

**In Pursuit of an Uncommon Literacy:
African American Adults' Experiences in a Computer Code Bootcamp**

By
Antonio A. Byrd

A dissertation submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy
(English)

at the
UNIVERSITY OF WISCONSIN–MADISON
2019

Date of final oral examination: May 7, 2019

The dissertation is approved by the following members of the Final Oral Committee:
Kate Vieira, Associate Professor, Curriculum and Instruction
Christa Olson, Associate Professor, English
Morris Young, Professor, English
Matthew Berland, Associate Professor, Curriculum and Instruction
John Diamond, Kellner Family Distinguished Chair in Urban Education,
Education Leadership and Policy Analysis

Table of Contents

Abstract..... ii

Acknowledgments..... iv

Chapter 1:
Introduction: Computer Code Bootcamps, Racial Inequality, and Black
Liberation..... 1

Chapter 2:
“Like Coming Home”: African American Adults Playing and Tinkering Toward a Computer
Code Bootcamp..... 22

Chapter 3:
Between Learning and Opportunity: A Study of African American Coders’
Networks of Support 49

Chapter 4:
Making the Pivot with Coding Literacy..... 80

Chapter 5:
Conclusion: Critical Imaginings of an African American Coding Literacy Practice 110

Works Cited 124

Abstract

This study complicates claims that computer programming can facilitate the social mobility of underrepresented people. Such claims often reproduce what literacy scholars have called the “literacy myth”—the idea that prestigious literacies can help people overcome entrenched inequalities. To this end, my study addresses the following questions: In what ways do students’ experiences of code bootcamps help them overcome (or not) the effects of systemic racism on their literacy learning? And what do such experiences reveal about the relationship between race and writing in our current moment of technological change?

To answer these questions, the dissertation takes the experiences of African American adults attending a Midwestern computer code bootcamp called Clearwater Academy as its focus. The project uses interviews, participant observation, drawings, and fieldnotes to capture participants’ racially inflected personal histories with digital literacy, their learning practices in the present, the network of relationships needed to support their learning, and what accounts for their using coding literacy to receive reward and opportunity.

I find that despite the strategic ways participants accessed computer literacy, it appears that for many structural racism, from early schooling to workplace, may overwhelm the opportunities those emerging technologies may afford to racially marginalized people. I argue that efforts to democratize any emerging literacy for opportunity must promote an antiracist philosophy.

After reviewing scholarship on race, literacy, and digital technology, I present three findings. Chapter Two, “Like Coming Home”: African American Adults Playing and Tinkering Toward a Computer Code Bootcamp,” suggests students draw on personal legacies of informal “tinkering” (and not their experiences of racially biased schooling) to learn computer coding. Then, Chapter Three, “Between Learning and Opportunity: A Study of African American

Coders' Networks of Support," demonstrates that success in a computer code bootcamp in a context of white supremacy necessitates a complex and often ad hoc network of social support, while Chapter Four, "Making the Pivot with Coding Literacy," argues that even after successfully completing a computer code bootcamp, systemic racism may hinder African American adults' success in the tech workplace. However, outside of the tech workplace, participants can create new ways to use computer programming in their lives. I close by reviewing primary findings, describing limitations of the study, and explaining next steps for this project.

Acknowledgements

This project would not be possible without the voices of the participants described in this dissertation: Zeus, Rosie, Isaiah, Alex, Sufjan, Zelda, DeAndre, Kevin, Nadaline, Keisha, Princess, Alice, and Pierre. I thank you for sharing your knowledge with me, and I pray my presence in some way blessed your lives. I hope the readers and hearers of your words will help teach them that even in a post-racial and technologically advanced society, racially marginalized people still seek liberation from white supremacy. The words written in this document would not be possible without the kindness and generosity of Clearwater Academy's program director who permitted a graduate student and total stranger to step into a classroom of brilliant and determined women and people of color to learn from them and report on their exceptional knowledge. Finally, this project would not be possible without Clearwater Academy's instructors, Richard and Jane. Both often offered ideas on who I should interview and what events would be interesting to observe. Thank you for asking if I was okay and if I was getting the information I needed to complete this dissertation.

The project is the culmination of many encounters, relationships, and locations over the course of my time in Madison, Wisconsin. Beginning with the faculty from Composition and Rhetoric, Curriculum and Instruction, and Education Leadership and Public Analysis. My thanks to Professors Maisha Winn, Mary Louis-Gomez, Carl Grant, and Amy Johnson Lachuck from Curriculum and Instruction for showing me that I need to always pay attention to ethics when doing research for and with my own people and for showing me how to tell their stories in a way that honors them, respects them, and turns their mic up. Professor Matthew Berland took a strong interest in my research from before I asked him to be an outside reader. Our meetings throughout the summer about my first chapter helped ease me into the process of writing a dissertation; your

experience and knowledge of computer science also helped me understand how I should think about computer programming during my research. From Education Leadership and Public Analysis, thank you to Professor John Diamond for your quiet and calm teaching about the ridiculous ways our education system perpetuate oppression and for giving me language to describe racial inequality and literacy.

However, the foundation for my work is in learning from and thinking with faculty from my own program, Composition and Rhetoric. For the past five years, Professor Kate Vieira has served as wonderful model for mentorship insight on everything, from setting up a strong research project to negotiating the sometimes-fraught business of finding a job, has grounded me in the field and she has served as wonderful model for mentorship. I extend my thanks and gratitude to Professor Christa Olson for her careful critiques of my teaching and research and Professor Morris Young for his collected advice on navigating the PhD program and the next steps I may take in my academic career.

In the Writing Center, I encountered two wonderful mentors: Brad Hughes, thank you for your helping me consider how to have difficult conversations while maintaining my own professionalism the dignity and humanity of others. Nancy Linh-Karls, thank you for your smile, for accepting my random appearances in your office to help me discuss my time on the job market, for collaborating on a range of projects, and, most important, being excited for my ideas. Your enthusiasm has always strengthened me. I greatly appreciate Jen Fendel for our pleasant conversations over the last two years about winter, the Catholic church, and happenings in the Writing Center. Thank you for printing out my handouts in color for my many teaching demonstrations. Because of you, all of my interviewers had something to remember me by.

Jason Rocha and Virginia Schwarz, thank you for your love and support. Jason showed me what hospitality looks like: throw BBQ and invite everyone you know. Your practical

thinking, compassion, and loyalty were all key to my own growth as a person. Virginia, thank you for introducing me to the vegetarian and vegan life and what it means to be California girl living in the cold tundra of Wisconsin. Your thoughtfulness and display of affection has helped me re-see my own relationships and has left me to more easily step into the shoes of others. Jason and Virginia, some of the best conversations about Madison, academic life, UW-Madison, and the directions our quality of life should take have happened over cold beer. In some way, the beer culture in Wisconsin has its benefits to graduate study. James Ryan, thank you for welcoming us into your home where we could experience your impeccable knowledge of board games. the love of your children, Jonah and Iris, and your no-nonsense wife, Heidi. Our game nights together always helped me recuperate from a long week of writing, research, and teaching.

To Chris Castillo, you are my best friend and brother. I have enjoyed sharing our personal struggles, feelings, and triumphs, and learning from you about hip hop and Chicago no other person could offer. Thank you for being a constant guest in my small studio apartment; your presence made it feel more like home. And thank you for holding up a mirror to see myself more clearly. I will miss you most of all.

And Julia Garrett, I remember visiting UW-Madison for recruitment days. We sat across from each other at The Great Dane. You offered your wisdom about the strengths of the department and what professional directions a PhD could take me. That was the start of years of wonderful support, clear and timely advice, and many laughs. I'll miss our study sessions in the Wisconsin Historical Society and at various coffeeshops. There is, of course, a list of others to thank in my department and beyond: Neil Simpkins, Elisa Findlay, Matthew Fledderjohann, Brenna Swift, Brandee Easter, Mia Alafaireet, Tori Thompson-Peters, Calley Marotta, Kathleen Daly, Jeff Godsey, Mari Lewis, Zhe Yu Lee, Maddie Pape, Kevin Walters, Jon Tan, Emily and David Colo, and Amy Sue.

Thank you to the Graduate Christian Fellowship, Faith Community Bible Church, the Christ In Us Bible Study Group, and FCBC Sunday Friend group for making the years here in Madison fun, exciting, and grounded in Christ. Thank you to my family in the Humanities Working Group for being a source of comfort and prayer and informal mentorship on how to navigate the job market without offending or bringing dishonor to Christ. Thank you, Maggie Flamingo, for your awesome giving spirit and the many rides to and from the airport during my time on the job market. Joel Ballivian, for your engaging arguments on ethics and Christian theology; Giselle Martinez for excitement and passion for Christ and for my and others' success.

Thank you to all the coffee shops I ever visited to write my dissertation in Madison, especially Michelangelo's. I visited enough times starting in November 2017 that the baristas knew my order before I stepped into the line! Thank you for the idle chats about video games and board games; they were a welcome distraction (or a wonderful respite) from data analysis and writing. I will miss your hummus plate most of all.

Chapter 1

Introduction: Computer Code Bootcamps, Racial Inequality, and Black Liberation

Computer code has become the basis for our navigating everyday lives, and leisure activities; it “shape[s] people’s interactions and transactions, and mediate all manner of practices in entertainment, communication, and motilities” (Kitchin and Dodge 9). In recent years, educators, politicians, Silicon Valley entrepreneurs, non-profit organizations, among others, have claimed that because computer code powers and structures our everyday tasks everyone should have a deeper understanding of how our digital technologies work. Knowing computer code could accrue personal well-being and economic and political participation. A citizenry knowledgeable of programming will be eligible to fill job vacancies in an ever-growing tech sector, have the ability to make the tasks of non-tech professions more efficient or more productive, and learn to apply the problem-solving methods of computer science in a variety of situations. At the very least, exposure to learning computer programming may inform people on how the digital world works and help them see programming language as an option for creative expression (Guzdial).

Most important for this study is the expectations that computing education can help combat economic inequality and institutional racism in general as well as in the tech industry. Silicon Valley attracts and recruits mostly white men, creating a perceived “brogrammer culture” that is hostile to women and other underrepresented people. The presence of a majority white male tech force can also result in software that center the cultural experiences and knowledge of whiteness rather than address the lived experiences of marginalized communities. A diverse workforce will help challenge the unconscious biases of these homogeneous companies and the

software they create. Software development is a lucrative profession, offering low-income minoritized people an opportunity to obtain social upward mobility.

An industry of computer code bootcamps have come out of efforts to increase access to coding literacy education in public schools. With the financial and material assistance of sponsors like national and local tech companies, computer code bootcamps provide adults accelerated training in computer programming and job skills. The end goal is that graduates of these programs may become paid intern, fulltime employees of a tech company, or freelance web developers, among other high-paying careers. Some programs actively recruit people of color, such as Yes We Code, Black Girls Code, Code Start, Black Men Code, and Hack the Hood under a social justice framework. In addition to coding literacy training for financial well-being, then, they also seek to build the self-esteem and self-worth of people of color and help them envision themselves as key participants in creating and maintaining digital cultures.

But does this approach work to meet the twin goals of upward social mobility and community contribution? In my review of research on computer code bootcamps I have found few studies that answer this question. Kyle Thayer and Andrew Ko published a paper that examined the barriers twenty-six computer code bootcamp students encountered during and after their training; meanwhile, Course Report gives comprehensive reporting on the computer code bootcamp industry including tracking the graduate rates of students in North American. Thus, I find a need to further describe the socio-cultural, economic, and technical forces that shape the process of learning programming language over several weeks or months. Without a more detailed look into these forces, computing education appears to be about learning a contextless skills-based process that anyone can participate in without difficulty. However, naming these forces shows how social structures operate in the lives of marginalized people as they learn programming language.

This dissertation explores that process and its consequences through a qualitative study of African American adults' experiences learning coding literacy in a computer code bootcamp. In doing so, this study updates understandings of the connection between literacy and race. Reliance on digital divide rhetoric to claim that programming can facilitate the social mobility of underrepresented people, can often reproduce what literacy scholars have called the "literacy myth"—the idea that prestige literacies can help people overcome entrenched inequalities. My study complicates this claim. I ask: In what ways do students' experiences of code bootcamps help them overcome (or not) the effects of systemic racism on their literacy learning? And what do such experiences reveal about the relationship between race and writing in a moment of technological change?

To answer these questions, I developed a qualitative research protocol that captured twelve participants' racially inflected personal histories with computer literacy, their learning practices in the present, and the network of relationships needed to support their learning. I found that: a) students draw on personal legacies of informal "tinkering" (and not their experiences of racially biased schooling) to learn computer coding; b) to succeed in code bootcamp in a context of white supremacy necessitated a complex and often ad hoc network of social support; and c) even after successfully completing the bootcamp, systemic racism hindered their ability to succeed in the tech workplace. Despite the strategic ways participants accessed coding literacy, it appears that for many structural racism, from early schooling to workplace, may overwhelm the opportunities those emerging literacies may afford to racially marginalized people. I argue in this dissertation that efforts to democratize any emerging literacy for supposed opportunity must promote an antiracist philosophy.

By focusing on computer code bootcamps and racially marginalized adults, this study makes three contributions. First, it opens a new chapter in the ongoing historical narrative of

African American literacy; this study examines how African American communities keep up with emerging digital technologies and use them to transform their own literacies to improve their sociomaterial circumstances. This research provides literacy studies and computing science educators commentary on the relationship between learning programming language and racial identity. Second, the study gives attention to accelerated literacy learning programs as a crucial sponsor of literacy for marginalized people. These programs can be embedded in communities that are economically and socially stratified. We can see what happens when sponsors of literacy offer marginalized people access to the material, social, and cognitive affordances of programming language amidst these conditions. Third, my ethnographic study looks at the implications for classifying programming language as a tool for social justice. Here social justice is, as Leigh Patel explains, “the quest for the realignment and rectification of social inequalities” (90). When a workplace literacy program seeks to realign social inequalities using coding literacy, how does it negotiate the logics of the workplace and programming language with the mission to humanize African American adults so they can do self-humanizing work? While this dissertation does not give definite answers to these questions, it is a starting point for addressing these questions. In this introduction, I situate my study within relevant scholarship on literacy and racial justice. Then I describe my study before outlining the dissertation’s three data chapters.

Conflicts Among Literacy Education, Work, and Racial Stratification

In this section I review scholarship on the relationship among print literacy/digital literacy, employment, and social stratification and institutional racism. I first review research on both academic and public belief in literacy—from print to coding—as a cure-all for social ills, an assumption that New Literacy Studies has shown to be more complicated and even more dangerous to marginalized people. Then I narrow the focus on how these ideals manifest

themselves in sponsors of literacy—institutions that disseminate literacy as raw material for economic use. I consider the workplace as a sponsor of adult literacy education, a key source for financial stability to workers, profit for companies, and economic well-being for the nation. Describing this scholarship shows that despite the belief that literacy and digital literacy are beneficial, social institutions can still create a hierarchy of which literacies matter the most to maintain institutional power. Training marginalized people in coding literacy as a form of equity and equality conflicts directly with this logic of stratification.

From Words to Code: Tracing the “Benefits” of Literacy to Individuals

Belief in the social and economic advantages of literacy, what is called autonomous model of literacy (Street), have historically persisted even into our present moment. Over time, scholars and laypeople alike have suggested literacy a panacea for all social ills. The social value of literacy became prominent at the end of the nineteenth century in North America alongside visibility of marginalized people in mainstream society. Illiteracy became a way to measure society’s health and attach it to a series of societal problems perpetuated by such marginalized communities. Writers like George R. Stetson and James P. Monroe blamed the rise in crime and immorality on immigrants, African Americans, and the working class, and they prescribed literacy education and intellectual engagement as the only way to “keep the lower orders docile” (Ohmann 677). Literacy, in theory, helps create the necessary “conditions of civilization” in people: literacy makes you moral, an upright citizen, productive in the national and global economy, financially well-off, rational, empathetic, become a cultural tastemaker, become a homeowner, and less likely to commit crimes (Graff; UNESCO; Street). The very word literacy carries connotations of public good such that one can easily attach it to any other skill that the United States citizenry should have. Thus, when “literacy” attaches itself to other kinds of knowledge (i.e. financial literacy), we start paying attention (Wysocki and Johnson-Eilola).

Cultural anthropologists and psychologists in the mid-twentieth century sought evidence that the development of modern Western civilization hinged on literacy civilizing the mind and advancing human cognition. Jack Goody and Ian Watt and David Havelock, for example, used the ancient Greeks as recipients of literacy's power. Reading and writing, these scholars observed, helped create a logical mind that could judge the worth of long-held ideas and produce new ones. The ancient Greeks could then separate myth from history, unlike pre-literate oral societies which transmitted tradition and myth in face-to-face communication and relied on memory and mnemonic devices to carry that cultural knowledge from one generation to the next. For oral societies there was no true past; they lived in a constant "nowness" (Goody and Watt; Ong). Recording oral stories and cataloging genealogies and narratives, however, led Greek readers to question both the accuracy and distance between their time and the time in which these narratives took place. The ancient Greeks frequently put narratives, histories, epic poems, and traditions under scrutiny and found some inconsistent and unreliable; literate Greeks rewrote and made canon new versions of these traditions. This mental work through writing—"providing definitions, making all assumptions and premises explicit, and observing formal rules of logic"—is the foundation for Western civilization (Olson 278).

