college and they likely closely associated community service with a feeling of competence and self-efficacy. Though CS has been responsible for the creation of ample things that benefit most communities, introductory CS courses did little to convince the participants of this. Their experiences of low competence in CS courses supported two arguments: that CS courses did not provide adequate connection between an altruistic need to help communities, and that students missed the feeling of competence and self-efficacy they gained from being able to relay their knowledge to others. They were attracted to other fields by their perceptions of opportunities to work on things that mattered to themselves and their communities, but they also felt competent in their abilities to succeed in such fields.

Last, CS may be one of the more time-intensive majors. The participants found other majors to be less time-intensive. One participant opted for an engineering major, and that major, perhaps due to opportunities to collaborate with peers on projects, was less time-intensive for the participant. The introductory level coding assignments proved more time-intensive than were the introductory assignments in the eventual majors of the participants. Nine of 10 participants worked to pay for tuition and living costs, and

among those, four reported that they cut back work hours to complete CS homework, an additional burden that contributed to their decision to switch majors. The need to work to provide financial stability competed with the time necessary to complete course work successfully. This circumstance may mark CS as a field that favors socioeconomic advantage.

Participants were enticed by majors that let them finish their bachelor's degrees within 5 years. Many did not feel they could do so with a CS major after struggling and retaking mathematics and introductory programming courses. Some also felt pressure due to financial aid limits, as well as personal pride. As first-generation college students and academic achievers, they sought majors that allowed them to finish in what they felt was a respectable amount of time.

Recommendations

The recommendations of the research were based on the findings, results, and conclusions of this study and appear with the researcher's interpretations. They are designed to improve institutional practices at community colleges in an effort to increase persistence in CS courses. Recommendations for further research are included.

Recommendations for Institutional Leaders

California community college administrators should consider the following:

1. Ensure CS tutoring is available and staffed with approachable, capable tutors. Integrate the findings of this study into tutor training and assessment.
2. Offering CS specific scholarships and grants. Subject-specific scholarships and grants are not likely to originate from community

colleges. However, opportunities may exist for administrators to develop partnerships with industry, government, and philanthropy.

3. Create partnerships with K-12 districts to facilitate exposure to programming courses when no such courses are offered at local high schools. This requires additional investment in promotion to ensure students are aware of the available opportunities.
4. Review existing STEM multicultural support organizations and check for participant exposure to CS opportunities within those organizations. Enable student exposure to CS professionals, conferences, and community coding events.

Recommendations for Faculty

The research has shown CS to be unfriendly to underrepresented students; therefore, the need is essential for faculty to review curriculum and course formats for ways to improve retention. One example to examine is the CSIT-In-3 program at Hartnell College in Salinas, California. This grant-funded program provides underrepresented students enrolled in CS with a summer bridge program, performance progress tracking, priority registration, tutoring, funded research projects, summer internships, field trips, professional development workshops, weekly meetings, scholarships, and the opportunity to complete a CS bachelor's degree in 3 years through a partnership with CSU Monterey Bay (CSIT-In-3, 2014). California community college faculty should also review CS0 and CS1 courses offered at 4-year colleges for curricula and instructional changes that promote a supportive climate. Best practices from 4-year colleges include,

1. Software for visual learning,
2. Peer instruction (PI),
3. Breadth-first approach,
4. Flipped classroom approach,
5. Exposure to major CS intellectual and societal contributions with a focus on relevance and community service,
6. Cohort-based sections for underserved students,
7. Sections separated by prior programming experience, and
8. Student-selected group projects instead of one-size-fits-all solo assignments

Recommendations for Further Research

This research represented an attempt to begin the conversation about this specific group of students, in the hope that larger studies would come about to ultimately improve the experiences of not only this participant group, but all groups who seek to study CS and who share commonalities with the group. Specifically, female students from all ethnic backgrounds in the United States or any student who finds himself or herself as a distinct minority in a CS classroom may encounter experiences in computer science similar to those of the participant group. To accomplish the goal of an improved student experience in CS, academicians must know more about a number of topics. The following further research should be considered:

1. Research to replicate this study with more participants and locations.
2. Research to replicate parts of this study with underserved students who did not leave CS to identify any points of difference.
3. Research on STEM development activities, or the lack thereof, within MESA.

4. Participants in this study all had parents with low computer self-efficacy. A study exploring the levels of computer self-efficacy among parents or guardians and any connections between child self-efficacy levels could further explore this phenomenon.
5. A case study presenting the community college CS programs with the most CS transfer students. Such research may provide insights into what already works at community colleges.
6. A study of exit surveys of community college students leaving CS would allow for a larger participant sample.
7. A study of pilot programs that survey and place incoming CS students into cohorts based on computer self-efficacy and experience with programming.
8. A study comparing the CS curriculum of California community colleges, public 4-year colleges, and private 4-year colleges may provide insight into any practices that can be implemented or changed to increase persistence.
9. Participants universally identified an absence of CS tutoring at their community colleges. A larger study analyzing CS tutoring support at community colleges statewide may expand on this finding.
10. This study found high computer literacy among participants. Five stated they were the eldest child in the family and the other five made no mention of birth order. A future study on birth order and computer ownership in low-SES families could further explore this finding.

Summary

Locating interviewees is never a simple task. The study was expanded to include the entire state in an effort to help overcome the difficulty. At the onset of this study, the researcher had over 60 colleagues who regularly worked with underrepresented STEM students in colleges throughout California; however, this source did not initially produce the numerous interviewees projected. Participants were ultimately located through targeted phone calls to the researcher's closest STEM colleagues. This effort resulted in conversations that further highlighted one additional barrier to underrepresented CS matriculation and persistence at community colleges: exposure to CS professionals.

One illustration of this barrier arose during a phone call with a colleague from a Southern California community college. The researcher has the utmost respect for this colleague and has long admired her energy and caring for the underrepresented STEM students whom she guides like a determined and proud parent. The researcher contacted her to ask for advice while considering adjustment of the topic, yet she discouraged a change. During the conversation, she realized that she often had colleagues from all other STEM fields visit with her students and provide tutoring, insight into the career fields, and information about internships. Yet, in her long tenure as a MESA director, she never had a CS professor or colleague visit. Though she could not locate a participant for this study, the conversation encouraged her to reach out and find contacts who could open up CS possibilities to her students. She highlighted that she had never thought about it before and her eyes were now open to the possibilities. This moment of understanding infused the researcher with the determination to continue this study as originally envisioned.

After many more phone calls to colleagues across the state, enough participants were located. Through those conversations, many participants emerged who did not quite meet the study requirements: talented but underrepresented students who had attended 4-year colleges immediately after high school as well as students who had stopped attending community college altogether. Though their stories were not told here, the researcher looks forward to telling their stories in the future.

The researcher set out to explore the phenomenon of community college CS student dropout through the eyes of a particular segment of the community college population. The conclusions of the research showed definitive patterns of computing self-efficacy and academic achievement among the participants, yet a lack of relationship between the identified factors and success in CS majors. Conclusions revealed a lack of exposure to experiences that could result in better preparation for CS as currently taught at community colleges.

Recommendations for community college leadership and future research were an important part of the study. The study findings revealed barriers as well as life-changing moments that ultimately placed participants on alternative educational paths. Though no singular condition can bear full responsibility for the participants' decisions to leave CS, an environment that facilitated growth and learning was universally absent.

The need for environments to contain support for CS students is paramount. Remarkably, seven participants found environments centered on STEM and/or underserved students on their campus. However, according to participants, none of the environments contained elements to specifically support them in CS. Paradoxically, though this study found CS education in community colleges to be less collaborative, the

industry has moved towards more collaborative environments. Encouraging collaboration in CS education would therefore increase persistence and better prepare graduates for industry.

In closing, an important note is that over half the study participants longed to continue their CS studies at a future date, and all participants had utilized their knowledge of technology as a key toolset in their current majors or careers. This signified that the door was not closed. Although the students did not continue in the field, their interest and passion for the subject as they understood it had not disappeared. An opportunity exists to create environments that nurture the initial passion of students entering CS and to facilitate the development of skills needed to succeed. Change may better serve those who will come, as well as those who may yet return.

References

- Ahmad, K. N. (2012). *Measuring the impact of app inventor for android and studio-based learning in an introductory computer science course for non-majors*. (Doctoral dissertation). Retrieved from <http://cardinalscholar.bsu.edu/>
- Akbulut, A. Y., & Looney, C. A. (2007). Inspiring students to pursue computing degrees. *Communications of the ACM*, 50(10), 67-71. doi:10.1145/1290958.1290964
- Akbulut, A. Y., & Looney, C. A. (2009). Improving IS student enrollments: Understanding the effects of IT sophistication in introductory IS courses. *Journal of Information Technology Education*, 8, 87-100. Retrieved from <http://www.jite.org/documents/Vol8/JITEv8p087-100Akbulut297.pdf>
- Akingbade, A., Finley, T., Jackson, D., Patel, P., & Rodger, S. H. (2003). JAWAA: Easy web-based animation from CS 0 to advanced CS courses. *ACM SIGCSE Bulletin*, 35(1), 162-166. doi:10.1145/792548.611959
- Aronson, J., Lustina, M. J., Good, C., Keough, K., Steele, C. M., & Brown, J. (1999). When White men can't do math: Necessary and sufficient factors in stereotype threat. *Journal of Experimental and Social Psychology*, 35, 29-46. doi:10.1006/jesp.1998.1371
- Astin, A. W., King, M. R., & Richardson, G. T. (1981). *The American freshman: National norms for Fall 1980*. Los Angeles: Higher Education Research Institute, UCLA.
- Astin, A. W., Korn, W. S., & Berz, E. R. (1991). *The American freshman: National norms for Fall 1990*. Los Angeles, CA: Higher Education Research Institute, University of California—Los Angeles.
- Bailey, T., Jeong, D. W., & Cho, S. W. (2010). Referral, enrollment, and completion in developmental education sequences in community colleges. *Economics of Education Review*, 29, 255-270. doi:10.1016/j.econedurev.2009.09.002
- Baldassare, M., Bonner, D., Petek, S., & Shrestha, J. (2013, June). *PPIC statewide survey: Californians & information technology*. San Francisco, CA: Public Policy Institute of California. Retrieved from http://www.ppic.org/content/pubs/survey/S_613MBS.pdf
- Baldassare, M., Bonner, D. Paluch, J., and Petek, S. (2008, June). *PPIC statewide survey: Californians & information technology*. San Francisco, CA: Public Policy Institute of California. Retrieved from <http://www.ppic.org/main/publication.asp?i=831>
- Bandura, A. (1986). *Social foundations of thought and action*. Englewood Cliffs, NJ: Prentice-Hall.