Autonomous models of literacy continued in the wake of the personal computer's widespread availability in the 1980s. Considering computers' relatively vast capabilities, resounding assumptions about the social consequences of accessing computer literacy abounded. Significant stakeholders such as software companies, educators, and federal government suggested that knowing how to use a computer was essential to keeping up with a changing economy. Thus, they followed a simple mantra: "get computers into schools, get kids involved in computing, and the earlier the better" (Sterne 197), as later in life individuals can acquire high-paying jobs and thus obtain social upward mobility (Selfe; Noble; Goodson and Morgan). In

addition to computers' economic contribution, American citizens' digital literacy seemed to determine the health of national security. The Reagan administration hoped that collaboration among military, industry, and private sectors on technology development and deregulating the tech industry under a neoliberal philosophy of individual freedom and competition would attract a workforce that could then strengthen the United States' position in the Cold War.

These underlying ideas continue into our present literacy era (Selfe and Hawisher) with the latest ideological iteration falling onto the very bones and muscles that hold our societal infrastructure together. Programming language is promoted as the fourth 'R' in the list of basic skills students should learn alongside writing, reading, and arithmetic. Thus, learning algorithms and algorithmic thinking is a "human right" and mass computing education creates the "economic, educational, and business products of the 21st Century" (Strauss). However, the benefits of coding were touted long before the learn to code movement in 2012. Since the 1960s, prominent computer scientists and educators such as Alan Perlis, Seymour Papert, Cynthia Solomon, and Adele Goldberg and Alan Kay have argued that coding could liberate people from corporation's control over humanity (Greenberger; Papert; Solomon; Kay and Goldberg). Papert himself developed the Logo programming language to help children extend their own cognitive learning to new heights through the computer. Today, the coding for all movement claim that programming language empowers individuals, expands the mind and human capability, fosters citizenship, collective progress, and employment (Vee).

Nevertheless, we must be wary of the resounding "good news" of coding. Just as the autonomous theory of literacy masks ideologies that contribute to oppressing the many for the benefits of a few, the computing for all movement can mask the United States' undemocratic educational system (Margolis) and the ways this education leaves some people more literate than others. To truly assess the claims of the learn to code movement and also parse through the ways

coding contributes to both empowerment and inequality, we must empirically investigate how computing science education follows students into other contexts. The next section of this literature review will discuss how stratification plays out in the context of literacy education for job marketability.

Workplace Literacy Sponsors, Policy-making, and Education Inequality

The learn to code movement advocates for equal access to programming language, but it can actually participate in doling out programming language as a limited resource. The grassroots efforts of Code.org and other non-profit organizations have galvanized federal and state action on democratizing computing science education (Lapowsky; White House; Connecticut State Board of Education; Wisconsin Department of Public Instruction). The public education system is perhaps the largest sponsor of literacy in the United States and has thus accumulated the responsibility to distribute literacy. However, implementation of computer science education in schools requires money, time, and resources (Guzdial). Computer code bootcamps give accelerated training in coding literacy, bypassing public school systems and years of college study to possibly link adolescents and adults directly to work faster (Lewin; Mims).

Despite these best intentions, different ideologies on literacy can form depending on what dominant group owns or possesses literacy. Whether that ownership is local, national, or global, the ideas that an individual, group, or institution has about literate practice shape the ways others use literacy. Thus, the aforementioned benefits of literacy, for example, are themselves ideological and learned in a social context (Street; Stuckey). Because computer code bootcamps are considered a viable and faster alternative to formal schooling in programming language, we can see how code camp ideologies are rooted in theories of sponsorship and stratification.

Toward the end of this section, I will show how the literature demonstrates that precisely because of systemic oppression, racially marginalized people may be drawn to learning coding literacy.

To produce these ideologies of literacy benefits, social institutions set the terms for what “literate” means. These sponsors of literacy— “any agents, local or distant, concrete or abstract, who enable, support, teach, or model, as well as recruit, regulate, suppress, or withhold literacy -- and gain advantage by it in some way” (Brandt 19)—regulate access to literacy and shape the ways their beliefs become policy and how those policies then shape the literate (and illiterate) lives of the sponsored. Analyzing sponsors of literacy can reveal social stratification in action. To highlight division of literacy, scholars in the field have broadened literacy’s definition as a resource (Brandt). Under this rubric, literacy’s abundance is a myth. Instead, it’s perceived as a precious limited resource that must be hoarded and carefully shared among the highest-ranking social categories (Weber). Those without that resource cannot fully participate in every day lived experiences. Across civic and economic life, for example they are left disadvantaged. Literacy is a marker of and bridge to the social and cultural capital that sponsors dole out to reproduce the status quo and maintain social categories (Marx).

As sponsors of computational literacy, computer code bootcamps belong to a larger history of workplace literacy education for the American worker. Work and literacy were mostly divorced for the better part of several centuries: literacy was taught for religious salvation while learning to work happened on the job (Bessen; Gowen; Graff). However, the industrial economy drew literacy into the work environment substantially. As part of President Lyndon Johnson’s Great Society program, Congress passed the Adult Education Act of 1966 to fund adult literacy education. If American citizens lacked basic literacy skills for the workforce the United States could not compete economically with other countries. Adult literacy education “centered more

on preparing adults for workforce training, reflecting and reinforcing a human capital perspective that shapes definitions of literacy and rationales for funding literacy education” (Bannon 318).

In subsequent years, and as recent as 2014 with the passage of Workforce Innovations and Opportunity Act Title II: Adult Education and Family Act, literacy has been treated as a skill serving not political or aesthetic needs (Brandt, “Losing Literacy”), but the needs of employment (Bannon). When the literacy crises of the 1980s erupted (Sheils), not only were schools implicated for failing to teach youth literacy well but the labor market was also blamed for failing to train highly skilled workers in the changing economy (Johnston). For this reason, public schools and college composition were not the only influential sponsors to follow the back-to-basics movement; employers created literacy programs to solidify worker productivity.

Our current knowledge economy has transformed literacy into a writing-centric practice as “raw material, as labor power” to ensure the creation and circulation of information in the global economy (Brandt, *Literacy in*, 171; Farrell). In her book *The Rise of Writing*, Deborah Brandt’s extensive life history interviews workaday writers show that as the knowledge economy and digital technology develops, more people spend less time reading and more time writing for production and profit. In thinking about workplace literacy programs in this new context, many research participants verified that school had not prepared them for the writing required from their jobs; instead, much of what they knew as learned on the job.

A hierarchy might be established within the workplace based on who has what skill. Knowledge workers, those with the ability to maintain and share valuable information across multiple platforms, are considered more literate and thus more valuable than “traditional workers”, those who “[operate] as a kind of robot remotely controlled by a manager” (Castleton 559). Traditional workers, then, may not keep up with the changing nature and definitions of literacy. Lacking the skills needed to contribute to a company, these workers are in danger of

losing their jobs as companies downsize and put more emphasis on project-based work (Castleton). A nine-month ethnography on a hospital's workplace literacy program for African American traditional workers—"housekeeping, food service, and laundry"—confirms this at the intersections of race and gender. Participants reported feeling underappreciated and experienced microaggressions at the hands of white employees who resided at the top brackets of the hospital's workplace hierarchy. The results of the program showed significant stratification: only one employee reported achieving some upward social mobility (Gowen).

What skills are and are not valuable does not just happen in official spaces for training workers. The literacy individuals learn depends in some ways on the literacies the economy needs. For example, a life history study revealed the significant outcomes of two different literate lives—Lopez, a Mexican American learning Spanish, and Branch, a European American learning programming language. Both grew up in the same Midwestern town, but Lopez and her family had more difficulty finding resources for Spanish learning than Branch, who had easy access to computers and communities of programming pirates for his coding education. As programmers developed their skills, the economy expanded computer software as a necessary tool for everyday living. Spanish-speaking and Spanish translation, for Lopez, was only useful in certain contexts. Meanwhile, Branch's programming language was useful in multiple contexts as he moved from one location to the next (Brandt, *Literacy in*). On one hand, literacy education categorizes according to need, but it also suggests that we may cease to think of literacy in terms of time, space, and skill, but rather networks of space and place. What you can do with literacy puts you in particular social position (Wysocki and Johnson). The call for programming language to help some students get the edge over others, as Guzdial explains, is one of the reasons all students at his university learn computer programming.

African Americans have relied heavily on sponsors of literacy for social advancement and upward mobility since first arriving in this country as slave labor in 1619. Scholarship in African American literacy history demonstrates a vast undertaking in directly and indirectly accessing or creating, sometimes with guile, sponsors for literacy education: Reconstruction makeshift schools, churches, hush harbors, Freedom Schools, Citizen Schools, historically black colleges and universities, and informal community literacy programs, to name a few examples (Anderson; Cornelius; Graff; Heath; Vogel; Henry; Fisher). The plethora of out-of-school literacy programs show that obtaining literacy has been a key part of African American philosophy of education for generations (Perry). Although there is an assumption that African Americans are inept with technology, a problem partly perpetuated by narrow definitions of the digital divide (Banks), scholarship shows that in fact African Americans have actively sought out Internet technology for civic participation (Everett) and upward social mobility (Nelson), continuing a tradition of using any new literacy that arises for empowerment and resistance to white supremacy.

Coding for All: A Continuation of an Old Story

The literature review above suggests that discourse on coding as a literacy today continues a genealogy of discourse on literacy's ability to reap positive consequences. But those material and social consequences remain uncertain and unevenly distributed across social and racial groups across multiple contexts. Nevertheless, there is something about education in general, and literacy more specifically, that keeps drawing people into its wings, leading to a vast network of sponsors of literacy. We might continue to follow digital literacy in the same way we have for traditional literacy, but with a closer eye on discourse about programming language as a universal skill that rewards material gains for all, the sponsors of the literacy who emerge, and the ways these sponsors shape the sponsored digital literacy.

Although initially confined to the tech industry, programming language can spread into other professions. Evolution of coding into a type of writing widely used would “disrupt accepted understandings of what counts as working knowledge, and the ways in which new knowledge might be made and used in globally distributed workspaces” (Farrell 193). There is an economic incentive for public schools to muster the resources needed for nation-wide computer science curricula to train a “college and career ready” citizenry. In the meantime, several computer code bootcamps sponsor students’ coding skills and directly link them to work. For some computer code bootcamps, this is a matter of social justice, an opportunity to quickly and immediately give access to low-income minorities to a highly specialized and lucrative skill.

We have an opportunity for continuing the work already set out to closely study African American experiences with emerging technologies. We also must treat the workplace as another point of access to the materials that lead to equity for African Americans, as they negotiate or take up the ideologies they encounter in training and reap (or don’t reap) the sociomaterial consequences of this education. In addition to school, home, and community literacies, we might complicate workplace literacies and workplace environments when they position themselves as sites of liberation in local communities, and then outline what happens at the nexus of African American rhetorical and literate tradition and workplace literacy. In the final section, I describe the context of the study, the theoretical frameworks that direct my analysis, and my approach to data collection. I end with a chapter overview of the dissertation.

An Opportunity for Learning More

This project seeks to understand the experiences of African American adult students at Clearwater Academy, a full-time, fourteen-week intensive coding boot camp offered to low-income women and people of color in a mid-sized Midwestern city. Clearwater Academy is an extension of a larger non-profit organization that provides community programming to address

poverty, racism, and sexism in the city. In addition to teaching web development, Clearwater teaches students basic job skills needed for the workplace, such as writing resumés and cover letters, presenting elevator pitches, and creating digital portfolios. After students complete this training, they may apply for three to six-month paid internships in a local company where they build on their training. The program hopes that students will have the tools needed that they can find jobs on their own.

This project explores the process of learning a highly specialized and uncommon literacy amid stratified material conditions, the consequences of this learning, and the perception of programming language as an object of and for social justice in the context of a workplace literacy training program. This ethnographic study explores the following theoretical question: What are the parameters for classifying programming language as a part of social justice literacy and what is the justification for this classification? To flesh out this theoretical question, I turn to answer the following primary research question: How do low-income African Americans experience the social, cultural, economic, and technical inequalities associated with literacy, as they bet their chances on coding literacy's promise for upward social mobility? To help answer this question, I investigate answers to two sub-questions. First, how do these African American students learn coding literacy as they manage other obligations that fall disproportionately on members of communities of color? Second, what are the social and material consequences of programming language for these African American students? In addition to these questions, I explore to what extent students see programming language as an act of social justice.

Theoretical Frameworks

My study draws from multiple perspectives on literacy and learning, many of which are used to develop the theoretical question of how programming language operates as a form of social justice in the context of labor and the knowledge economy. Wysocki and Johnson-Eilola's

re-definition of literacy for a digital era proves useful for this study: “a process of situating and resituating representations in social spaces.” In these spaces, the digitally literate not only access information and move with it but are also “active participants” in creating and arranging information. This perspective on literacy reflects how the knowledge economy as a concept works: knowledge workers manage, trade, and create information using digital technology (Brandt). In addition, the suggestion that the literate adopt a position in space relates to Rob Kitchin and Martin Dodge’s code/space concept, which I also use to theorize what coding literacy does: For Kitchin and Dodge, code/space is “any space that is dependent on software-driven technologies to function.” The authors use check-in areas and supermarket checkout lines as examples of spaces that rely on software: if the computers in an airport shut down, the check-in area becomes “a fairly chaotic waiting room”; if the electronic cash registers in the supermarket crashes “a supermarket instead [becomes] a temporary warehouse . . . The facilities to process payments manually have been discontinued, staff are not trained to process goods manually (they no longer rote-learn the price of goods), and prices are not usually printed on items” (Kitchin and Dodge 17).

In these examples, the work of code relies on the physical conditions that surrounds it just as much our physical conditions rely on code’s conditions. This research study takes the ways software can enter objects and regulations literally and metaphorically. We often think about the ways humans and machines are blending— learning how to code is an example of that blending. This poses a series of questions that run in the background of the study: What happens to marginalized people’s lives when the act of coding enters their material conditions? What happens when they are first coded and then sent out to code the spaces they live in? And what happens if and when they “crash” in the process of being coded, when embodied and material conditions interrupt their workflow?

Socio-cultural perspectives on literacy help track interactions between, among, and within students and teachers in the classroom and how principles in programming language shape both teaching and interaction. Meanwhile, socio-material perspectives on literacy will provide a lens on how humans interact with non-human actors. I write “within” to note how in the process of learning programming language students may also reshape their own identity as potential workers and members of their communities and their families. But this perspective will also help enlighten us on programming language’s value to participants as they pursue this literacy.

A socio-material perspective also provides a lens for describing how participants leave the camp and attend to the responsibilities awaiting them in the community or at home and the ways those responsibilities shape their commitment to learning programming language and working on the job. Then, finally, this study will trace the material consequences of gaining access to the workplace as a developer. Now owners of the literacy that empowers them, where do these students take their programming language? During the process of learning programming language, how do they experience larger forces in the material world—policy, relationships, money, health, whiteness— as helping or hindering them in their pursuit for upward social mobility?

One-Year (and Beyond) with African American Adult Coders

This dissertation describes a qualitative study conducted in the Spring and Fall 2017 semesters at Clearwater. Between these two semesters I recruited 12 participants and one 2015 Clearwater graduate who successfully started a career in software development. Most participants identified as African American, although some challenged “African American” as a term, calling themselves Afro-Latinx or “African *and* American.” Participant ages ranged from 21 – 56 and all save except one was from the Midwest. During my year-long qualitative study, I

visited Clearwater Academy weekly to conduct participant observation, working alongside students, when able, in various classroom activities, such as practicing coding on an interactive website and engaging in group assignments. Observation included documenting teacher-student and student-teacher and how these interactions assist in the process of learning workplace computational literacy. I conducted one-on-one and group interviews with participants and the class's two instructors—Richard, the technical skills instructor; and Janet, the job skills instructor and case manager.

Interviews covered a range of topics that I detail in the following three data chapters. Throughout both semesters, I met with participants individually to document their life history with literacy and digital literacy (see Chapter 2). As previous research has shown (Brandt; Selfe and Hawisher; Kirtley; Lachuk), life history interviews of participants help draw relationships between their social lives and the social structures that surround, impede, and, in some ways, support participants. Life history catalogs the ways social structures have followed participants and shaped success with learning how to code. Other interviews documented the strategies participants used to navigate barriers to learning coding literacy at Clearwater while also attending classes fulltime. These interviews highlighted participants' drawings of their support—people, objects, events, thoughts, and feelings that helped their address the challenges of attending Clearwater Academy (see Chapter 3). Finally, in order to understand the social and material consequences of learning coding literacy for these African American students, especially how participants used coding literacy for social mobility, I conducted follow-up interviews three to six months after participants graduated from Clearwater Academy (see Chapter 4).

I also conducted informal interviews, or conversations, with participants throughout class time. These conversations were to build repertoire as well as to understand to how they

responded to Clearwater Academy’s curriculum, how they viewed and constructed their relationships with their classmates and instructors, and their own expectations and goals for after completing the training program. With permission from the participants, I audio recorded these interviews. The majority of the time I wrote summary notes and fields notes of my observations and conversations. I also sought to collect assignment descriptions, guest speakers’ presentation materials, and some participants’ work, such as resumés, cover letters, and blog posts written for Clearwater Academy’s website.