- Bandura, A. (1995). *Self-efficacy in changing societies*. New York, NY: Cambridge University Press.
- Barbatis, P. (2010). Underprepared, ethnically diverse community college students: Factors contributing to persistence. *Journal of Developmental Education*, 33(3), 14-24. Retrieved from <http://ncde.appstate.edu/sites/ncde.appstate.edu/>
- Barker, L. J., & Garvin-Doxas, K. (2004). Making visible the behaviors that influence learning environment: A qualitative exploration of computer science classrooms. *Computer Science Education*, 14(2), 119-145. doi:10.1080/08993400412331363853
- Barker, L. J., McDowell, C., & Kalahar, K. (2009). Exploring factors that influence computer science introductory course students to persist in the major. *ACM SIGCSE Bulletin*, 41(1), 153-157. doi:10.1145/1539024.1508923
- Beach, J. M. (2011). *Gateway to opportunity? A history of the community college in the United States*. Sterling, VA: Stylus.
- Berger, P. L., & Luckman, T. (1966). *The social construction of reality*. New York, NY: First Anchor Books.
- Bettinger, E. P., & Long, B. T. (2009). Addressing the needs of underprepared students in higher education: Does college remediation work? *Journal of Human Resources*, 44, 736-771. doi:10.1353/jhr.2009.0033
- Beyer, S., Rynes, K., Perrault, J., Hay, K., & Haller, S. (2003). Gender differences in computer science students. *ACM SIGCSE Bulletin*, 35(1), 49-53. doi:10.1145/792548.611930
- Blinder, A. S. (2007, March). *How many U.S. jobs might be offshorable?* (CEPS working paper No. 142). Princeton, NJ: Princeton University. Retrieved from <http://www.princeton.edu/ceps/workingpapers/142blinder.pdf>
- Bloomberg, L. D., & Volpe, M. (2008). *Completing your qualitative dissertation: A roadmap from beginning to end*. Thousand Oaks, CA: Sage.
- Bohn, S., Reyes, B., & Johnson, H. (2013). *The impact of budget cuts on California's community colleges*. San Francisco, CA: Public Policy Institute of California. Retrieved from http://www.ppic.org/content/pubs/report/R_313SBR.pdf
- Briggs, T. (2005). Techniques for active learning in CS courses. *Journal of Computing Sciences in Colleges*, 21, 156-165. Retrieved from <http://dl.acm.org/>
- Busch, T. (1995). Gender differences in self-efficacy and attitudes toward computers. *Journal of Educational Computing Research*, 12, 147-158. doi:10.2190/H7E1-XMM7-GU9B-3HWR

- California Community Colleges Chancellor's Office. (2011). *2010-11 MESA survey results* [Data file].
- California Community Colleges Chancellor's Office. (2012a). *Basic skills accountability*. Retrieved from http://www.californiacommunitycolleges.cccco.edu/Portals/0/reportsTB/REPORT_BASICSKILLS_FINAL_110112.pdf
- California Community Colleges Chancellor's Office. (2012b). *Datamart* [Data files]. Retrieved from <http://datamart.cccco.edu/>
- California Community Colleges Chancellor's Office. (2012c). *Fall reports on staffing* [Data file]. Retrieved from <https://misweb.cccco.edu/mis/onlinestat/staff.cfm>
- California Community Colleges Student Success Task Force. (2012). *Advancing student success in the California community colleges*. Retrieved from http://www.californiacommunitycolleges.cccco.edu/Portals/0/Executive/StudentSuccessTaskForce/SSTF_Final_Report_1-17-12_Print.pdf
- California Department of Education. (2013). *Facts about English learners in California*. Retrieved from <http://www.cde.ca.gov/ds/sd/cb/cefelfacts.asp>
- California State University. (2012a). *Undergraduate degrees granted by campus and discipline division, 2011-12*. Retrieved from http://www.calstate.edu/as/stat_reports/2011-2012/deg04.htm
- California State University. (2012b). *Budget facts Gov. Brown's 2011-12 funding proposal*. Retrieved from <http://www.calstate.edu/pa/BudgetCentral/BudgetFactSheet2011.pdf>
- Carnevale, A. P., Smith, N., & Melton, M. (2011). *STEM: Science, technology, engineering, mathematics*. Washington, DC: Center for Education and the Workforce, Georgetown University. Retrieved from <http://www9.georgetown.edu/grad/gppi/hpi/cew/pdfs/stem-complete.pdf>
- Carter, L. (2006). Why students with an apparent aptitude for computer science don't choose to major in computer science. *ACM SIGCSE Bulletin*, 38(1), 27-31. doi:10.1145/1121341.1121352
- Castagnaro, A. V. (2012). *Evaluation a sixth graders' self-efficacy in response to the use of educational technology*. (Doctoral dissertation). Retrieved from http://scholarship.claremont.edu/cgi/viewcontent.cgi?article=1070&context=cgu_etd
- Chaney, B., Burgdorf, K., & Atash, N. (1997). Influencing achievement through high school graduation requirements. *Educational Evaluation and Policy Analysis*, 19, 229-244. doi:10.2307/1164464