This research project is an asset- and systems-based project. Instead of searching for deficiencies or pathologies with African American coders, this study considers ways of learning coding in the midst of stratification and the powerful moves used to keep up with a computer code bootcamp. This dissertation also explores how African Americans integrate coding literacy into their own lives, whether for social mobility or for personal survival or sustainability. For this reason, my interactions with research participants take an emic perspective—gathering and describing the participants’ interpretation of experiences rather than my own.

Computer code bootcamps are unique learning spaces for marginalized communities and have only begun to show its contribution to the coding for all movement. There may be some expectation that this dissertation suggests interventions for healing the personal problems of racially marginalized communities. As Leigh Patel points out, too often social science research studies why marginalized people fail while more affluent (often white) people succeed. This approach makes successful white people the norm and the standard to judge all other success. Suggesting interventions for “at-risk” people, Patel argues, only address the branches of the systemic racism tree. Social science research should follow W.E.B. Du Bois as a model for researching marginalized populations with the intent of revealing the ways societal structures

create inequality. This project, then, will in some ways highlight the computer code bootcamp and participants' relationship with these societal power structures.

Chapter Overview

In Chapter Two, “Like Coming Home”: African American Adults Playing and Tinkering Toward a Computer Code Bootcamp,” suggests students draw on personal legacies of informal “tinkering” (and not their experiences of racially biased schooling) to learn computer coding. Then, Chapter Three, “Between Learning and Opportunity: A Study of African American Coders’ Networks of Support,” demonstrates that to succeed in code bootcamp in a context of white supremacy necessitates a complex and often ad hoc network of social support, while Chapter Four, “Making the Pivot with Coding Literacy,” argues that even after successfully completing the bootcamp, systemic racism hinders their ability to succeed in the tech workplace, but outside of the tech workplace, participants create new ways to use computer programming in their lives. I close by reviewing primary findings, describing limitations of the study, and explaining next steps for this project.

Chapter Two, “Like Coming Home”: African American Adults Playing and Tinkering Toward a Computer Code Bootcamp,” analyzes digital literacy life history interviews from participants attending Clearwater Academy. The exigence for chapter is to establish a theory on why African American adults pursue learning coding literacy despite having little to no prior experience with it. I demonstrate that learning coding literacy is a continuation of participants’ tinkering with physical and digital technologies in childhood, and not their experience of racially-based schooling, which reduced digital technology to tools that supported conventional print-based writing. In this chapter, I suggest that scholars in education and literacy studies consider how African American tinkering in so-called resource-poor environments can inform maker education curricula.

Chapter Three, “Between Learning and Opportunity: A Study of African American Coders’ Networks of Support,” takes drawings of participant’s networks of support as a unity of analysis. To inform my argument, I bring together theory of ecological writing with critical race theory to show how participants learn and practice code in a system of racial disadvantage. I define networks of support are the combination of people, objects, thoughts, emotions, and past experiences that help students approach institutions of literacy sponsors—the story literacy practice begins. I demonstrate that to succeed in code bootcamp in a context of white supremacy necessitates a complex and often ad hoc network of social support. This chapter provide fertile ground for lifespan research on how networks of support change in a marginalized person’s life and how those changes influence their digital literacy practices. It also contributes to how studies on ecological writing includes more specifically institutional racism and its influence on the literacy materials marginalized communities gather.

I further analyze how African American adults use coding literacy as tools for social mobility after graduating from Clearwater Academy in Chapter Four, “Making the Pivot with Coding Literacy.” I show how opportunities associated with learning coding literacy for the workplace are often inflected with expectations to speak to and follow Discourses of whiteness. Despite having knowledge of web development and interest in learning more coding literacy, systemic racism may hinder ability to succeed in the tech workplace. However, I also show that even if participants do not begin a career in tech, they create new ways to use computer programming in their lives. This chapter’s contribution highlights the ways a computer code bootcamp may give students new ways to see or use their repertoire of literacies and that coding literacy’s power is in its variety of uses, especially in the personal lives of marginalized adults.

I close in Chapter 5 “Critical Imaginings of African American Coding Literacy Practice” by reviewing primary findings from this dissertation and engaging in a discussion of the study’s

contributions for various academic and practical conversations, teaching, and research. Finally, I note my study's limitations and directions for future research into the coding literate practices of African American adults in computer code bootcamps.

Conclusion

This research study contributes to understanding what happens when African Americans encounter new digital literacies in the context of computer code bootcamps as sponsors who have direct links to getting social upward mobility quickly and efficiently (as these sponsors claim). Most important for my research is what happens when the workplace embeds itself in their community to act as a bridge into Black liberation. This study examines the material consequences of learning coding from an accelerated training program for marginalized people. Understanding how adult African American students balance coding with other obligations, challenges, and triumphs helps us broaden what it means to teach emerging technologies, or uncommon literacies, whose promise is yet to be known.

Chapter 2

“Like Coming Home”: African Americans Tinkering and Playing Toward a Computer Code Bootcamp

In the past decade, computer programming has been called the new literacy necessary for everyday life. Echoing the historical and persistent myth that reading and writing “results in economic development, democratic practice, cognitive enhancement, and upward social mobility” (Graff and Duffy 32), computer programming may offer a range of individual and societal benefits: computer programming can be a tool for teaching students computational thinking—“a way of reasoning that compiles several high-level skills and practices that are at the heart of computing, but applicable to many areas far beyond computer science” (Flórez et al.), and computer programming itself is an advanced communication toolⁱ for lucrative job opportunities in information technology (Dishman; Rushkoff). Regardless of the desired consequences for these different pedagogical approaches, discussion on democratizing coding literacy can too often center the experiences of K-12 learners and undergraduates, leaving adult learners’ experiences with coding literacy unexamined.ⁱⁱ

While I agree educators and policymakers should closely consider what exactly youth and young adults should learn and why, we should pay attention to adult learners who are already perpetually updating their literacies “in response to rapid social change” (Brandt 75) to remain relevant in the economy. In other words, if employers come to value computer programming as an important or desired skillset, we should closely examine how exactly coding literacy is learned for the workplace. The need to grow repertoires of literacies matter especially for low-income racially marginalized adults, as they must navigate systemic racism that prevent their accessing new literacies that may promote their social advancement or their own survival.

In this essay, I address this concern by examining the digital literacy life histories of African American adultsⁱⁱⁱ who attended Clearwater Academy, a non-profit computer code

bootcamp that trains low-income women and people of color in computer programming and job skills. These participants indicated that they had learned little to no computer programming when they were young. Nevertheless, they sought to learn coding literacy in adulthood for social mobility and to better navigate a labor market that is gradually valuing programming as “the lingua franca of the modern economy” (Lohr, “Where Non-Techies Can”). A recent report showed that “across a range of industries” computer programming is highly valued if not required amongst employers. The report goes on to explain that “[h]alf of all programming openings are in Finance, Manufacturing, Health Care, and other sectors outside of the technology industry” (Burning Glass Technologies 7). Given the increasing value of computer programming, I offer a discussion of the preconditions of learning coding literacy later in life and the relationship of print literacy and procedural literacy to coding literacy.^{iv} Exploring these relationships will help us understand what exactly scaffolds learning coding literacy for adults and, in turn, help better our own pedagogical approaches to teaching coding literacy in Writing Studies.

Based on participants’ digital literacy life history interviews, I argue that digital literacy was the kind of computer-based literacy learned in school and it was associated with print literacy. Both literacies were withheld from men participants in ways that were racist while women did not describe difficult school relationships with literacy. Either way their literacy experiences with computers were not procedural. Ultimately, for reasons having to do with schools’ white supremacy and limited conceptions of procedural literacy, participants did not encounter procedural literacy in the otherwise literacy-sanctioned space of the school, implying that school-based digital literacy did not scaffold or promote the possibility of learning coding literacy later in adulthood.

This study reconsiders the contexts in which procedural literacy may be practiced. Ian Bogost calls procedural literacy “the ability to reconfigure basic concepts and rules to understand and solve problems, not just on the computer, but in general” (32). Similar to other scholars, Bogost argues that computer programming can be a vehicle for conceptualizing “the process of grammatical learning in general, and it helps create adults who are able to express themselves through technology” (33). He also contends, however, that “*any activity* that encourages active experimentation with basic building blocks in new combinations” teaches procedural literacy (Bogost 36 emphasis mine). Bogost writes elsewhere that these building blocks help students grasp at a process—“the methods, techniques, and logics that drive the operation of systems, from mechanical systems like engines to organizational systems like high schools to conceptual systems like religious faith” (Bogost, *Persuasive*, 3). In addition to manipulating pieces of a process, Bogost envisions procedural literacy as a way to read procedural rhetoric, or the ways processes combine to make persuasive expression (Bogost, *Persuasive*, 258). Videogames, notes Bogost, is a suitable medium for learning and practicing procedural literacy and procedural rhetoric; they do more with code and they are “uniquely, consciously, and principally crafted as expressions” (Bogost, *Persuasive*, 44-45). But the most useful kinds of videogames are those designed to “make arguments about the way systems work in the material world. These games strive to alter or affect player opinion outside of the game” (Bogost, *Persuasive*, 47). The primary location for teaching procedural literacy is the classroom (Bogost 260), leaving an opportunity to update the contexts in which we see procedural literacy at work.

This study examines the school and out-of-school literacies of African American adult coders as seen in their life histories of engagement with digital technology. As a result, this study converses with several existing inquires. The study speaks to scholarship that note the

different affordances of and tensions between racially marginalized people's home-based and after school literacies and school-based literacies, which tend to favor the language practices of White middle-class students. Studies on these conflicts advocate for the need to reconcile these two worlds to promote racially marginalized communities' well-being in a racist society (Heath), while acknowledging that after school literacy activities can offer academic and social capital-building opportunities that racially marginalized students may not encounter in schools (Watkins). Because this chapter often highlights the experiences of male participants, this study builds on existing research on Black men and literacy, especially work on using African American boys' interest in playing video games as incentives "to look inside the black box of video games to see the power of computation in their lives" (DiSalvo et al. 1). This study joins others in valorizing the language and writing practices Black men use that often get denied, ignored, or silenced in schools (Kirkland).

As an update to these inquiries into both print and digital out-of-school literacies and their consequences, this chapter suggests that participants demonstrate in their digital literacy life histories awareness that our digital infrastructure is a sociotechnical construction of digital processes that mediate every day communication. This real-world tinkering and play with a variety of physical and digital technologies, as seen in participants' out-of-school practices, call us to conceptualize technology as more than mere communication devices but rather as objects persistently active in our social environment. Procedural literacy may offer a pathway toward coding literacy, and the combination of these two literacies may offer users lifelong ways to control the activities these technical objects practice in their lives. Finally, this study implies that literacy scholars examine the ways funds of knowledge develop in racially marginalized communities when learning to navigate their digital ecosystems and their sociomaterial

conditions, how this knowledge can often involve procedural and multimodal literacy, and how these practices promote the development of other literacies later in life.

In the next section, I highlight how computer code bootcamps have placed themselves at the forefront of rapid training for careers in information technology. For some of these training programs, they not only desire to strengthen U.S. global competition and innovation but also to develop pathways toward “the realignment and rectification of social inequalities” through coding literacy education (Patel 90). Then I explain the context and data collection of my year-ethnographic study before describing the digital literacy life histories of participants. I highlight in particular the relationship among procedural literacy, digital literacy, and coding literacy in the midst of everyday racial conflict. Finally, I describe the implications for this study and call for research on procedural literacy practices among marginalized communities.

Coding Literacy for the Economy and Racial Justice

Information technology is vital to maintaining our present knowledge economy. As it proliferates and evolves, current working adults must learn and then re-learn literacy over a lifespan to accumulate and maintain one’s social and class status to remain relevant to the economy (Brandt). Computer code bootcamps participate in upskilling digital literacies by offering adults accelerated (14 weeks in-person; 15 weeks online) part time or fulltime training in computer programming. They also train students in soft skills, such as interviewing, conducting elevators pitches, and networking. Upon graduating, students may begin paid internships or full-time employment. These camps are sponsored by a range of national and local companies and non-profit organizations, such as Code.org. Code bootcamps must always update their curricula to match the needs and recent coding practices of the software developer profession; those bootcamps that fail to adapt or evolve their curricula according to the field’s standards will close. Dev Bootcamp, for example, was one of the first for-profit code bootcamps

to open in 2012 and had multiple camps throughout the United States before it faltered in maintaining a contemporary curriculum and closed in 2017 (Lohr, “As Coding Bootcamps Close”).

Computer code bootcamps have grown into a profitable industry in the United States. According to Course Report, a website that captures data on the latest trends in this industry, the number of fulltime computer code bootcamps grew from 67 to 106 between 2015 and 2018 and the industry has generated \$240 million in profit. A total of 20,316 students graduated from computer code bootcamps in 2018, a 20% increase from 16,867 graduating students in 2017 (Eggleston, “2018 Coding Camp”). The average tuition cost for in-person computer code bootcamps is \$11,906 (online courses, Course Report notes, are less expensive). Some bootcamps are eligible for federal student aid while others offer deferred tuition or income sharing agreements. The return on investment seems worth the money and time: some graduating students report increasing their salary by \$23,724 after completing the training (Eggleston, “2017 Coding Bootcamp Outcomes”). African Americans make up just one percent of computer code bootcamp graduates and reported an average salary of \$61,476 (Truong), up from \$43,300, the average household income for African American families as of 2014 (Pew Research Center).

Computer code bootcamps appear to be ideal sites for promoting diversity as they have fewer barriers to entry than higher education (Stewart). Camps that serve African Americans, and other people of color, like Yes We Code and Black Girls Code, help combat the prevention of racially marginalized people from equitable learning, or what education researchers call the education debt (Ladson-Billings, “From the Achievement”). This debt includes other disparities related to learning such as sending better financial funding to white affluent school districts, withholding the right to civic participation from racially marginalized people, and refusing to

pay moral reparations for failing to recognize the contributions of marginalized people. The combination of these structural debts ensure that the best learning opportunities will likely accrue for White students (Ladson-Billings, “Stakes is High”). The education debt continues its influence in coding literacy education but some possibilities to rectify the problem exist. For years, few African American students had taken Advanced Placement computer science in public schools due to a range of inequities: racist tracking methods, anti-intellectual peer pressure, and expensive fees for taking these courses (“More Blacks are Competing”). However, the College Board introduced AP Computer Science Principles in 2016, a high school course that teaches the fundamental concepts of computer programming and digital technology’s global influence. This new curriculum has garnered more enrollment among African Americans and has better prepared them for the AP Computer Science exam (College Board). This suggests that it is possible to repay the education debt through careful institutional and curricular inclusive practices that welcome racially marginalized students and their experiences with digital technology.

Similarly, computer code bootcamps as an institution can directly or indirectly serve social justice. They can distribute literacy as currency re-paid to the descendants of oppressed people. Bringing racially marginalized people into the tech pipeline,^v some argue, would alleviate socioeconomic stratification, close the digital participation gap between people of color and whites, and diversify an otherwise white male-dominated information technology field. The rhetoric of these bootcamps suggests that the path to racial justice is paved with programming and money. But what are the personal literacy legacies that scaffold such success for African American participants? And what does this teach us about coding literacy among African American adults?

Context and Methods

Context

This chapter addresses these questions using the digital literacy life histories of seven participants who attended Clearwater Academy during spring 2017. Located in the Midwest, Clearwater Academy is one of many services provided through a non-profit organization whose mission is to combat racism and poverty in the local community. To this end, Clearwater trains low-income people of color and women in front-end web development (HTML, CSS, and JavaScript) and soft skills. Over the duration of three and a half months, students must attend class on time four days a week, eight hours a day. Students learn how to design and code websites in and outside of class on their own, in pair programming, and in teams. In addition, students tour local tech companies and speak to information technology professionals. To solidify their success on the job market, students also learn to write cover letters and resumes, present elevator pitches, and practice mock job interviews, and guest speakers visit to discuss a variety of job-, finance-, and tech-related topics.

Clearwater's students do not pay tuition or a fee and are lent MacBook Air laptops from Clearwater that students can keep upon graduation. After completing their education, students may find work as freelance web designers or complete a paid 3- to 6-month internship. Ideally, if Clearwater graduates do well on their internships, their employers may ask them to return as full-time hires. Although getting work in tech is the ultimate goal for Clearwater Academy, any kind of full-time work counts as a success. The instructors and program coordinators provide mentorship to ensure students choose the track that best fits their technical skills. In 2017, when this study took place, Clearwater instructors postulated that, based on their own research, they were the only accredited computer code bootcamp in the country. Thus, students can use their certificate from Clearwater for college credit at a local community college.

When this study began, Clearwater had graduated four cohorts since it opened two years prior; however, Clearwater faced two challenges. First, students arrived contending with structural barriers in their lives that forced many to leave the bootcamp: homelessness, substance abuse, healthcare, family responsibilities, and financial hardship. In addition to attrition, Clearwater sent some of the few students who did graduate to intern for local companies' information technology departments. Although these students had been trained in computer programming, they were still unable to adapt to the workplace culture's emphasis on practicing whiteness.

As a response to the barriers that lead to dropping out of the program, Clearwater directed its few resources toward protecting and assisting students with these challenges as much as possible. Clearwater gives students gas cards and bus passes each week, and it finds services that may help students with childcare and housing. Occasionally, Clearwater will help pay one month's rent. To better prepare students for the culture of whiteness in the tech industry, Clearwater developed a new curriculum that emphasized teaching soft skills and expects students to behave as workers in the classroom. Since Fall 2016, Clearwater Academy has regained its reputation in the local tech industry: they no longer have to recruit or find students for class, students come to them or are referred to Clearwater by their graduates, and employers send job announcements to instructors that students may apply for.

Methods

The spring 2017 class, the focus of this chapter, was the fifth cohort of admitted students. Clearwater recruited nine African American adults, seven of whom volunteered to share their experiences with me before, during, and after the bootcamp. Ranging from ages 21 to 56, six participants self-identified as African American and one participant self-identified as "African and American." Five had grown up in the Midwest and one was born in Sudan but

spent much of her life in the Midwest; another participant moved frequently around the southern region of the United States before settling down in the Southwest. Of these seven, only three participants—DeAndre, Nadaline, and Isaiah—recalled working directly with programming at home or at school on their own. These brief encounters seemed formative for their attending Clearwater. Nadaline, the participant from Sudan and who identified as “African and American,” was in the 5th grade going on to the 6th grade when she started chatting with other users on America Online’s instant messenger program AIM. She learned that some users had created their own websites. “They were just giving out links to their website,” Nadaline recalled, “and I looked at their website and I asked, ‘How did you do this?’ And somebody helped me out.” Nadaline played around with HTML for a while but soon took on other interests in grade school: writing for the school newspaper, practicing the cello, and making art. When I first asked Nadaline to participate in my study, she told me that coming to Clearwater Academy was like “coming home.”

In addition to discovering when and where coding entered someone’s life, the anecdote above shows the insights we can gather about literate practice when collecting participants’ life histories with digital literacy. The literacy life histories of individuals help to understand the social history of literacy or to understand a specific social use or pursuit of literacy over time (Brandt). Using this method of data collection, scholars have been able to reveal how undocumented immigrants’ movement cross borders into the United States changes their literacy practices (Vieira), how people change their sociomaterial literate lives post-aphasia (Miller), how literacy empowers African Americans living in the rural southeast (Lachuk), and how the writings of historical queer writers influence the literate lives of present-day queer African Americans (Pritchard).

In this study I drew on digital literacy life history interviews to discover if participants had encountered coding earlier in life and determine how that exposure played a role in their decision to attend Clearwater Academy. If participants had not learned programming before, interviews could show what else in their literacy life history scaffolded their learning coding literacy in adulthood. What I present below highlights a complex relationship between digital literacy and procedural literacy and how they are mediated by racist ideologies in school and sociomaterial conditions at home. These relationships played significant roles in shaping the quality of participants' digital literate lives as preconditions toward learning coding literacy.

When Digital Literacy Means Print Literacy

I present two kinds of empirical evidence that suggest why school-based digital literacy does not easily scaffold learning coding literacy. First, participants learned to use computers to research and write essays in school, but rarely encountered coding literacy or procedural literacy in classrooms. Second, participants felt disenfranchised from the digital literacy and print literacy they learned in school because of racist perspectives on the toolbox of literate abilities they brought to the classroom. Their stories demonstrate that a strict focus on how to write digitally constrained them from exploring what computers can do and that a racist or unwelcoming learning environment contributes to their devaluing educational institutions as safe spaces for their well-being. This evidence, drawn from participants' experiences, indicates the ideologies around literacy and technology their schools promoted, ideologies that were especially limiting to African American students given race and class-based inequity among public schools. This section implies for my larger argument that educational institutions may promote an approach to learning digital technology that reduces technologies to mere communication devices that work similarly over time rather than as objects that evolve and emerge constantly and thus need to be learned and re-learned over a lifetime.

I focus on participants' schooling because schools are instrumental in teaching digital literacy widespread. Scholars in computer science believed programming would offer children new ways of learning, thinking, and problem-solving and public schools would be poised for this education (Papert; Kay and Goldberg). The graphical user interface, however, made using computers easier. Instead of working directly with complex code, the user could use various software programs to point, click, and type to complete daily tasks. Few people learned how to write programs for computers, but the return of mass coding literacy today gives schools another chance to influence how people use or don't use programming in their lives.

The digital literacy life histories of African American participants from Clearwater show that very few encountered programming in public schools, if not at all, and they hardly used computers for exploring the power of computing. As an adult, Kevin, a 28 year-old educated in culinary school, valued using his phone to write "a ton of notes. From poetry to just my thoughts. Things that I don't feel comfortable saying, I just write it down." Despite all of this writing he did during the study, Kevin said during our interview, "I've never been a writer. Like a writer. English was my worst subject." Kevin summarized his entire experiences with digital literacy at school during the 1990s in the following way:

Okay, at school, they really didn't reach us how to use computers. It was more like educational games. Writing games. Spelling games. And then on occasion you had those *Oregon Trail* days! . . . In middle school, high school, we had computer classes but it was . . . word processor . . . How to use PowerPoint. How to use Excel."

Kevin learned how to be a proficient writer using computers over time. Educational games helped him learn spelling in elementary school, and then he learned rapid touch-typing to draft

essays and reports in middle school. Kevin didn't count thing the writing he did in elementary and middle school important was real, as teachers assigned short pieces that covered topics like "What did you do over the summer?"

"Real" writing for Kevin began in high school when teachers taught him how to do research using library databases and integrating scholarly sources into his papers. But even these assignments frustrated him: Kevin was required to write "3 pages, 6-page, 8-page papers" on mundane topics or topics irrelevant to himself: "Like American history stuff. You get someone like Benjamin Franklin. 'Write a three-page paper on Ben Franklin.' And you're just like, 'Why? Why would I want to do this? This is crazy.' In high school, I felt like that a lot. 'Why would I want to do this?' This is crazy." The link between digital literacy with digitalized print literacy deepened as Kevin described difficulty attributing credit to sources gathered from the library. When learning research writing, Kevin said he seemed to have missed the lessons on plagiarism. In his freshmen year of high school, Kevin would "turn in papers copy and pasted from websites and stuff. No reference . . . They'd[teachers] give me credit for them. Boom, boom! Credit!" But it wasn't until later in the second half of class Kevin learned from his teacher about citing sources and giving credit, as if she had been teaching them citation since the beginning. When Kevin learned he was supposed to attribute ideas in his paper to their authors, he was more dismayed and left wishing he had known about plagiarism earlier. Nevertheless, we see here an example of exploring multiple, mainly text-based resources that, thanks to the copy and paste feature of the computer, easily lifted and placed onto another text-based research paper. Regardless of the context, Kevin reported using computers heavily for writing print.

Kevin did recall learning more "advanced" features with PowerPoint, Excel, and Outlook in computer classes. Under financial constraints, his family moved from one school

to the next between the South and in the Southwest, and during those travels Kevin never encountered programming language in any of his schools. Like other participants' life histories with digital literacy, much of Kevin's work with computers involved typing alphabetic text on a computer screen, and never looking behind the interface to see what was in the black box that made it all work.

In the 2000s not much had changed in the schools 21 year-old DeAndre attended in the Midwest. DeAndre split his formative years between a major metropolitan city and a mid-sized city. His family never owned a computer, so his first encounter with a computer was in school, recalling using an iMac in the second or third grade but, again, mostly learning how to type. DeAndre recalled inconsistencies between his second- and third-grade curricula in the mid-sized city and his sixth-grade curriculum in the metropolitan city. "After 5th grade was when I left from the white school system and went to the [metropolitan] school system," DeAndre explained. "And I was learning in 6th grade what I had learned in the 2nd grade." As a third grader in the white school system, DeAndre learned how to use a MacBook and the basics of writing with a computer. Like Kevin, he learned how to use Word, PowerPoint, and Excel. He even learned how to structure essays from writing assignments, such as one that asked him to imagine how he would run a school if he were its principal. DeAndre learned how to type using an educational software called Type Time Machine, which helped accelerate his typing skills in elementary school: "We had our own [keyboards] . . . And they started teaching us all that shit. And when it went to this and then it went to that, and then eventually I was just fast with it. Just moving around without looking down."

The metropolitan city school had a majority Puerto Rican student population. Moving to this school, DeAndre felt he had gone back several steps in his learning. This school didn't receive computers until he was in the 7th grade. Even then, the computers were outdated. "It was

one of those old ass fucking police monitor things that sit up in the car,” DeAndre recalled. “And I’m like ‘What the fuck is this? This’ll break. You guys don’t have Apple?’” His teachers explained that they didn’t have the money for new computers. “I’m like ‘Uhhh, okay. All right.’ I didn’t know.” As an adult looking back on his formative years, DeAndre wondered why the 8th grade curricula was on a second-grade level, why he needed to relearn what he knew six years before.

DeAndre and Kevin had first-hand experience with how educational resources are unevenly distributed across school districts entire states, based on the class and race of school funders. Even though DeAndre and Kevin moved from school to school, they always seemed to arrive at the same destination: schools teaching how to transfer writing on paper to writing on the screen using the most common and basic software—the Microsoft Office Suite. Their schools taught them enough to navigate the basic demands for using digital technology that would make them suitable workers and citizens, but these practices and contexts did not help participants conceptualize *other possible abilities* of digital technology. Instead, for participants in this study, digital literacy education saw computers as devices for textual production and consumption. This way of thinking about technology may fail to promote a digital ecosystem exists made up of a series of networked functionalities that require a diverse set of literacies possessed by different kinds of people that shape how others access various goods, resources, and services in their lives. Racially marginalized students’ limited knowledge of such complex connections confine access to what they themselves as users can do: they may be considered for low-level, menial workers given less opportunity to join highly competent, creative, and critical class of literate subjects that determine the logic of these technological processes.

Ideologies on Literacy and Race Limit Learning

In this section, I describe participants' mixed experiences in school that showed how ideologies on literacy and technology actively limited or withheld learning from participants. Teacher instruction and perception and hostile school cultures were significant players in shaping how participants' engaged with school-based digital literacy. If the section above identifies narrow conceptions of digital literacy practice, the following paragraphs here uncover ideologies on whose literate practices matters and who is worthy of accessing new literacies. First, I offer examples on how sociocultural forces such as race, class, and the limitations of meritocracy helped withheld crucial digital literacy experiences from participants.

DeAndre's jump from an advanced curriculum to a slower curriculum marked him as an exceptional academic student. He later leveraged his academic success for profit: from sixth grade to eighth grade, teachers gave students one hundred-fifty vocabulary words and exercises. DeAndre would complete these problems in the workbook and then sell the answers to his classmates. For three years. Every year. DeAndre was so ahead in his other classes that school teachers and administrators wanted to promote him to a higher-grade level. DeAndre and his mother refused the option; he reasoned that although he knew "the whole damn grade," DeAndre could benefit from learning material he may have missed or didn't learn.

Despite his learning digital literacy early on at one school, and doing well in school overall, his academic success negatively impacted DeAndre's peer relationships. When they learned of how well he was doing in class, DeAndre's classmates asked him for tutoring, but this request was a ploy to beat him up out of jealousy and frustration. Getting bullied for advanced digital literacy, and for doing academically well in general, had considerable influence on DeAndre's motivation to continue doing academically well. Caring too much, DeAndre explained, only got him "whooped." To protect himself, he stopped caring about others and

started looking after for himself. DeAndre quit high school when he was seventeen and took up carpentry instead. Full-time work satisfied what DeAndre called his hunger for money.

In his life history interview, Alex, 36 year-old who had served in the Navy and worked in various jobs in radio and finance before coming to Clearwater, admitted he didn't remember much from computer classes in school. He associated computer labs with hearing the O.J. Simpson case's final verdict and the excitement he and the few Black students in the school felt. What came to the surface of his memory was teachers' negative perceptions of him because he was African American, and this racism became more visible when Alex refused to engage with what he considered an easy curriculum.

Alex told vivid positive accounts of literacy sponsors in his childhood. In second grade, Alex had his first and only African American teacher who "put books in my hand and put stories in my mouth. And she sat me down in front of her class as a second grader and allowed me to read to her first graders." To him, his second-grade teacher was "exceptional . . . She's magical." This experience in school overlapped with his home literacy. His mother loved books just as much as his father did. She encouraged Alex to read and build his repertoire of words by giving him a new vocabulary word each day before going to school. He remembered being a ferocious reader himself, even jumping into *Othello* by the sixth grade. The language to him was "weird," so Alex read slower to comprehend the play. By the time he started middle school English classes, Alex was a well-read student. However, his time in middle school was the opposite of his learning literacy at home. For Alex, he could do well in sixth grade English. As he explained, Alex "could debate these points" and "express these theories," but he didn't want to because he didn't "appreciate" the "slow" curriculum. Because the work was so easy to him, Alex "clowned" around and "tested badly." His behavior backfired, as his teacher mistook his lack of engagement with school for being "remedial." He recalled teachers telling him, "You

slow! So we're gonna put him in remedial reading.'" Although they claimed he was remedial, Alex believed he was placed in this class because he was African American and "gregarious." Nevertheless, Alex was so embarrassed that he plowed through the work in the remedial class and "by the last semester they had moved me to advanced [reading]."

Like Kevin, Alex had spent his formative years moving from school to school. To his eighth-grade teacher, this constant moving suggested Alex had been expelled from these schools for misbehavior. But what the teacher didn't know was that Alex moved because money was tight in his family. Nevertheless, his teacher told him that he would be a problem in her class. Racism mounted further later that year. Alex and his African American friends were walking from the school buses when a white student came from behind and said, "Move out the way, niggers." When Alex and his friends reported the student, the school administration sympathized with the white student's claim that they were "scared because you're Black." From that point, Alex stopped caring about school; the remainder of his education, school was "about smoking weed and it was about having fun. And skipping class." A few years would pass before Alex came to what he considered his next significant educational opportunity after leaving the Navy: a community college in Iowa where he studied broadcast radio, but financial difficulties would later force him to drop out.

Both DeAndre and Alex encountered at the outset of their childhoods affirmation of their literacies. However, later in their schooling they encountered students and teachers that undervalued their knowledge. The system of racial and class stereotypes marked Alex unworthy of empathy and care. Constant migration from one school to the next due to financial difficulties "while Black" attracted cultural beliefs that influenced his undervaluing the worth of learning school-based literacy and digital literacy. For DeAndre, academic meritocracy was limited. His academic achievement, thanks in part to a relatively advanced curriculum in one

school, marked him as a talented scholar and a source of derision. Doing well in school did not reward him the accolades he expected. Earning nothing from his work in school, DeAndre quit high school and worked in construction. Ultimately, what I find is that participants like DeAndre had difficulty accessing digital literacy because of its association with racist school practices.

School-based digital literacy has slippery characteristics that make them difficult to lead to desiring to learn coding literacy. In fact, it's difficult to separate print literacy from digital literacy as a technology but also in that the problems of writing on paper tended to transform into new problems on computers. In whatever form literacy took, it was too baffling, opaque or demanding to promote trying out coding literacy. In addition, institutional support of racism and classism and the limitations of meritocracy makes education and digital literacy unattractive or a trap. These barriers did not prevent Kevin, Alex, or DeAndre from becoming literate subjects, as we can glean from their own admitted engagement with reading and writing outside of school, but it does mean their literacies within schools were limited or withheld. Pathways toward coding literacy, then, may be stifled without additional avenues for exploration, play, experimentation.

For this reason, I show in the next section that participants recounted their learning about computers in their homes and communities as procedural literacy. In other words, in these supposedly resource-poor environments, participants actively sought out what computers were capable of or how they worked. Their life histories suggest that desiring to know how computers worked or what they could do may scaffold the desire to learn coding literacy in adulthood when that option becomes available to participants.

Tinkering as Literacy

In this section, I explain that participants recalled fond memories of exploring, discovering, and experimenting with technology in their homes, communities, and the workplace. Across each of these contexts, different kinds of people appear that encouraged this self-motivated interest in tinkering and play as brokers of procedural literacy: family, friends, strangers on the Internet, and employers. We will see that even in so-called resource-poor environments, different funds of coding knowledge can blossom from the technologies available to participants.

Rosie was born in the 1950s and recalled regular updates in her print literacy throughout her formative years. She moved from writing her name on blue-lined trace paper at home to learning how to use a typewriter in high school. Rosie had always wanted to go into information technology, but after graduating from high school, Rosie she was called to work to support her family. This decision, however, was only the beginning of her learning to play with technology, as Rosie's work experience exposed her to learning how computers worked. Rosie got a job with the state legislature as a clerical worker in the late 1980s doing "word processing." She worked in a large room with rows and rows of desks where she and other women typed and revised letters, memos, and statutes based on instructions from their superiors that were either given in-person, on paper, or through audio recordings.

Rosie's word processing is similar to the word processing that younger participants learned during the 1990s and 2000s; the difference is that typing required that Rosie put in commands for the computer, such as pressing shift and another key to instruct the computer to start a new paragraph. Still, she suggests that training involved more than simply learning word processing. The state government trained Rosie in how to use a computer and to understand "why it was doing what it was doing," what was happening when the computer was commanded to print or calculate. Rosie was not learning how the computer code worked, but

she seemed to be learning more than the rudimentary functions of computers. Learning why a computer works the way it works is an explicit nod to procedural literacy; that is, Rosie learned the building blocks of computers to conceptualize their design and function in her mind.