- Cheryan, S., Plaut, V. C., Davies, P. G., & Steele, C. M. (2009). Ambient belonging: How stereotypical cues impact gender participation in computer science. *Journal of Personality and Social Psychology*, *97*, 1045. doi:10.1037/a0016239
- Ching, C. D. (2013). *Why race? Understanding the importance of foregrounding race and ethnicity in achieving equity on college campuses*. Los Angeles, CA: Center for Urban Education, Rossier School of Education, University of Southern California. Retrieved from http://cue.usc.edu/assests/CUE_WhyRace_2013.pdf
- Cphoon, J. P. (2007). An introductory course format for promoting diversity and retention. *ACM SIGCSE Bulletin*, *39*, 395-399. doi:10.1145/1227310.1227450
- Cphoon, J. P., & Tychonievich, L. A. (2011). Analysis of a CS1 approach for attracting diverse and inexperienced students to computing majors. Proceeding 11: *Proceedings of the 42nd ACM technical symposium on Computer Science Education*, 165-170. doi:10.1145/1953163.1953217
- Community College League of California. (2013). *Fast facts 2013*. Retrieved from <http://www.ccleague.org/files/public/FF2013.pdf>
- Cook, C. R. (1997). CS0: Computer science orientation course. *ACM SIGCSE Bulletin*, *29*(1), 87-91. doi:10.1145/268085.268119
- Creswell, J. W. (2007). *Qualitative inquiry and research design* (2nd ed.). Thousand Oaks, CA: Sage.
- Crisp, G., & Nora, A. (2010). Hispanic student success: Factors influencing the persistence and transfer decisions of Latino community college students enrolled in developmental education. *Research in Higher Education*, *51*, 175-194. doi:10.1007/s11162-009-9151-x
- CSIT-In-3. (2014). *Program information*. Retrieved from <https://sites.google.com/site/csitin3/program-information>
- Deil-Amen, R., & DeLuca, S. (2010). The underserved third: How our educational structures populate an educational underclass. *Journal of Education for Students Placed at Risk*, *15*(1), 27-50. doi:10.1080/10824661003634948
- Ed-Data. (2011). *State of California education profile: Fiscal year 2010-2011* [Data file]. Retrieved from <http://www.ed-data.k12.ca.us/>
- Executive Office of the President. (2012a, January). *Remarks by the President in the state of the union address*. Retrieved from <http://www.whitehouse.gov/the-press-office/2012/01/24/remarks-president-state-union-address>
- Executive Office of the President. (2012b, February). *Engage to excel: Producing one million additional college graduates with degrees in science, technology, engineering, and mathematics*. Retrieved from http://www.whitehouse.gov/sites/default/files/microsites/ostp/pcast-engage-to-excel-final_2-25-12.pdf

- Fisher, A., Margolis, J., & Miller, F. (1997). Undergraduate women in computer science: Experience, motivation and culture. *ACM SIGCSE Bulletin*, 29(1), 106-110. doi:10.1145/268084.268127
- Frantz, C. M., Cuddy, A. J., Burnett, M., Ray, H., & Hart, A. (2004). A threat in the computer: The race implicit association test as a stereotype threat experience. *Personality and Social Psychology Bulletin*, 30, 1611-1624. doi:10.1177/0146167204266650
- Gergen, K. J. (1985). The social constructionist movement in modern psychology. *American Psychologist*, 40, 266-275. doi:10.1037//0003-066X.40.3.266
- Gonzalez, V., & Soltero, S. W. (2011). Alternative multidimensional models explaining and improving academic achievement in Latino students. *Bilingual Research Journal: The Journal of the National Association for Bilingual Education*, 34, 265-278. doi:10.1080/15235882.2011.625464
- Goode, J. (2010). The digital identity divide: How technology knowledge impacts college students. *New Media & Society*, 12, 497-513. doi:10.1177/1461444809343560
- Green, D. (2006). Historically underserved students: What we know, what we still need to know. *New Directions for Community Colleges*, 135, 21-28. doi:10.1002/cc.244
- Gwinner, K. P., Prince, J. B., & Andrus, D. M. (2006). Attracting students into careers in food supply veterinary medicine. *Journal of the American Veterinary Medical Association*, 228, 1693-1704. doi:10.2460/javma.228.11.1693
- Hakimzadeh, H., Adaikkalavan, R., & Wolfer, J. (2011). CS0: A project based, active learning course. *International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies*, 2, 493-506. Retrieved from <http://tuengr.com/>
- Horn, C., McCoy, Z., Campbell, L., & Brock, C. (2009). Remedial testing and placement in community colleges. *Community College Journal of Research and Practice*, 33, 510-526. doi:10.1080/10668920802662412
- Howles, T. (2007). Preliminary results of a longitudinal study of computer science student trends, behaviors, and preferences. *Journal of Computing Sciences in Colleges*, 22(6), 18-26. Retrieved from <http://dl.acm.org/>
- Haungs, M., Clark, C., Clements, J., & Janzen, D. (2012, February). Improving first-year success and retention through interest-based CS0 courses. In *Proceedings of the 43rd ACM technical symposium on computer science education* (pp. 589-594). doi:10.1145/2157136.2157307
- International ICT Literacy Panel. (2002). *A framework for ITC literacy*. Princeton, NJ: Educational Testing Services. Retrieved from http://www.ets.org/research/policy_research_reports/publications/report/2002/cjik

- Jones, E. G. (2010). *Factors influencing program progression and degree completion among information technology students in the community college*. (Doctoral dissertation). Available from ProQuest Dissertations and Theses database. (UMI No. 3446949).
- Karp, M. M. (2010). An exploration of Tinto's integration framework for community college students. *Journal of College Student Retention*, 12(1), 69-86.
- Katz, S., Allbritton, D., Aronis, J., Wilson, C., & Soffa, M. L. (2006). Gender, achievement, and persistence in an undergraduate computer science program. *ACM SIGMIS Database*, 37(4), 42-57. doi:10.1145/1185335.1185344
- Kumar, A. N. (2003). Learning programming by solving problems. In L. Cassel & R. A. Reis (Eds.), *Informatics, curricula, and teaching methods* (pp. 29-39). New York, NY: Springer.
- Locke, L. F., Spirduso, W. W., & Silverman, S. J. (2013). *Proposals that work: A guide for planning dissertations and grant proposals*. Thousand Oaks, CA: Sage.
- Lucile Packard Foundation for Children's Health. (2013). *Free/reduced price school meals*. Retrieved from <http://www.kidsdata.org/data/topic/dashboard.aspx?cat=39>
- Macaluso, R. (2010). *Savoring the first byte: Girls and boys introductory-level high school computer science classes*. (Doctoral dissertation). Available from ProQuest Dissertations and Theses database. (UMI No. 3465737)
- Marakas, G. M., Yi, M. Y., & Johnson, R. D. (1998). The multilevel and multifaceted character of computer self-efficacy: Toward clarification of the construct and an integrative framework for research. *Information Systems Research*, 9, 126-163. doi:10.1287/isre.9.2.126
- Margolis, J., Holme, J. J., Estrella, R., Goode, J., Nao, K., & Stumme, S. (2003). The computer science pipeline in urban high schools: Access to what? For whom? *IEEE Technology and Society*, 22(3), 12-19. doi:10.1109/MTAS.2003.1237467
- Margolis, J., Estrella, R., Goode, J., Holme, J. J., & Nao, K. (2008). *Stuck in the shallow end: Education, race, and computing*. Cambridge, MA: MIT Press.
- Margolis, J., & Fisher, A. (1997). Geek mythology and attracting undergraduate women to computer science. *Impacting change through collaboration: Proceedings of the Joint National Conference of the Women in Engineering Program Advocates Network and the National Association of Minority Engineering Program Administrators* (pp. 137-142). Retrieved from <http://journals.psu.edu/wepan/article/viewFile/57956/57644>
- Margolis, J., & Fisher, A. (2001). *Unlocking the clubhouse: Women in computing*. Cambridge, MA: MIT Press.

- Margolis, J., Fisher, A., & Miller, F. (2000). The anatomy of interest: Women in undergraduate computer science. *Women's Studies Quarterly*, 28(1/2), 104-127. Retrieved from <http://www.feministpress.org/wsq>
- Mathematics, Engineering, Science Achievement (MESA). (2014). *About MESA*. Retrieved from <http://mesa.ucop.edu/about/>
- Maxwell, J. A. (2005). *Qualitative research design: An interactive approach* (2nd ed.). Thousand Oaks, CA: Sage.
- McClenney, K. M., & Waiwaiole, E. N. (2005). Focus on student retention: Promising practices in community colleges. *Community College Journal*, 75(6), 36-41. Retrieved from <http://eric.ed.gov/> (No. EJ873952)
- McCullagh, D. (2013, March 6). Silicon Valley stymied on immigrant worker plan. *CNET*. Retrieved from http://news.cnet.com/8301-13578_3-57572744-38/silicon-valley-stymied-on-immigrant-worker-plan/
- Merriam, S. B. (2002). *Qualitative research in practice: Examples for discussion and analysis*. San Francisco, CA: Jossey-Bass.
- Merriam, S. B. (2009). *Qualitative research: A guide to design and implementation*. San Francisco, CA: Jossey-Bass.
- Mertens, D. M. (2009). *Research and evaluation in education and psychology: Integrating diversity with quantitative, qualitative, and mixed methods* (3rd ed.). Thousand Oaks, CA: Sage.
- Moustakas, C. (1994). *Phenomenological research methods*. Thousand Oaks, CA: Sage.
- Murphy, L., Richards, B., McCauley, R., Morrison, B. B., Westbrook, S., & Fossum, T. (2006). Women catch up: Gender differences in learning programming concepts. *ACM SIGCSE Bulletin*, 38(1), 17-21. doi:10.1145/1121341.1121350
- National Center for Education Statistics. (2013). *National assessment of educational progress: The NAEP glossary of terms*. Retrieved from <http://nces.ed.gov/nationsreportcard/glossary.asp#s>
- National Science Foundation. (2012). *Science and engineering indicators 2012: Demographics of the S&E workforce*. Retrieved from <http://www.nsf.gov/statistics/seind12/c3/c3s4.htm>
- Ng, E. S., & Sears, G. J. (2010). What women and ethnic minorities want. Work values and labor market confidence: A self-determination perspective. *International Journal of Human Resource Management*, 21, 676-698. doi:10.1080/09585191003658847
- Nora, A. (1987). Determinants of retention among Chicano college students: A structural model. *Journal of Higher Education*, 67, 120-148. doi:10.1007/BF00991932