Other workers reacted to the new computers with confusion or ambivalence. “Some people were like, ‘How do . . . that’s just too much to be working with. You sit there and type?’” Rosie explained. While her co-workers felt uncomfortable using computers, Rosie had relished working with them. In her interview, she explained, “I knew when I started that job I just felt comfortable. I wasn’t intimidated. I learned so much from the content. That’s another thing that intrigued me, too, you know. The fascination of the computer itself; what it was doing. How it was manipulating what I was doing to some paper. Things like that.” The state legislature sponsored additional training with computers so that Rosie and the other clerical workers remained updated on their technical skills to meet the state’s needs: She had the opportunity “to be on a team where we created software packages that we wanted to be in-house to do what the software [did] to make our jobs better.” But these trainings only inspired her to learn more about how computers worked. When Rosie had time between work and taking care of her family, she would attend night classes at the local community college and, years later, take online courses.

This exploration of computers was a thirty-year journey for Rosie. Just before attending Clearwater Academy, she had nearly finished a certificate in information technology from a community college in town. But Rosie couldn’t do a required internship to complete the program because she had been diagnosed with lupus and took early retirement. Clearwater Academy was at once a continuation of her learning about computers as well as a pathway to finally getting her certificate. Rosie hoped that after graduating from Clearwater she could go on to complete an internship that the college would count as fulfilling the certificate requirements. Rosie’s procedural literacy practice may indicate her being a lifelong student. This is certainly

true, but it was not economic demand or being able to use computers for everyday tasks that drew her to constantly learn about digital technology's capabilities. When I asked Rosie if she thought it was important that her children know computers for their everyday lives, for example, she shrugged and replied, "I didn't think of it like that. I always thought it was fun."

The idea of enjoying how things worked was an ongoing theme for other participants, like Kevin. Kevin also explored analog and digital technology at home and in his neighborhood. But he lived around "a lot of negativity" growing up. He remembered his family losing their house, and then moving from apartment to apartment around the South before finally settling down in Arizona. He spent most of his life between Arizona and Mississippi, although Kevin was more likely to claim Arizona as home. In Mississippi, Kevin explained, he and his siblings "just lived. We had our fun times and stuff. We survived." They put "much brain power on anything that could happen positive." Print literacy and digital literacy became ways to escape poverty and racism while growing up in Mississippi, and they helped Kevin bond with his siblings, like competing with his older brother in fighting video games, joining his family for board games, or reading *The Hobbit* with his older sister.

His healthy family relationships played a prominent role in his figuring out how things worked. Once they recognized he had an interest in something, they began providing resources that Kevin could explore on his own. His literacy history before getting a computer, for example, is peppered with tinkering, exploring, and figuring out how different types of technologies and symbols worked. Kevin's earliest memory of reading and writing was actually watching his older sister complete multiplication homework when he was in the fourth grade. Taking a peek at her homework, Kevin became fascinated by how numbers grew exponentially:

“I mean being able to transform something into something greater. That was always my thing. I’ve always been one of those kids who takes a part their roommate’s TV remote to see how it works. Took it apart just to see if I could put it back together. Just the magic. To me it was just magic because I didn’t know how any of this stuff worked. My sister was on multiplication stuff. I was like, ‘What? This doesn’t make any sense!’ I guess it was like turning nothing into something . . . I had a high interest in it. I was always intrigued by that.”

When his older sister realized Kevin was fascinated by the magic of numbers, she let him read her math textbook and showed him what else numbers could do.

Pencils and papers did not often circulate in Kevin’s household, but he recalled science boards being significant to his procedural literacy practice. These games allowed Kevin to run different kinds of experiments, but they were, for his mother, less about playing as they were about preventing Kevin from taking his sense of exploration too far in the household. His mother bought these science boards often so Kevin could “tinker” with them, and “If she didn’t,” Kevin said, “I would eventually start taking apart the remote and stuff like that . . . I was interested in how things worked.” But there was a limit to what else Kevin could take apart. For several Christmases Kevin asked his mother to buy a computer. When she could finally afford to purchase a desktop and computer processing unit (CPU), Kevin was curious how they worked. So he took the back cover off the CPU and looked down inside it. “And my mom walked in. And she was angry. I remember getting spanked . . . I never got spanked for taking apart the remote, expect when I couldn’t put it back together. I just knew it was serious.”

His tinkering would continue in Arizona in his last year in middle school. “That was around the time we started messing with hardware,” Kevin said. “Like computer hardware . . .

That's when I got into computer tinkering." He met a friend who would visit a waste facility and find computer parts to assemble into his own computer. For Kevin and his friend, playing with hardware was less about discovering a career in tech than pursuing a hobby. Something fun to do. Messing around with hardware extended into his playing video games. As I wrote earlier, Kevin played video games with his brothers and, by all accounts, was still a gamer in adulthood. During our interview, Kevin expressed his excitement for the Xbox One's new controllers that players could take apart and customize. For an adult who grew up taking apart computers in his home, the controllers was a dream come true, and confirmation of the procedural literacy others like Kevin enjoy. "I was a tinkerer. I still am," he said, reflecting on the broader implications for his interest in the customizable controller. "I've always been. I like to mess around."

Rosie and Kevin relied on communities, families, friends, and their own self-motivation to practice tinkering, exploring, and playing with digital and analog technologies. Rosie's workplace trained her on how computers worked, which was a significant amount of fuel added to her lifelong interest in figuring out technology's abilities; the workplace, a key location for adopting new technologies for productivity, became an unexpected literacy sponsor for Rosie. Meanwhile, Kevin approached technologies with his own curiosity that family and friends later noticed and proceeded to cultivate as brokers of procedural literacy. Rosie and Kevin were just examples of drawing on overlooked funds of coding knowledge that can exist within African American lives. When we consider the ideological differences between school-based digital literacy, as well as print literacy, and participants' own procedural and digital literacy practices independently, stark implications for teaching coding literacy and procedural literacy rise to the surface, which I tease apart in the conclusion.

Conclusion and Implications

In their interviews with me, participants did not encounter coding literacy early in childhood, or if they did, they did not have extensive exposure. Digital literacy education for these adult learners of coding focused on typing documents on the computer in their formative years, not exploring computers' potential problem-solving power; this school-based digital literacy was less likely to scaffold coding literacy learning for participants later in adulthood as they brushed up against racism and ideologies of literacy that limited their access to more advanced interaction with digital technology. Contrast this experience to their homes and communities which, although relatively resource-poor, nevertheless encouraged their practicing procedural literacy. Participants reported playing with both analog and digital technologies, sometimes encouraged by institutional literacy sponsors, such as the workplace, or by brokers of literacy such as family members and friends. In their stories, participants expressed more excitement and interest doing this self-sponsored play. Digital inequality, created by institutional stratification, seemed to have created alternative ways of promoting procedural literacy. Building on Bogost's argument that procedural literacy isn't confined to computer programming, digital literacy life histories show that the act of play and discovery, not necessarily interacting with computers themselves, may scaffold a pursuit of coding literacy.

These findings suggest that Writing Studies scholarship challenge racist perspectives that African Americans suffer from technophobia; in this formulation, black cultural identity embraces a "gritty reality; not virtual reality" and operate as "living links to the lost world of unmediated spontaneity, deeply felt physicality, and social connectedness" (Dery 34). They actively resist civilized tools of convenience, economics, and communication. Digital inequality solutions rely on this "pathological rhetoric" to drive African Americans out of their primitive state and into the fold of White techno-literate society. This research builds on existing scholarship that show African Americans in fact "suffer" from technolust. Other scholars and

practitioners of digital media have shown the tight relationship between Blackness and technology: African American communities have always found ways to access new technologies and pull these digital resources together for collective artistic, cultural, and political engagement, and racial uplift (Banks; Everett; Driscoll). The ways African Americans use digital literacy may include Afrofuturists' efforts to "[challenge] both the implicit whiteness of nerds and the explicit technological absence of both realist and romantic black essentialisms" (Eglash 60). These collective efforts recall Ashleigh Greene Wade's argument that using technology to make Blackness viral is a kind of world-making. Viral blackness, she writes, is a "deterritorializing mode of subversion to white supremacist systems that seek to restrict the movement of Black bodies, silence Black voices, and quell Black thought" (Wade 36). African American digital literacy practice can be a response to systemic racism or poverty, as suggested in participants' interviews. Any effort to democratize coding literacy may include instructional practices and curricula that unpacks how coding literacy contributes to sociocultural world-building. This expands the goals of teaching coding literacy to include social mobility, inclusion, and computational thinking as well as how coding literacy may help re-distribute power among racial and ethnic communities.

This study identifies procedural literacy as not only a learning goal for formal education, as Bogost suggests, but also a practice that can be taken up in other contexts. Participants' interviews suggest there's rich funds of knowledge. For this reason, literacy scholars may examine the circumstances under which racially marginalized people tinker or explore. Additional qualitative research on how marginalized communities use sets of processes for learning and then enacting persuasive expressions across different physical and digital environments may offer fertile ground for further theorizing links between racism, writing, and digital technology. However, investigating the ways racially marginalized people create learning

and literacy practices in their communities amidst racial inequality is not meant to suggest that racism is a *social good*; it does ask scholars to re-focus what kind of environments foster literacy learning for survival or sustainability and the long-term consequences of these learned practices for accessing the knowledge economy.

Notes

ⁱFor rhetoric and composition, computer programming is a type of writing (Vee). Rather than teach computer programming itself for career in software development, writing instructors would expand students' rhetorical awareness of coding's many affordances (Sample and Vee) (Brock and Lindgren).

ⁱⁱ In *Learner-Centered Design of Computing Education* Mark Guzdial suggests that the best way to democratize coding is through K-12 education. Computer science undergraduates, he notes, should be encouraged to seek careers in public school teaching rather than industry. These graduates would then help address the lack of formally trained computer science teachers, one key barrier to cementing widespread coding literacy education.

ⁱⁱⁱ An intersectional analysis of race and gender would provide more insight on participants' digital literacy life histories. However, in this chapter I catalogue the unique barriers to racial or ethnic identity and the role those barriers play in coding literacy learning. See Rachel E. Luft's "Intersectionality and the Risk of Flattening Difference: Gender and Race Logics, and the Strategic Use of Anti-racist Singularity" for more on the benefits of using single-issue tactics in social justice training.

^{iv} Coding literacy is the ability to read and write in computer programming languages. Procedural literacy is "the ability to reconfigure basic concepts and rules to understand and solve problems, not just on the computer, but in general" (Bogost 32). Print literacy refers to the practice of reading and writing alphabetic text on paper. Finally, digital literacy, based on participants' interviews, is ability to use computers to type alphabetic genres, primarily essays and research papers. Following New Literacy Studies tradition, I believe each kind of literacy is a sociocultural practice inflected with ideological beliefs about what they should do and who may benefit from these tools.

^v In *Digital Dead End: Fighting for Social Justice in the Information Age*, Virginia Eubanks observes that the "tech pipeline" often means "highly paid technological positions such as tenured faculty positions at research universities or managerial positions in high-tech industries" (27). She reminds readers that the tech pipeline also includes low-wage positions that support tech industries.

Chapter 3

Between Learning and Opportunity: A Study of African American Coders' Networks of Support

Despite the ubiquity of digital technologies in the United States, racially marginalized adults are still less likely to develop high quality digital literacy skills for meaningful problem-solving practices (Reder 16). Racial disparities between Whites and African Americans in health, education, income, and mass incarceration (Geronimus et al. 826, 833; Sentencing Project; Jones, “Does Race Matter,” 150–51; Martin, Fasching-Varner, and Pulley) can exacerbate the unequal use of these technologies and prevent racially marginalized people from accessing the multiple resources that may ensure their learning digital literacies will afford “full participation in life opportunities” (Warschauer and Tate 69).

The expected benefits of digital literacy echo the historical ideological belief that reading and writing are necessary to individual and societal progress, what Harvey Graff calls the literacy myth (Graff, *The Literacy Myth*; Graff and Duffy 41). The literacy myth interprets literacy as a useful skill that’s free of the messiness of political and cultural ideology. It persists and extends into digital technology, which surmises that the digital divide is a dichotomy between those who have access to the Internet and those who do not. If given physical access to digital technology, racially marginalized people can overcome inequality and have full participation in life.

However, scholars in Writing Studies have found literacy a socially constructed practice whose consequences vary among people as a result of cultural and political interests (Street 2; Graff, “Myth at Thirty”). Understanding literacy as a social practice, a range of scholars in composition and rhetoric have called for a more nuanced discussion on digital divide rhetoric and education policy (Banks 41; Selfe; Moran 206). Annette Powell, for example, argues that we look deeper than mere physical access to digital technology and “recognize [the] social, political,

and economic factors implicated in the literacies individuals bring to technology and the circumstances under which these literacies are deployed.” How racially marginalized people access and use digital technologies is an *on-going process* deeply affected by their social conditions (Powell 17). Virginia Eubanks notes that we consider that digital technology is “an assembly of practices for organizing the world that encode some norms, values, and ways of life at the expense of others.” Rather than teaching a skill set, digital literacy education should build awareness of how social location, citizenship, and surveillance play a role in how marginalized people interact with technological systems (Eubanks 25, 30).

Computer code bootcamps offer a unique context for further research on the social and material conditions that influence marginalized communities’ access to emerging technologies and on the realistic, complicated consequences of those processes of access. These bootcamps emerged from an ongoing nationwide campaign for giving all youth, especially underrepresented youth, opportunities to learn programming in public schools, as coding may be the new necessary literacy for everyday life (Smith; Treene; Bramson; McGowan; Bruckman et al. 86; Guzdial 10). Assisting in this goal to make coding literacy more accessible, computer code bootcamps offer rapid and short-term (approximately twelve weeks or less) training courses in web or app development for, in some cases, low-income racially marginalized people. Some bootcamps that meet in-person require students to dedicate thirty-two to forty hours a week to learning coding, leaving students with less time for tending to other life responsibilities such as work and taking care of loved ones. The results of this intense training present a new coding literacy myth: an imagined future in which coding literacy education addresses digital racial inequality, helps low-income people of color have a lucrative career in software development, and evolves the tech industry into a more culturally inclusive space.

However, my qualitative study shows how conditions on the ground actually play out as twelve African American adults attend a code bootcamp in the Midwest and become educated under the regime of white supremacy. I asked participants to draw detailed maps of support illustrating the various kinds of internal and external support that they relied on as they progressed through the bootcamp. I then conducted one-on-one interviews about their maps of support, asking participants to explain how those kinds of support assisted in their learning coding literacy. Using principles of ego network analysis (ENA) to analyze these maps and interviews about these maps, I was able to learn that participants called on these various clusters of support in their network to provide the personal resources coders need to code, resources that are hard to come by in situations of racial injustice. Learning coding literacy occurred under drastically different sociomaterial conditions such that each participant seemed to follow a unique process of gathering their resources. For this reason, I describe in this chapter three loosely connected example resources that shaped participants' learning coding: emotional investment, temporary time and space, and mind/body preparation.

Integrating Ecological Writing and Critical Race Theory

This study is grounded in ecological theories of writing and critical race theory. Ecological theories of writing “understand how networked people experience multiple encounters with a variety of other people, texts, and objects overtime” (Laquintano and Vee 53). In this new materialist formulation, writers are enmeshed in, and not the center of, a complex, constantly evolving system of relationships between the social and the material (Cooper 371-72; Alexis 84-85; Syverson 23). Qualitative studies have investigated writing ecologies from a variety of angles. For example, Cydney Alexis conducted a qualitative study on writing habitats and how objects in writers' preferred locations for writing mediates their literate activity. And,

more recently, Yvonne Teems teases out the ways we can learn about how older adults' writing ecologies shape their literacy practices for accommodating aging.

However, writing ecologies can implicate large macro-level forces that bring materials into our literate lives. In her chapter describing writing habitats, Alexis writes, "These habitats are constructed for *us* when we are young . . . we construct our own spaces in which to work that often echo the spaces of our early learning" (84–5). As a reader I'm curious how the space and objects that make up writing habitats indicate the writer's privilege; in other words, how have economic disparities among racial groups help construct writing habitats? For example, in his digital literacy life history interview, 27 year-old Kevin, one of the participants in my study, explained he had spent part of his childhood moving from one place to the next between the South and the Southwest regions of the United States. During his travels, Kevin saw his family go from owning a home to renting apartments due to financial stress. Kevin said he enjoyed tinkering with the family desktop as one way to learn problem-solving; this experience played some role in his digital literacy learning. However, because the kinds of materials for literacy learning Kevin owned and interacted with depended on his family's mobility, how might his digital literacy be different if his family sold the desktop for extra money?

Critical race theory offers directions for studying the ways structural racism influences the construction of writing ecologies. A method for uncovering and challenging systemic racism, critical race theory, in part, shows that "anti-black attitudes and practices" (Ture and Hamilton 5) fuse into institutional policies and norms that create "inherited disadvantages" in "access to the goods, services, and opportunities of society" (Jones, "Confronting" 10). However, literacy studies has not sufficiently integrated critical race theory into its scholarship. Carmen Kynard observes that considerations of the social context of literacy has involved methods from other disciplines, such as anthropology and history. But literacy studies has not adopted a racial

perspective that explains how “deep political and ideological shifts” have left “structured inequalities and violence” in place (Kynard 6). In her review on critical race theory and literacy, Arlette Ingram Willis also notes that literacy studies has neglected adopting a race-conscious method of analysis to have “adequate discussions that address economic and social inequities, historically and in contemporary contexts that give rise to unequal access and opportunities for literacy learning: homelessness, immigrant and citizenship status, and poverty” (23).