- Nora, A., & Cabrera, A. F. (1996). The role of perceptions of prejudice and discrimination on the adjustment of minority students to college. *Journal of Higher Education*, 67, 120–148. doi:10.2307/2943977
- Ortiz, L. (2009). *The persistence to graduation of Hispanic community college students*. (Doctoral dissertation). Available at PQDT Open: <http://pqdtopen.proquest.com/pqdtopen/doc/250064948.html?FMT=ABS>
- Palmer, R. T., Maramba, D. C., & Dancy, T. E. (2011). A qualitative investigation of factors promoting the retention and persistence of students of color in STEM. *The Journal of Negro Education*, 80, 491-504. Retrieved from <http://www.journalnegroed.org/>
- Patton, M. Q. (2002). *Qualitative research and evaluation methods*. Thousand Oaks, CA: Sage.
- Paulsen, H., & Griswold, J. (2010). Understanding the impacts of socioeconomic status on first-generation students: A case study. In T. H. Housel & V. L. Harvey (Eds.), *The invisibility factor: Administrators and faculty reach out to first-generation college students* (pp. 75-90). Boca Raton, FL: Brown-Walker Press.
- Porter, L., Garcia, S., Glick, J., Matusiewicz, A., & Taylor, C. (2013). Peer instruction in computer science at small liberal arts colleges. *Proceedings of the 18th ACM conference on Innovation and technology in computer science education—ITiCSE '13*. doi:10.1145/2462476.2465587
- Price, K. W. (2013). *Using visual technologies in the introductory programming courses for computer science majors*. (Doctoral dissertation). Available from ProQuest Dissertations and Theses database. (UMI No. 3558099)
- Pryor, J. H., Hurtado, S., DeAngelo, L., Palucki Blake, L., & Tran, S. (2011). *The American freshman: National norms for fall 2010*. Los Angeles, CA: Higher Education Research Institute, University of California—Los Angeles.
- Radovic, A. (2010). *Factors impacting persistence for African-American and Latino community college students*. (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses database. (UMI No. 3432891)
- Reed, D. (2001). Rethinking CS0 with JavaScript. *ACM SIGCSE Bulletin*, 33(1), 100-104. doi:10.1145/366413.364552
- Romeike, R. (2007, November). Applying creativity in CS high school education: Criteria, teaching example, and evaluation. In *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research*, 88, (pp. 87-96). Sydney, NSW, Australia: Australian Computer Society.
- Rosson, M. B., Carroll, J. M., & Sinha, H. (2011). Orientation of undergraduates toward careers in the computer and information sciences: Gender, self-efficacy and social

- support. *ACM Transactions on Computing Education (TOCE)*, 11(3), 14.
doi:10.1145/2037276.2037278
- Rozell, E. J., & Gardner, W. L., III. (2000). Cognitive, motivation, and affective processes associated with computer-related performance: A path analysis. *Computers in Human Behavior*, 16, 199-222. doi:10.1016/S0747-5632(99)00054-0
- Rursch, J. A., Luse, A., & Jacobson, D. (2010). IT adventures: A program to spark IT interest in high school students using inquiry-based learning with cyber defense, game design, and robotics. *IEEE Transactions on Education*, 53(1), 71-79. doi:10.1109/TE.2009.2024080
- Salinas, R. (2008). *The digital divide goes to college: Latino undergraduates and barriers to digital information*. (Doctoral dissertation). Available from ProQuest Dissertations and Theses database. (UMI No. 3322105)
- Sax, L. J., Astin, A. W., Korn, W. S., & Mahoney, K. M. (2001). *The American freshman: National norms for fall 2000*. Los Angeles, CA: Higher Education Research Institute, University of California—Los Angeles. Retrieved from <http://www.heri.ucla.edu/PDFs/pubs/TFS/Norms/Monographs/TheAmericanFreshman2000.pdf>
- Schneider, M., & Yin, L. M. (2012, April). *Completion matters: The high cost of low community college graduation rates*. Washington, DC: American Enterprise Institute for Public Policy Research. Retrieved from http://www.aei.org/files/2012/04/02/-completion-matters-the-high-cost-of-low-community-college-graduation-rates_173407573640.pdf
- Schunk, D. H., & Mullen, C. A. (2012). Self-efficacy as an engaged learner. In S. L. Christenson & A. L. Reschly (Eds.), *Handbook of research on student engagement* (pp. 219-235). New York, NY: Springer.
- Schunk, D. H., & Pajares, F. (1997). The development of academic self-efficacy. In A. Wigfield & J. Eccles (Eds.), *Development of achievement motivation* (pp. 16-29). San Diego, CA: Academic Press.
- Shashaani, L. (1997). Gender differences in computer attitudes and use among college students. *Journal of Educational Computing Research*, 16(1), 37-52. doi:10.2190/Y8U7-AMMA-WQUT-R512
- Smaill, C. R. (2010). The implementation and evaluation of a university-based outreach laboratory program in electrical engineering. *IEEE Transactions on Education*, 53(1), 12-17. doi:10.1109/TE.2009.2022323
- Steele, C. (1999, October 4). Interview by M. Chandler. [Transcript of video recording]. *Frontline* (No. 1802). Washington, DC: PBS. Retrieved from <http://www.pbs.org/wgbh/pages/frontline/shows/sats/interviews/.html>

- Steele, C. (2003). Stereotype threat and African-American student achievement. In T. Perry, C. Steele, & A. Hilliard, III (Eds.), *Young, gifted, and Black: Promoting high achievement among African-American students* (pp. 109-130). Boston, MA: Beacon Press.
- Taylor, H. G., & Mounfield, L. C. (1994). Exploration of the relationship between prior computing experience and gender on success in college computer science. *Journal of Educational Computing Research*, 11, 291-306. doi:10.2190/4U0A-36XP-EU5K-H4KV
- Tillberg, H. K., & Cohoon, J. M. (2005). Attaching women to the CS major. *Frontiers: A Journal of Women Studies*, 26(1), 126-140. doi:10.1353/fro.2005.0016
- Tinto, V. (1993). *Leaving college: Rethinking the causes and cures of student attrition* (2nd ed.). Chicago, IL: University of Chicago Press.
- University of California, Office of the President. (2011). *The facts: UC budget basics*. Retrieved from http://budget.universityofcalifornia.edu/files/2011/12/Budget_fact_11.29.11.pdf
- U.S. Census Bureau. (2011). *Census and population*. Washington, DC: US Government Printing Office.
- University of California. (2012). *Statistical summary of students and staff*. Retrieved from <http://legacy-its.ucop.edu/uwnews/stat/statsum/fall2012/statsumm2012.pdf>
- Valadez, J. R., & Duran, R. (2007) Redefining the digital divide: Beyond access to computers and the Internet. *High School Journal*, 90(3), 31–44. doi:10.1353/hsj.2007.0013
- Vekiri, I. (2010). Socioeconomic differences in elementary students' ICT beliefs and out-of-school experiences. *Computers & Education*, 54, 941-950. Retrieved from <http://www.journals.elsevier.com/computers-and-education/>
- Wells, R. (2008). The effects of social and cultural capital on student persistence: Are community colleges more meritocratic? *Community College Review*, 36(1), 25+. doi:10.1177/0091552108319604
- Werner, L.L., Hanks, B., & McDowell, C. (2004). Pair-programming helps female computer science students. *ACM Journal of Educational Resources in Computing*, 4(1). doi:10.1145/1060071.1060075
- West, M., & Ross, S. (2002). Retaining females in computer science: A new look at a persistent problem. *Journal of Computing Sciences in Colleges*, 17(5), 1-7. Retrieved from <http://dl.acm.org/>
- Wheeler, A., & Harris, P. (2008). Creativity & personalization: Freshman orientation for the millennial generation. *LOEX*, 137-140. Retrieved from <http://commons.emich.edu/cgi/viewcontent.cgi?article=1028&context=loexconf2006>

Wilson, B. C., & Shrock, S. (2001, February). Contributing to success in an introductory computer science course: A study of twelve factors. *ACM SIGCSE Bulletin*, 33(1), 184-188. doi:10.1145/366413.364581

Zickuhr, K., & Smith, A. (2012, April). *Digital differences*. Retrieved from Pew Research Center website: http://www.pewinternet.org/~media/Files/Reports/2012/PIP_Digital_differences_041312.pdf

Appendix A: Interview Protocol

Interview Time: _____ Interview Date: _____
 Interview Location: _____ Interviewer: Daniel Gilbert-Valencia
 Interviewee: _____ Title: _____

This study seeks to explore the experiences that lead underserved computer science students at California community colleges to transfer out of the computer science major into other areas of study. The audio and video-recorded interview is anticipated to take up to one hour as you respond to 10 questions regarding your experiences and perceptions that led to your decision to transfer to another program. I will take notes throughout the interview to record pertinent observations to this study.

Confidentiality is important. Your name as an interviewee will be replaced with a fictitious name (pseudonym) to maintain confidentiality. All data collected will be maintained in a secure locked cabinet at Drexel University Sacramento.

As a requirement of this research project, I must have your stated consent to participate in this study. As a reminder, you can withdraw from the study at any time. At this time, I am inviting you to ask any unanswered questions. Do you agree to participate? (Turn on the video and audio recorder, read the formal consent statement and verbal consent). Thank you for your participation.

I will now turn on the recording devices and begin recording.

Interview Questions

1. At what point did you know that you wanted to study CS? How did that happen?
2. How did your experiences influence your desire to study CS? K-12? In-home/family? Peers?
3. How comfortable or confident do you feel about your academic preparation for college?
4. Describe your CS education before you changed majors. How many CS courses did you take, what content did they cover, and how was learning approached?
5. How did being a CS major compare to your expectations of what you thought it was going to be like?

6. What part or parts of your CS program were the most memorable? Why?
7. What would you say were the main reasons you chose not to continue studying computer science?
8. How did CS program instructors and colleagues influence your choice to change majors?
9. How did you select your new major?
10. What haven't I asked you about yet that would help to understand why you left CS for another major?

Closing

Thank you for your time and participation. After I've completed the interviews, I will write a summary of your interview. Would you like a copy of the interview we've conducted today? Again, thank you.