Bringing ecological theories of writing and critical race theory together may be productive for Writing Studies scholars, as they direct our attention to how marginalized communities’ interaction with people and the materials of writing systems is also an interaction with the system of white supremacy. When literacy scholars study the writing ecologies of Black communities using a race-conscious lens for analysis, they can consider that the materials available to these, and other marginalized communities, are not mere givens but rather the products of well-thought out historical formulas of institutional oppression. For example, the abundance of white and male software developers in the tech industry reflects a historical project of distributing resources and power to white families that can begin as early as childhood (Margolis 53-7). In the coding for all movement, many afterschool programs rectify this unequal access and use of coding literacy. Black Girls Code and Maydm both introduce youth of color and girls to computer science in hopes of inspiring and cultivating their interest in seeing themselves as coders. Meanwhile, for young racially marginalized adults, computer code bootcamps such as Yes We Code and Hack the Hood offer direct access to the material and social resources of coding literacy and a possible career in software development.

Taking critical race theory and ecological writing ecologies together, I understand that adults in computer code bootcamps learn a new prestigious literacy practice in the midst of a complicated system of inherited disadvantages that shape how they learn computer

programming. I uncover in this chapter how the *learning practices* of participants are embedded in material and nonmaterial systems that they have devised for themselves. I analyze how the space, time, objects, and bodies participants name as part of their network, even if mentioned in passing, cohere as a *necessary response* to generational racial disparities that may impact how they learn coding literacy.

Context

Clearwater Academy is a career training program attached to a larger non-profit organization that offers a variety of services to the local community to end racism and economic disparities. A method for achieving these goals, Clearwater teaches low-income people of color and women web development (HTML, CSS, and JavaScript)ⁱ. For three and a half months, students attend classes four days a week, eight hours a day to learn how to design and code websites individually, in pair programming, and in teams. Two-thirds of the class teaches soft skills: students write and workshop cover letters and resumés, present elevator pitches, and practice mock job interviews. They also listen to invited speakers and have one-on-one conferences with industry mentors on a variety of job-, finance-, and tech-related topics. This extensive training may help these adults get paid internships or fulltime employment in the local tech industry.

Students at Clearwater Academy do not pay tuition, but they do have emotional and physical challenges that determine how well they learn programming. Learning coding literacy is a long process for students as they juggle many burdens in their lives, so Clearwater offers to help students: passing out free bus passes and gas cards, paying one month's rent, and referring students to social services that can help with other concerns. But, as study participant Patricia explained, Clearwater can drain its resources quickly, so "they don't have any funds to help you with rent, utilities, stuff like that." Clearwater is an incomplete sponsor for participants living

between the realities of their everyday lives and a possibly equitable future; *between* the systemic racial stratification that governs their lives and the opportunity to achieve social mobility using coding literacy. Because Clearwater has limited resources to help them in this “between space,” participants find other resources to stay in the code bootcamp while keeping life challenges associated with racial oppression out.

To understand life between learning and opportunity, I conducted a year-long ethnographic study at Clearwater that included participant observation and interviewing participants. I recruited twelve participants ranging in ages twenty-one to fifty-six between the Spring and Fall semesters of 2017. Six students identified as female and six identified as male. Seven of the ten participants self-identified as African American; two participants were biracial but described themselves having an affinity toward the African American identity. One participant explained that he was Afro-Latinx, but his life experiences, he noted, grounded him in the everyday lives of the Black diaspora. Unfortunately, at the halfway point of the Spring semester one participant left Clearwater due to poverty and childcare needs. Thus, I present data from eleven out of twelve study participants.

To return participants’ generosity, I offered to tutor all students—not just study participants—with job-related writing assignments. Richard, the coding instructor, and Janet, the soft skills instructor, welcomed my helping students write elevator pitches, cover letters, blog posts for the Clearwater Academy website, and offering feedback on students’ final presentations and portfolios. To learn programming, students completed exercises in HTML, CSS, and JavaScript on an interactive website. I myself completed several exercises on this website to get a sense of what the students experienced. I later helped students work through some of these coding challenges. In this way, I built rapport with participants and their classmates.

Methodology

Data Collection and Analysis

In order to understand the broader literacy contexts participants inhabited, I asked participants to draw a mapⁱⁱ of their support, the people and objects that helped them keep learning coding literacy despite racial disparities in their lives. Studies of literacies may concern themselves with the direct interaction with literacy practice and its context. In this chapter, however, a network of support asks that scholars stretch even further back before literacy practice happens to the multiple ways a person interacts with people, objects, thoughts, and emotions that jumpstart literate activity. In the context of a computer code bootcamp training low-income people of color, then, I asked participants to document and tell stories about the people, emotions, objects, and past experiences that circulate in their personal lives that helped them *approach* Clearwater as a literacy sponsor possessing the social (e.g. tech industry mentors) and physical (e.g. laptops) materials used for learning coding literacy.

Using maps to understand networks of support recalls previous studies that used drawing to help researchers understand literacy practices (Brooke and McIntosh 134–41; Prior and Shipka 182–6; Mason 96–102). For a computer code bootcamp that moves quickly through its curriculum, maps allowed me to slow things down to see clearly how events, objects, and people impact code bootcamp students' learning coding literacy. They also provided a response to the constraints of participant observation and interviews. Because both methods of data collection are confined to place and a time limit, depending on the responsibilities participants in my study had, details maybe limited or details maybe difficult to follow up on. With maps, participants and I could lengthen the time and space that seems in short supply during interviews and observation by writing a range of more concrete items on the page and discussing when and how they have operated throughout the semester. Participants can then demonstrate their agency over naming what matters to them in their sociomaterial conditions and describe how those items work, or

don't work, in their lives. In these interviews and maps, participants provided rich details on how these networks created useful resources for them.

Frequently used across disciplines such as sociology, communication, and economics, ego network analysis (ENA) understands “network” as a complex web of social interactions and social relations. ENA looks at an “individual’s social environment” rather than their specific attributions (e.g. race, class, gender) for “explanations, whether through influence processes (e.g., individuals adopting their friends’ occupational choices) or leveraging processes (e.g., an individual can get certain things done because of the connections she has to powerful others)” (Bargotti et al. 892). The range of interviewees named in their network are called “alters.”ⁱⁱⁱ Areas for analysis include the number of people in an individual's network, how those people relate to one other, and the strength of the relationship among the individual and people (See Figure 1) (Prell 8). These areas are often visualized quantitatively (e.g. charts, digraphs, or Bernoulli graphs) (Prell 9 –18; Marin and Wellman 21–22; Provan and Milward 4–12). In this way, researchers use the kinds of behaviors, knowledge, and resources that arise out of the networks to understand macro-level social patterns (Marin and Wellman 13).

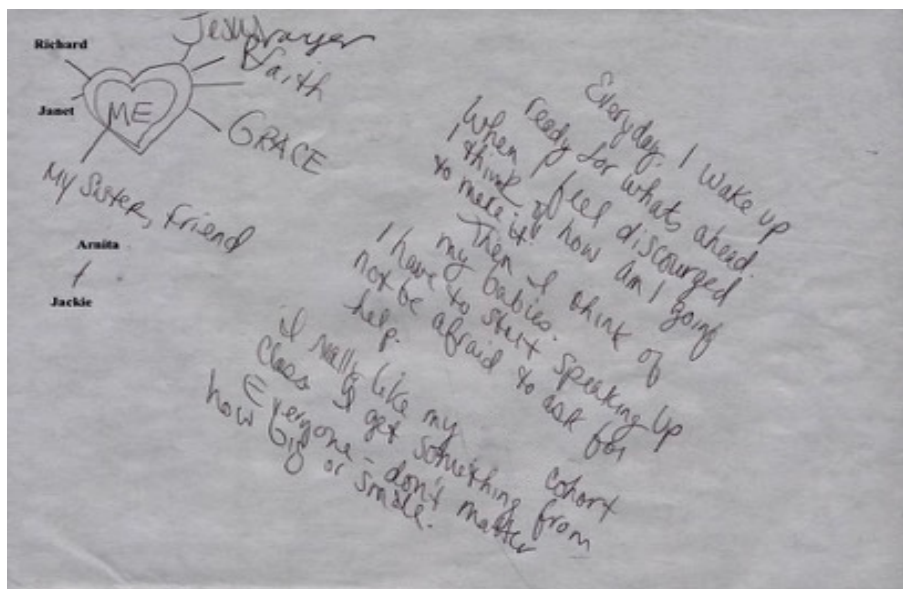


Figure 1. Patricia's heart-shaped network and reflection

ENA provides a systematic analytical approach that expands ways of studying writing systems. However, it often relies on participants naming *specific people*. This anthropocentrism is the antithesis of ecological writing, which decenters the writer from literacy practice and sees them participating in the circulation of material (e.g. desks, pencils, writing) and non-material (e.g. emotion) objects. ENA and ecological writing theory addresses one another's limitations: study the total sociomaterial experiences of literate subjects using a more careful and directed analysis. Taking note from ecologies writing theory, I extend "network" further to include the objects and places participants noted in their interview.

After drawing their maps, each participant explained how the person, object, or pet on the map helped them with learning coding. Follow-up questions during our conversations deepened the specificity of those answers. I conducted and recorded these thirty-five- to sixty-minute interviews on the Clearwater Academy campus during lunch or during extended class breaks, and one interview was conducted over video chat. Although this participant, Alice, drew her map during the video chat, I was unable to collect the map. However, I took copious notes during our conversation to capture her stories. In my first round of analysis, I did open coding, which involved creating a chart that listed each participant's set of named alters and then assigning each alter to a support type (cognitive, affective, financial, or physical). These types arose from how participants discussed each alter on their map. Frequently, participants named how and what kinds of resources the different alters provided, which helped avoid any ambiguity in categorizing the different support types. In moments when participants were not more direct, I inferred from the transcripts the potential resources the alters provided. I briefly summarized the interactions participants had with each alter, extracting sample quotes from the transcriptions as evidence. In a final category, I described in one or two sentences the resources that resulted from that interaction.

During the second round of analysis, I revisited the initial descriptions of the resources and the type of support they provided for accuracy and refining the language based on reexamination of the maps and interviews. Finally, in the third round of analysis, I conducted closed coding, examining how alters in each network related to one another based on the resources provided. This also involved counting the number of possible connections among them, which suggest the strength of participants' networks of support (Hanneman and Riddle 341- 44). Alters that provided similar resources and had the most influence on participants' learning programming and their relationships with others, according to the interviews, became the core theme of each network. I used the support types and resource descriptions to create names that summarized the most dominant kind of resource (see "Resources from Networks" in Table 1).

I resisted comparing each participants' themes to establish a theory, following ecological writing theory's argument that systems are so complex and unique that comparing systems would not help us accurately understand writing. I applied the same principle to this ecological view of literacy learning. Moreover, the comparison would flatten the unique experiences each study participant described on their maps and in their interviews. Although we see similar resources across participants in Table 1, the ways their networks operated had a life of their own. Thus, I describe three robust examples of the resources shared with other participants. Keisha, Kevin, and DeAndre's stories offer rich details, making them exemplars of the larger group in terms of resource representation. Keisha's network represents the emotional investment that Paul, Isaiah, and Alex gather from their networks. Similarly, Kevin's detailed explanation of how negotiating for time and space to practice computer coding speaks to the experiences of Alice and Rosie's ways of getting time to work on coding. Finally, DeAndre's use of marijuana and music typifies the ways networks help the body prepare for activating coding literacy

learning. As with Keisha and Kevin, DeAndre represents the experiences of Zelda, Patricia, and Zeus' work to deal with the physical demands of coding.

TABLE 1. PARTICIPANTS' NETWORKS OF SUPPORT AND RESOURCES

Participants	Number of Alters Identified	Number of Possible Relations	Resources from Networks
Zeus	8	9	Embodied Self-Discipline
Isaiah	5	8	Emotional Investment
Patricia	6	7	Strength in Body to Work
DeAndre	6	8	Body/Mind Preparation
Alice	7	8	Temporary Space and Time
Kevin	7	9	Temporary Space and Time
Keisha	11	13	Emotional Investment
Zelda	8	8	Body/Mind Preparation
Paul	7	11	Emotional Investment
Alex	7	10	Stability
Rosie	8	7	Secure Time

Findings and Discussion

Keisha: Emotional Investment

In her map (Figure 2), Keisha, a 34 year-old certified nursing assistant, showed that her most important alters for emotional investment were her daughter and siblings, Clearwater's technical skills and soft skills instructors, classmates, and, finally, Jesus. She wrote these alters on a circle, suggesting that Keisha passes through them for different kinds of emotional support before returning to Clearwater as a student prepared to learn programming. The outer ring names

the material conditions that assist in stabilizing the inner ring of the map. In other words, living in an affordable home and having a part-time job create a sturdy foundation for accessing emotional resources from her most important alters. She could not receive inspiration from her daughter, for example, if Keisha could not use government assistance and a salary from her reduced hours to protect her daughter's own well-being.

Keisha placed herself at the top ("Me") because she had control over many decisions in her own life before attending Clearwater. She "lived at work" as a certified nursing assistant, working from 6AM to 8PM—roughly one hundred forty-six hours every two weeks. Keisha worked this hard for two reasons: First, she had been homeless before, which she described as "the most ugliest things I saw." In fact, Keisha explained during our interview, that experience had given her "a fear of being broke" again. Second, Keisha used most of her income for her daughter, Althea, a lesson she had learned from her mother who also spent all her money on Keisha and her siblings during Christmas. Thus, Keisha relied on herself to stay afloat.

However, Keisha's decision to attend Clearwater Academy endangered her position as the central person of her network. She must attend class Monday through Thursday for eight hours a day. Because most of her weekdays were spent at the computer code bootcamp, Keisha had less time to work. Reducing her work hours impacted her ability to pay rent, utilities, and food. The consequences of this could be severe for her emotionally: while coding a tribute webpage to Althea in HTML and CSS, Keisha assessed her life and concluded that she "had never been this broke before" and began to cry. In addition, Keisha's family disagreed with her attending Clearwater Academy. They believed reducing her work hours to learn web development irresponsible of Keisha. How would she support herself and her daughter without a good paycheck? At the beginning of the semester, Keisha's boyfriend offered financial assistance to cover expenses; however, he later asked her to leave Clearwater, finding it a waste

of time and financially risky. When she refused, he stopped helping pay rent and refused to allow Keisha to use his credit cards anymore. Keisha's mother also believed she was irresponsible for trading work for a literacy that may never reap rewards. In fact, other family members *expected* Keisha to quit Clearwater Academy because they knew her to be "a starter but not a finisher." Financial difficulties on one side and a reticent family on the other, Keisha was always at risk of losing emotional control over her life. Nevertheless, emotional investment maintained Keisha's own motivation to learn coding despite the anxiety and stress of family and work, which I will now detail below.

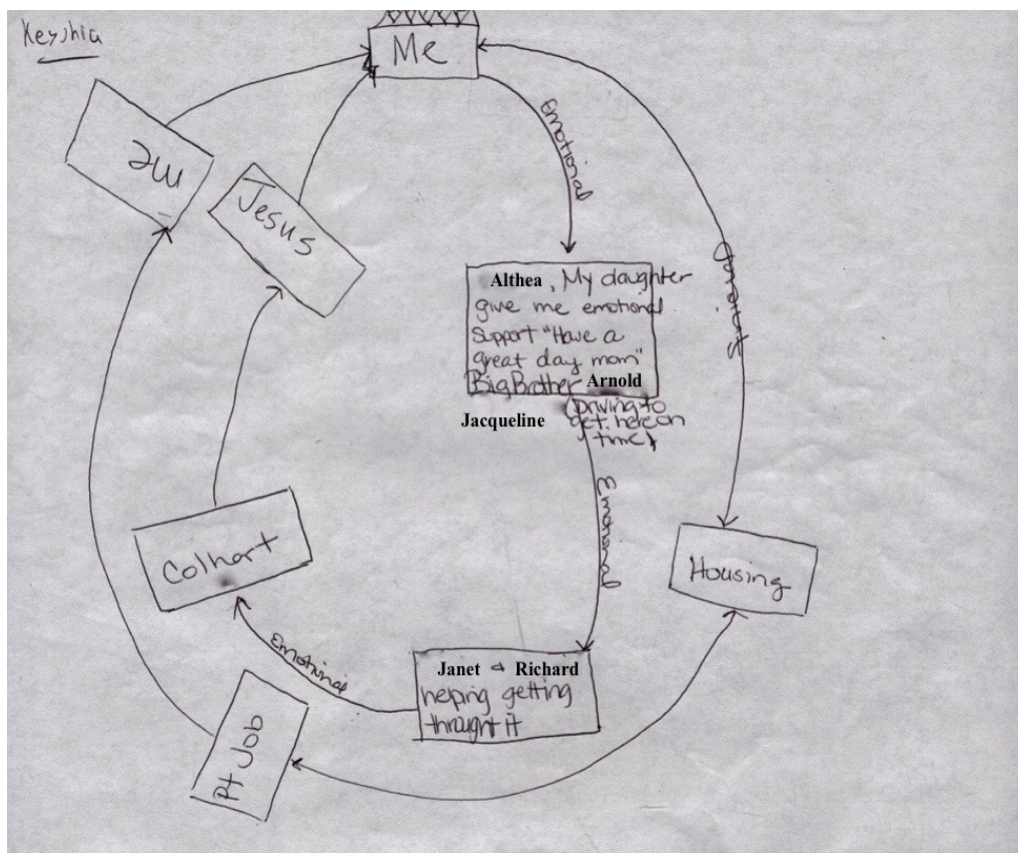


Figure 2. Keisha's network as two spirals.