Appendix B: Observation Protocol

CONFIDENTIAL

Observation Protocol: Participants

Interview Participant: _____ Date: _____ Time: _____

Descriptive Notes	Reflective Notes

Appendix C: Invitation to Participate

<DATE>

Dear <NAME>,

I am contacting you today in my role as a doctoral student at Drexel University. In partial fulfillment of the requirements for the Doctor of Education degree, I am conducting a study focused on experiences in computer science at California community colleges as seen through the eyes of students who have transferred out of computer science. I am writing to request your participation in my study, titled “Dropping Out of Computer Science: A Phenomenological Study of Student Lived Experiences in Community College Computer Science.” Dr. Kathy Geller, my dissertation supervisor will be acting as the Principal Investigator for this study and can be reached at (xxx) xxx-xxxx with any questions.

Study Synopsis:

Interest in Computer Science (CS) has waned as demand for CS workers surges. This phenomenon is widely researched however the community college segment of the CS pipeline has been rarely addressed. This phenomenological study will examine the experiences of students with barriers, interest development, and persistence support systems, specifically looking at how they influence student academic choice to leave CS. Semi-structured interviews and observations with students will be conducted, transcribed and coded. Data will be analyzed through a social-constructivist lens to provide insight into the shared cultures and how they can be navigated to create actionable strategies that can be applied to increase the number of overall computer science graduates at community colleges.

Considerations:

Your participation in this research study is strictly voluntary. Should you agree to participate, you will be asked to engage in a face-to-face, individual, semi-structured interview. The duration of the interview will last up to an hour and will take place at a mutually agreeable location. The open-ended questions that will be asked during the interview session are designed to provide insight into your experiences while studying computer science. You will also be asked to submit a resume to further convey your experiences. A resume template will be provided.

Confidentiality:

Should you agree to participate, all reasonable steps will be taken to maintain confidentiality and to safeguard your identity as a study participant. Information gleaned from the interviews will be maintained securely during the study period, and audio and video recordings of the interviews will be destroyed following the completion of the study. No personally identifiable information arising from your participation in the study will be shared with colleagues or administrators. Findings from the study will be reported in aggregate to protect the identity of all participants.

If you choose to participate in this interview you will be acknowledging your consent to participate in this study. You may opt out of the study at any time. Please feel free to present any questions or concerns at any point before, during, or after your participation.

Thank you for your consideration. If you are willing to participate in this research study, please contact me at your earliest convenience.

Sincerely,

Daniel Gilbert-Valencia
Doctoral Candidate
Drexel University
Center for Graduate Studies, Sacramento
xxx@drexel.edu
(xxx) xxx-xxxx

Appendix D: Resume Template

Sample Data

Name

Jill Perez

Education

Nevada College

B.A., Sociology

2013-2015 (expected)

California College

A.S., Biology

2011-2013

California College

Certificate, Networking

2009-2010

California High School

2005-2009

GPA: 2.5

Classes

CS 101 (Grade: B)

Biology 102 (Grade: A)

English 101 (Grade: C)

Sociology 103 (Grade: C)

Calculus 204 (Grade: D)

Clubs and Extracurricular Activities

MESA, Member (2011-2013)

Computer Club, Vice President (2009)

National Society of Black Engineers, Member (2009-Present)

SACNAS, Treasurer (2009-2010)

Professional Experience

Library Student Assistant

California College

April 2008- May 2009

Clerk

Target

May 2009-Present

Appendix E: Letter of Consent

Thank you for your willingness to participate in the research study, *Dropping Out of Computer Science: A Phenomenological Study of Student Lived Experiences in Community College Computer Science*, being conducted by Daniel Gilbert-Valencia, a doctoral candidate at Drexel University. This study is being conducted in partial fulfillment of the requirements for the degree of Doctor of Education in the Educational Leadership and Management program under the supervision of Dr. Kathy Geller, Principal Investigator and dissertation Committee Chair.

This study seeks to explore the experiences that lead underserved computer science students at California community colleges to transfer out of the computer science major into other areas of study. The purpose of this research is to study the reasons why so few students complete CS programs at community colleges. You were selected for this study because though you are an underserved student that studied CS and intended to major in CS but instead selected a different major. If you decide to participate in the study, you will engage in an audio and video-recorded interview that is expected to last up to one hour. You will respond to 10 questions regarding your academic experiences. I will also take notes throughout the interview to record pertinent observations to this study.

Confidentiality and privacy are critical and will be maintained throughout the study. Your name or any other identifying information will be omitted. You will be identified with a pseudonym only in reference to the interviews. All of the transcripts and notes pertaining to the interview will be synthesized and coded for purposes of this study. They will be maintained in a locked cabinet at Drexel University Sacramento and only available to Dr. Kathy Geller, Principal Investigator and myself.

Please understand that this study is strictly voluntary and at any given time you have the right to refuse or discontinue participation. Should you choose to end the conversation early, your data will not be included in the study's findings and conclusions. For your information, there are no known risks or discomforts associated with this study.

If you have any questions, please contact me at xxx@drexel.edu / (xxx) xxx-xxxx, if you have any questions regarding the interview. You may also contact the Principal Investigator Kathy Geller, Ph.D., Drexel University, School of Education in Sacramento at xxx@drexel.edu / (xxx) xxx-xxxx.

Please sign this consent form acknowledging the nature and purpose of the procedures. A copy of this form will be given to you for your records

Name Printed	Name Signature
Study Title	
Date	