To keep herself grounded in her pursuit of coding literacy, Keisha drew on different kinds of emotional labor. For example, Keisha admitted that she "puts on a front" in class—she

smiled and joked with her classmates, pretending that her life was okay. “And ninety percent of the time” she thought going to Clearwater was the right decision for her. However, “some days I be pissed off that I’m here. But a lot of days, I don’t. A lot of days I definitely don’t be pissed off that I’m here. Cause it gives me a purpose.” She further explained that not graduating was “when I would [feel] like all of this was for nothing. And I refuse to start over because then I get that look [from family].” Keisha demonstrated the emotional labor required to navigate life circumstances and stay committed to coding in a computer code bootcamp. She wasn’t so much as proving her family wrong to avoid more derisive comments on leaving yet another project unfinished; instead, Keisha seemed to *draw on* her family’s doubts and the potential negative consequences of failure to build motivation for learning coding literacy. This redirection provided one way for Keisha to maintain the control she had over her life since before attending Clearwater.

Keisha uses the other alters in her network to ground her emotional investment in learning coding. For example, her daughter Althea reminded Keisha of her Christian faith to supplement the emotional labor of learning code despite the temptation to return to work and fix financial problems. “She tells me every morning, ‘Have a great day.’ And she’s proud of me,” Keisha explains. Althea also plays the same gospel song every morning on a speaker. During our interview, Keisha could not remember the song and searched frantically for it on YouTube. This long search, however, demonstrates how much Keisha values her daughter’s presence each morning. And while her daughter wishes Keisha well every day, Keisha herself prays every morning, saying, “‘God, let me be great today.’ No matter what it is. Whatever I decide to do, even if I decide to go back to my job or Lord, please let me get a job offer, I’m bound to be great.” Althea reminded Keisha why learning coding matters: to please her daughter and to make life for Althea better. From her daughter’s well wishes and music, Keisha drew strength

and resolved to attend Clearwater every day. Keisha also built confidence that Clearwater was a good decision. After all, God had provided the computer coding bootcamp as a blessing, and despite the challenges while learning coding, God planned to provide Keisha with her every need, especially confidence in herself to do well in class. This faith in God strengthened Keisha's commitment to learning programming.

Despite drawing on her Christian faith and her daughter to help stabilize her emotional journey through Clearwater, Keisha had trouble staving off doubt: Richard, Clearwater's technical instructor, and Janet, the project coordinator, were key figures in the map that presented emotional investment as resources, as they worked to convince Keisha that she *was the great person* she had prayed to be throughout the semester. Three days into the bootcamp, Keisha explained during our interview, "I would've been gone. I know me. I wouldn't've stayed this long." In the first three weeks, Keisha wanted to return to 146-hour work weeks and a livable wage, the path out of coding and back to some stability. But Richard and Janet suggested that Keisha not downplay herself.

Making herself less important had always been Keisha's default attitude, even when she was growing up. Keisha's mother frequently went on trips with boyfriends, leaving Keisha to look after her younger brothers and sisters. As an adult, she still supported them, sacrificing her own needs to see that they succeeded in their own education and careers. What made Keisha stay despite these experiences and desires? For the first two weeks at Clearwater, Richard and Janet insisted that learning coding was worth her time and missing wages. She didn't believe them at first, but then the three-week review of her academic progress came. "[The review] said 'You are encouragable. You make people worth coming to this class.' So I stayed." Richard and Janet had acknowledged that she made a positive impact on her classmates, recognition that she rarely

received from family. In this way, authority figures in Keisha's life, unlike her mother and boyfriend, affirmed her existence and believed that she could indeed finish what she started.

Circulating around people like Althea and Clearwater instructors were institutional support. Government assistance brought some stability to Keisha's finances, which would help Keisha recover the independence she once had before going to Clearwater. Keisha paid \$600 in rent each month, but because Clearwater Academy required so much of her time during the week, Keisha could only work nine hours a month, making about \$400. She later received a 5-day notice for not paying full rent. But "by the grace of God [the landlord] took the letter that Janet had wrote for my acceptance letter" and got a section 8 voucher from the federal government that "dropped my rent down to zero." Keisha still paid for food and other utilities; nevertheless, that government assistance further protected her will to learn coding. Although Keisha placed "Housing" underneath financial support in her map, having the voucher offered her *emotional relief* from the burden of not having enough money to pay rent. Like the three-week review from Richard and Janet, the voucher gave Keisha less of a reason to delay or end her attempt at learning coding literacy.

Keisha named other alters in her network, like her sister who drove her to Clearwater and her older brother who encouraged her to keep going to class. But most significant are the ways resources from other alters allow Keisha to claim greater emotional investment in learning programming and reclaim the agency she's used to possessing. The combination of faith in God's provision, the presence of her daughter, the encouraging observations about her personality and contribution to the emotional well-being of her classmates, even doubts that she wouldn't finish, led to securing the motivation to learn programming.

Keisha's experience shows that learning coding at a bootcamp involves a great deal of emotional labor. Frequently, the emotions of African Americans, especially Black women, can

be misinterpreted, disrespected, oversimplified, or ignored (Richardson; Carey), as if Black women do not have a right to complex feelings in a world of white supremacy. A curriculum that centers coding literacy as a skill rather than the holistic well-being of its students runs the risk of perpetuating disrespect of Black bodies. However, as seen above, a coding program that tends to the affect of its adult students may help support student learning and provide a pathway to staying on track with coding literacy learning. The work of instructors and other support staff unknowingly operate alongside a host of other alters within students' lives. Further awareness of these other pieces may allow for deeper, more meaningful collaboration with other alters in the effort to balance the demands of a computer code bootcamps with the burdens of its adult students. In this formulation, respect and acknowledgement of African Americans' agency over their bodies and emotion center the work of computer code bootcamps, making them spaces of both liberation of and protection for Blackness.

Kevin: Temporary Time and Space

Twenty-nine-year-old Kevin's map (Figure 3) shows that his support produced two resources: emotional well-being, from family, friends, and his dogs in the Southwest where Kevin was from; and accumulating temporary time and space, from housemates and a non-profit he volunteers for. Emotional well-being is a long-distance resource as it traveled across the country to Kevin through telecommunication with family, his fond memories of his dogs, and his own goals to return home and find a good job after graduating from Clearwater. However, his alters in the Midwest offered the most help to Kevin's accessing ways of learning coding literacy. While Kevin wrote on his map that he received "Assistance, encouragement, and friendship" from his classmates and "work around school and homework" from his housemates, the alters actually worked together to "bend reality" and offer him more time and space. Gathering extra time and temporary space helped Kevin find respite from life responsibilities

that may have prevented him from doing well in Clearwater Academy. I explore this point in-depth in the following section.

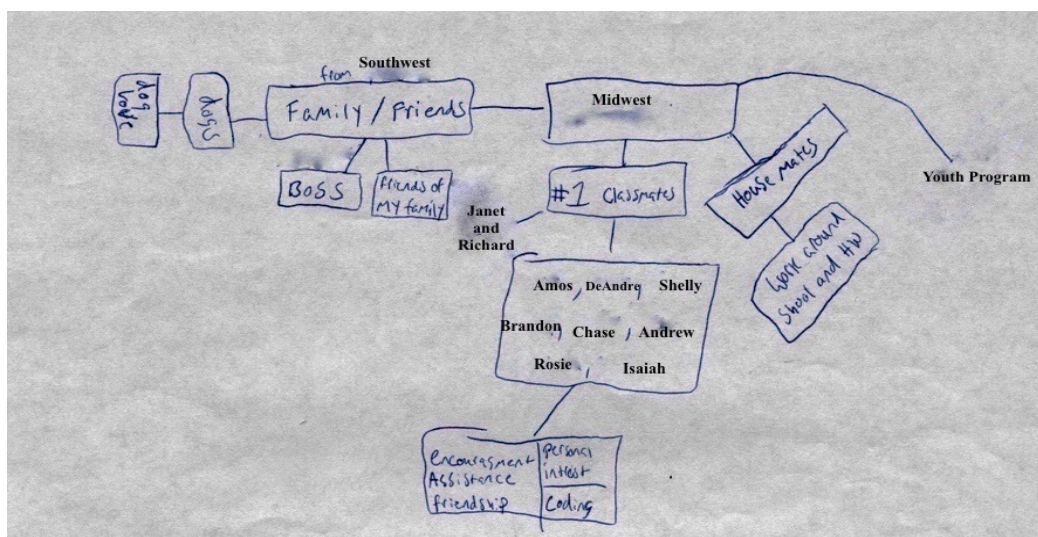


Figure 3. Kevin's support from the Southwest and the Midwest

To understand the value of getting more time and space for coding literacy, it's important to first envision the circumstances under which Kevin learns programming. When he arrived in the Midwest, Kevin moved into cooperative housing where he and his housemates shared chores, food, and bills. Because co-ops serve low-income residents, Kevin's expenses totaled to at most 40 dollars a month. Nevertheless, Kevin worked as a deli chef and as a bouncer to get a little extra income. On top of these two jobs, he fitted into his schedule volunteering at a non-profit organization where he pursued his devotion to helping marginalized youth in school. Kevin attended Clearwater 32 hours a week, but classwork tended to follow him home, taking up time he could use for work, family, friends, and self-care. Thus, coding became the center of Kevin's life and other responsibilities circulated around coding. Despite these challenges, Kevin gave his classmates and instructors the impression that learning coding came easy. They didn't realize that understanding coding wasn't a snap for Kevin. Much of what he learned from them at Clearwater Academy had to be later refined elsewhere. During our interview, Kevin offered his

thoughts to assumptions he learned coding easily, saying, “Dude, no I don’t [learn web development quick]. Everything we did in class is what I do when I go home . . . I work on this shit. And I do it the whole time . . . And I’m doing this until 11 and then I go to bed.”

At home, Kevin *negotiated* for time and space to make his success at coding possible. For example, asking his housemates to “[let] me work around my school schedule and school work, with household chores, so long as I actually do stuff” was the best way to work many hours at home. Kevin called his housemates his “helpmeet,” partly because they often asked if they could do anything for him and they even sometimes connected Kevin to potential clients for freelance web design jobs. When Kevin had too many assignments to do but it was also his turn to clean the bathroom, or do some other chore, the housemates volunteered to do his chores for him. “They be like, ‘This time I’ll do it and you’ll just take mine next week, or something, *when you have more time*’ . . . Obviously, I have to do it, but I don’t have to do it at that second.”

His housemates, then, are not just a helpmeet, but partners in his coding work. They sacrifice time they could use for their own responsibilities to give more time to Kevin. He could then keep up with Clearwater Academy’s work and also execute the lessons he learned from his classmates. With work at the grocery store and the downtown bar encroaching even further on Kevin’s ability to commit to deep, sustained practice on coding, his housemates are even more essential for time accumulation. Kevin quite literally soaks up his housemate’s time for himself, so he had a different relationship with the evening hours after returning from Clearwater Academy.

Overlapping with negotiating for time was Kevin renting spaces from his network to do work in comfort. For example, he called his bedroom “a little Japanese room” with a bed, a night stand, a hamper for clothes, but “I literally have one walkway. Two people in there would be way too much.” Kevin was tall and muscular. Because his room had space for one person, trying

to work on coding over long stretches of time would be uncomfortable. He needed isolation to focus, so the house's living room or kitchen would not help either. Recognizing the discomfort of Kevin's small bedroom and the need for isolation, his housemates offered him their bedrooms as workspaces. "They're just like, 'Hey, I'm going to be gone for the weekend. If you need to use my room, you can. Like it's cool,'" Kevin noted. The housemate would then put clean sheets and pillow cases on their bed. For the rest of the weekend, Kevin sat at his housemates' desk and worked on coding and writing assignments, stopping only for "bathroom breaks and food." When the housemate returned at the end of the weekend, Kevin would "just take off and leave their bed fresh [remove the dirty sheets], and they put on the new sheets when they get back."

Contracts for renting spaces continued with the non-profit organization that Kevin volunteered for throughout the week and sometimes on weekends. The non-profit offered programs and services for low-income LGBTQ youth, and Kevin tutored youth in algebra, but the organization was interested in his experiences at Clearwater Academy, too. "So, they're also kinda like the housemates," explained Kevin during the interview. "They're always and constantly asking me how I'm doing in class." The non-profit also offered him a room to study in. Typically, Kevin would spend two hours tutoring, but his working as a bouncer and as a deli chef squeezed his available time for coding. In response, the non-profit worked around his schedule, switching his two hours of tutoring with two hours of coding instead. Eventually coding in became a fulltime job for him: he went from tutor to student at the non-profit. In his interview, Kevin clarified further: "With the amount of work that I've had recently, they've had someone fill my void. Fill the fact that I'm not there [to tutor]."

Utilizing this time and space related back to Kevin learning from classmates and instructors. Classmates like DeAndre and Addie shared common interests in music and anime, so Kevin bonded with them throughout the semester. Most important, however, was they're

assisting Kevin with difficult coding exercises from the interactive website. Alongside his classmates, Kevin explains, the technical instructor Richard plays a significant role in helping him figure out if he should be a web designer and improving Kevin's soft skills, like designing his resume and pitching ideas to investors. Thanks to them "I'm definitely better at talking about myself in a professional sense. I'm not good at it, but I'm getting there . . . Coming to this class, with their assistance, and my classmates, too." Thus, the work established in class wasn't lost outside Clearwater. As explained before, the work of the computer code bootcamp followed Kevin to be used productively in the time and space borrowed from others.

Kevin's stories about his network of support bring into sharper focus the role time and space may play in learning programming from a computer code bootcamp. Not only time and space in of themselves, but what happens within that time and space. In this context, time and space are hot commodity Kevin alone isn't privileged enough to possess. Despite his limited sociomaterial conditions, Kevin pulls from a network of people and institutions for additional time and space. The non-profit provided two hours of tutoring so Kevin can work on code; his housemates could do other tasks with their time, but they, in a word, rent out those resources to Kevin by switching household chores. Both the non-profit and housemates rent out their classrooms and bedrooms to Kevin. Most telling, however, is how adding these resources to his own schedule also allows Kevin to step away from life responsibilities briefly; he doesn't have to attend to money or relationships, because others take on that work for him. This siphoning of time and space, and respite from other responsibilities, help him keep pace with the demands of learning coding literacy from Clearwater.

DeAndre: Body and Mind Preparation

DeAndre's map (Figure 4) emphasized the relationship between people and different kinds of external substances. The 21-year-old wrote that "Lots and lots of [weed]^{iv}" linked to his

friends while “Music” hovered nearby. Yet the other alters occupied a space of their own: two classmates at the bottom,” YouTube” on the right-hand side, “Best Friend” near the top seem to “free float,” suggesting they independently offer resources but are not explicitly related to one another in the same way as friends, weed, and music. These other alters, then, appeared to supplement the heavy work that weed, friends, and music do for DeAndre as he learned coding literacy. This section unpacks how these three pieces shape DeAndre’s behavior and thinking to help him prepare his body and mind for the demands of coding.



Figure 4. DeAndre's "Free Floating" Support

As he learned coding literacy, DeAndre grappled with racist readings of his body based on the color of his skin and wardrobe. He wore hoodies, sweats, and sneakers, and long

dreadlocks dangled from underneath a black baseball cap. Having dropped out of high school and feeling stuck in life, DeAndre decided to get his GED. The test proctor looked at DeAndre and assumed that he couldn't pass the required math exam. But DeAndre later surprised the proctor after he answered nineteen of the twenty test questions correctly. DeAndre ran into similar issues when he interviewed for a spot in Clearwater Academy. Richard, the technical instructor, assumed that DeAndre wouldn't survive Clearwater's rigorous, fast-paced curriculum "based on my looks, everything."

Being misread occurred outside of school, too. In Chicago, DeAndre's default expression was "mean-mugging." This posture was for defense against anyone thinking of jumping him. However, living in the majority-white communities around Clearwater, DeAndre noticed that people seemed embarrassed by his facial expression, as if he were "crazy." This taught him that he "ain't got nothing to worry about up here." Feeling no sense of danger, unlike in Chicago, DeAndre chose to have a nice, chill demeanor. DeAndre was aware of how race and place shaped readings of his body and so adapted his behavior to match the expectations of that atmosphere. Weed seemed to play a significant role in DeAndre's coding literacy learning and maintaining this chill demeanor.

To better understand DeAndre's testimony on how weed contributes to his learning and behavior, I want to place his experience in the context of current marijuana policy and academic research on marijuana's influence on the body. Currently, there's disconnection among established scientific knowledge, lived experiences of marijuana users, and the policies and laws that disrupt literacy education and oppress racially marginalized people. Studies on cognitive performance and marijuana in general are inadequate because the U.S. Drug Enforcement Administration (DEA) classify marijuana as a Schedule I drug. According to the DEA, marijuana "has no accepted medical use and a high potential for abuse" (Drug Enforcement

Administration). This classification significantly limits researchers' access to marijuana to study its effects. Any research on marijuana must first be approved by the DEA, as stipulated in the Controlled Substances Act. So-called concerns of marijuana abuse have been since the 1970s to enact racist methods of persecuting African American communities in disproportionate numbers compared to Whites (Lynch 186; Alexander 59; Beckett and Herbert 68). Nevertheless, marijuana has been approved to treat certain conditions in 32 states (Lopez), and there's an abundance of anecdotal evidence and, more important, recent studies that suggest marijuana can cause a variation in creativity and cognitive learning (Schafer et al. 298). Meanwhile, there is a long tradition of computer programmers using marijuana, and other substances like LSD, to address the strenuous work of coding at a computer or to (arguably) jumpstart creativity and rationality (Veshkin; Markoff 65–68). Given this context, DeAndre's story below represents a common experience for millions of literacy learners; however, his story also underscores the legal danger marijuana use poses to his own freedom and efforts to learn coding for social mobility.

As weed was an important component in shaping his behavior and learning, DeAndre relied on himself and friends for a continuous supply. "If I don't have no weed, my roommate has some weed," observed DeAndre. "If he don't [have weed], Charles will have some. And he don't, Marcus will have some, and if he don't, I will have some . . ." But they also participated in helping DeAndre work through his emotions whenever he was upset. Familiar with his swinging moods, when his friends would noticed something bothered him, even without DeAndre speaking about his problem, they would give him a blunt and sit and relax together in silence. Friends and a supply of weed served as the foundation DeAndre's experience in Clearwater.

First, weed helped DeAndre find a calm and collected attitude in class. DeAndre noted that he didn't use weed to "deal with stress" as stress was a life-long, daily experience that he

could deal with on his own; instead, he wanted to just be “chill” at Clearwater. Otherwise, DeAndre admitted, he could get angry easily, especially when classmates didn’t catch on to coding lessons as quickly as he had expected. Thus, weed helped him keep a calm, anxiety-free state of mind that allowed him to maintain healthy relationships with his instructors and classmates, such as T-Dub and F-Dougie. These two classmates had been key partners to his learning coding literacy, especially later in the semester when Richard withdrew his helping. “He told me, ‘We are at a point where I’m not going to help you no more . . . you smart enough to get stuff done on your own. Use your resources,’” DeAndre recalled. Both were willing to give DeAndre clues on how to figure out coding problems or an issue in app or web development. To DeAndre, T-Dub and F-Dougie were good resources to tap into but it is weed that helps mediate DeAndre reaching out to those students without incident.

In addition to mediating his emotions and behavior, weed assisted DeAndre in learning coding literacy itself. The drug, he says, “stimulates everything I wanted to do. I just do [tasks] faster. So if I wanted to focus on something I just smoke some weed . . . I can basically direct my effect to where I want it to go . . . to make it into a cerebral high.” DeAndre seems to even have agency over the chemical influences on his brain, that he can direct the high to keep him focused. Nevertheless, there’s a limitation to smoking weed. The high could be too much for him to handle in one sitting. One day while working on his resume, DeAndre was “high as hell.” Being too distracted from the high, he “went from one thing to another in less than 10 minutes because I kept thinking of new things. I kept switching [the resume], asking people what to do.” The distractions caused him to get nothing done in class. In light of these negative effects, he must keep in check how much weed goes into his body. Despite this danger, DeAndre stuck to weed because, if anything else, he liked not being too “turnt up” in class.

Music shared a similar cognitive and behavioral process of staying in step with Clearwater Academy, such as managing his attention deficit disorder. DeAndre found that he could feel one side of the brain activate when completing a task. He used playing video games as an example of this awareness: he feels activity happening on the right-hand side of his head, just above the temple. There's activity in the back of DeAndre's brain but "it's still distancing but that [the game] takes both parts of my brain. It's still a lot of [mental] work." During his time coding projects, DeAndre found himself needing to distract one thought while the other focused on actual work. One part of the brain dealt with the "numbers and letters," he explained, while the other part "has nothing [productive] to do."

For example, the exercises on the interactive website offered many options for how DeAndre might design the look and feel of his websites and mobile apps, but he also needed to turn in a web portfolio for class. These two tasks inevitably began to inform one another: "I keep thinking about certain things I got to do and certain shit I need to be coding. Like, 'Oh I could be adding this and such and such. Or I add this and this.'" The side of the brain that had nothing to do was, according to DeAndre, "the ADD side. That's the side that says, 'Yeah, let's do all that shit at one time.'" DeAndre wanted to use the ideas he had learned from the interactive website for the portfolio assignment, so he got sidetracked, tried multitasking, but got nothing accomplished. DeAndre was aware of this competition between himself and his ADD. To avoid conflict among tasks, he played music in his headphones to keep the ADD side of his mind occupied while the rest of the mind concentrates on coding. A variety of songs passed through his ears each day, from beats that he "just found fucking dope" to A\$AP Rocky to Adele. With music playing in his ears, DeAndre "sings along and do the code camp. It distracts me. And it keeps my brain working on that while my right side is working on this. While my left side is focusing on quotes and lyrics. *Occupied.*"

DeAndre created a mental model for how his ADD works. That awareness helped him hash out an effective method to not let it derail him from practicing coding. But this small task resonated throughout the network DeAndre established for himself. In other words, we can see how different alters in DeAndre's network impacts his relationships with those alters. His investment in objects like weed and music suggests that he staves off the consequences of bad habits: falling asleep in class, returning to a life of committing cons, and getting sidetracked not only from tabs on his browser but also his own thoughts. But this disciplining of the body and mind also helps with relationship-building. DeAndre might fall behind in his work and appear uninvolved in the class, so he learns to discipline himself based on the behaviors of his peers or advice from Janet and Richard. Finally, weed and music keep DeAndre professional-minded and professional-acting in Clearwater Academy. But this strategy has troubling implications: while learning to mentally focus may not indicate a racial ideology, the behavior of his body implies that he must balance acting out his authentic racial identity with attitudes of discipline frequently associated with whiteness.

Responsibilities related to racial oppression make coding as a material and embodied practice more visible for study participants. Keisha living on the border of homelessness, Kevin in search for time and space, and DeAndre's racial body implicate racial oppression as a risk to maintaining access to prestigious literacies. Programming looks like a cognitive process, but that dimension of learning may be a privilege for some who live in stable sociomaterial conditions. Participants call on and arrange additional alters to form their networks that help keep volatile conditions in check, within and without Clearwater Academy. Even these networks and movements along such networks might be a privilege, as Clearwater students from previous cohorts have been overwhelmed by physical and mental disabilities, childcare responsibilities, and domestic violence, forcing them to quit Clearwater and quit coding.

Conclusion and Further Research

Using ego network analysis, this study investigates the processes of how low-income African American adults access coding literacy as they manage the social and emotional consequences of white supremacy that may block that access. Clearwater Academy, the computer coding bootcamp that trains women and people of color in web development and job skills, provides ample resources for the cognitive process of learning coding literacy but doesn't have the same breadth of resources to circumvent the bodily, emotional, and financial responsibilities and challenges students must endure. For this reason, adults draw on an ad hoc cluster of support to design and patch up their vulnerable sociomaterial and financial circumstances. From these clusters, they gather resources that help them access coding literacy as a resistant response to inequality in their lives. Some resources from these networks of support include emotional agency, temporary time and space, and disciplining the mind and body. These resources may help create stability in the lives of participants across different contexts so that they can tap into an uncommon and infrastructural literacy like programming. In describing these experiences, the study contributes to understanding racially marginalized communities' strategies for accessing new emerging technologies. It shows how a complex network helps adults approach these technologies when made available, demonstrating the kinds of knowledge they possess to make access possible in light of racial inequality's barriers.

These findings encourage further research in other areas. First, as literacy scholars continue to study writing ecologies and the materiality of literacy, they may bring to bear the theoretical lens of critical race theory. Combining these ways of seeing literate practice help scholars understand how current racial formations (e.g. color-blind racism) intertwine with economic and political policy and influence literacy learning and the circulation of literacy materials. Second, this study captures networks of support in a specific moment in participants'

lives; there's opportunity for studying literacy practice and learning over a lifespan (Prior, "Forum," 217; Bazerman et al. 354 – 57). A long-view perspective of writing systems may help ecological writing theory take into account how writing ecologies change. By applying these theories to learning digital literacy, ego network analysis might be productive for Writing Studies scholars to reveal other kinds of relationships between material and nonmaterial in a system of writing and what flows among those alters to support digital literacy learning. As composing and learning become more networked but vulnerable to historical and ongoing forms of identity-based oppression, Writing Studies can take up new tools to study these happenings and possibly influence the policies and attitudes that determine the social consequences of these digital literacy practices.

Notes

ⁱ HTML and CSS structure and visually style webpages while JavaScript automates tasks, such as algebraic calculations. Software developers consider JavaScript a proper programming language over HTML/CSS. Coding literacy, I argue, does not make such distinctions; coding literacy encompasses the broad understanding and act of writing out a set of procedures, in any kind of programming language, for a computer to follow.

ⁱⁱ All maps in this chapter have been de-identified and anonymized.

ⁱⁱⁱ Alters seem to share characteristics with literacy sponsors—"people or institutions that can help or hinder literacy for their own advantage" (Brandt 167). However, in ego network analysis, alters are the items named in the maps themselves. Another distinguishing feature of alters is that the scope of sponsorship widens to include locations objects.

^{iv} Writing Studies has yet to adequately theorize the effects of drugs on writing bodies. Although a necessary area of research, this topic does not fit within the scope of this study.

Chapter 4

Making the Pivot with Coding Literacy

The *2018 State of Black America's* first-ever quantitative report on how digital technology create equally shared “business and educational opportunities” among racial groups show that across the majority of companies in the digital workforce fewer than five percent were African American while half the workforce were White (National Urban League 3, 5). In addition to hiring practices, social and structural barriers to consistent access to exploring digital technology such as computer programming in part lead to the underrepresentation of African Americans in tech (Wang and Moghadam 617).

A nationwide movement to establish more computer science curricula in public schools for racially marginalized people has attempted to breakdown such barriers and encourage their studying computer science. As part of this movement, computer code bootcamps offer adults a chance to learn programming when they had no such opportunity earlier in life. Their accelerated training in web and/or app development may lead to (paid) internships or full-time work as software developers. Some computer code bootcamps train marginalized adults in hopes to help diversify the tech industry and possibly increase chances of their upward mobility (International Telecommunication Union 4). The realities of achieving this goal seem modest: Course Report shows that approximately 22,949 students graduated from code bootcamps in North America in 2017; one percent of graduates were African Americans with an increased earning salary to \$60,000 (Eggleston 2017 and 2014), up from \$43,300 the average household income for African Americans (Pew Research Center).

But *how* do these African American graduates leverage coding literacy into economic and social advancement? This question matters because the tech industry has consisted of a labor force and a culture that seems to center Discourses of whiteness, or beliefs, actions, and ways of

communicating that reflects White middle class culture and ways of seeing the world. As I explain further in the theoretical framework section below, these white racialized spaces present a range of barriers that can further marginalize people of color as they attempt to participate in creating and shaping our digital cultures.

Explorations of race, computer code bootcamp education, and coding literacy in the workplace seem limited in computing science education and Writing Studies scholarship. In “the first [report] to explore the experiences and perspectives of bootcamp students” (Thayer and Ko 252), Kyle Thayer and Andrew Ko interviewed 26 code bootcamp students from different racial backgrounds and showed that graduates from code bootcamps faced personal and financial barriers to their success (246 – 252). However, the study does not focus on the experiences of African American participants, their internship experience, and what happens after leaving tech. Meanwhile, Writing Studies acknowledges that coding is a type of writing that supports and augments the capabilities of writing itself (Vee 27 - 42), and much scholarship investigates the rhetorical and cultural activities of software (Brown 5 - 6; Brock; Brown and Vee; Marino 473 - 474). Thus, there is a dearth of qualitative research in literacy studies on the critical role code bootcamps play in the literate lives of African American adults.

To bring more attention to Whiteness and coding literacy in the field, this chapter describes a small-scale qualitative study on how African American adults from Clearwater Academy, a non-profit code bootcamp in the Midwest, use coding to further their social position post-graduation. In the Spring and Fall semesters of 2017, I visited Clearwater weekly where I conducted participant observation and one-on-one and group interviews with a total of twelve African American students, ranging from ages 21 to 56. This chapter draws on one-on-one interviews with eight participants three to six months after their graduation from Clearwater. I also interviewed one graduate from the 2015 class who had become a highly successful full stack

developer. Open coding and closed coding of the transcripts drew out key themes and a working theory that I later describe.

Based on data analysis of these interviews, I argue that access to the opportunities of coding literacy require performing Discourses of whiteness within the workplace. However, coding literacy, and knowledge associated with it, can be re-purposed and attached to other literacies to address other needs, redirecting coding literacy practice away from serving the needs of the workplace and toward personal and, sometimes, professional well-being. This argument calls us to consider how coding literacy may belong to a larger digital literacy ecosystem and is not necessarily a dominant method of communication in the early phase of its democratization. When stakeholders elevate coding to a literacy, they adopt a convincing argument that coding can produce a host of societal benefits: digital technologies and their features, developed rapidly through programming, seem to “innovate” or “disrupt” existing paradigms in health, education, food culture, and political power, for example, what scholars in Writing Studies call the literacy myth. Responding to these expectations, this study digs deeper into the realities of computer code bootcamp students who sit on the frontlines of a labor market that demands more digital literacy. The dynamic relationship between the new economy and institutional racism offers a productive site to see how African Americans step into or are pushed out of the structures of opportunity that computer programming evidently affords.

Theoretical Framework

Literacy as white property can help us make sense of how racial identity’s entanglement with computer programming impacts the lives of study participants. According to legal scholarship, white identity has property value. Thus, the law confers and protects social and economic rights to whites (Harris 1714). Literacy as white property extends this observation and postulates that literacy is a key feature for distinguishing White identity from racially

marginalized people. Law and policy on educational civil rights preserve a system in which whites acquire and use their literacy as a raw economic resource to shore up privileges in education and work. Meanwhile, these same laws and policies deny racially marginalized people access to quality literacy and the opportunities it may afford (Brandt 5 –7, 203; Prendergast 7, 16 - 21; Ladson-Billings 5 – 6), subscribing a power imbalance among racial groups.

Literacy as white property offers a rubric for understanding how digital literacy, and coding literacy especially, can become an additional trait of white identity. Modern white ownership of technology recalls centuries of Western dominance over tools used to conquer and control non-Western civilizations in pursuit of progress for White society. That progress includes appropriating the technological practices or obscuring the technological contributions of Black and Brown communities (Dinerstein 572; Sinclair 4 - 5). The ideology that Western culture is more intelligent and ingenious than non-Western and Indigenous cultures governs our technological systems, what Jentery Sayers, citing Simone Brown, calls “protoypical whiteness.” Sayers goes on to write, “Across technical and cultural materials—through machines, actions, and ideologies—protoypical whiteness is complicit with structural racism. It profoundly shapes interpretation, and it operates as the basis upon which most technologies are sourced and maintained” (6 – 7). Narratives about the innate intellectual ability of people of color versus Whites fuels unconscious bias and can turn “small behaviors” into “macroscopic patterns of discrimination, marginalization, and underrepresentation” (Lewis, Shah, and Falkner 487). The overrepresentation of Whites in the software profession, then, is an inheritance of historical ideology and rich narratives that is further masked by the industry’s assessment of its coders using so-called color-blind meritocracy (Daniels 1383, 2015). As with other technologies, computer technology has evolved significantly in the last three decades, yet it has, in general,

benefitted White people rather than assisted in the social advancement of the racially marginalized (Daniels 591, 2013).

These theories inform my central argument that the consequences of whiteness circulating around and within structures of opportunity for coding diminishes the promises of coding literacy in participants' lives. While the problem of whiteness can be widespread in other majority white spaces, participant's experiences are more troubling as coding literacy has become valued for providing a pathway toward social mobility and social advancement. The high regard for coding, despite its unclear consequences for marginalized people, makes studying the journey from code bootcamp to internship a pressing matter for investigation.

Relevant Literature

The intersections of literacy and race as performance help further illuminate participants' encounters with coding practice based in whiteness. Given the expansion of opportunities to learning coding, computing science education currently recognizes the need to design educational tools for more diverse students learn programming (Cooper et al. 1 – 2; Kross and Guo; Krishnamurthi and Fisler 3). However, literacy learning also involves knowing when and how to use speaking and writing within the values and behaviors of a given community, called Discourse (Gee, "Literacy," 6 - 7). One cannot "teach anyone to write or read outside of any Discourse . . . you are always teaching more than writing or reading" (Gee, "Literacy," 11). Learning the literacy practices and social values from a social institution beyond one's own home literacy will likely confer on that student a set of social goods and privileges. In other words, literacy learning is a process of ideological acculturation and being, (Gee, "What is Literacy?", 19) one that's possible for racially marginalized people to achieve with the correct coaching from a member of the Discourse they seek to join (Delpit 298 - 299). Thus, the process of teaching and doing coding literacy is also a process of apprenticeship in the social practices of

coders (Coleman 496; Gregg 187, 193). Those who complete this apprenticeship—whether in three months or in four years—stand to possibly reap great rewards for performing according to coding Discourse: status, power, and money.

However, Discourses also use literacy and social values as gatekeeping devices to control the distribution of power and privilege. Reconciling African American culture, language, and knowledge with social institutions that value middle class Whiteness has been and continues to be a difficult and dehumanizing effort. Both W.E.B. Du Bois and Franz Fanon encapsulate this negotiation when they take up different strains of the dual identities African Americans must contend with. They are stuck between being Black and being American, hoping somehow to reconcile the two without losing access to the opportunities constitutionally promised to them (Du Bois 38 – 39). But just as pressing is the pathological desire to cast off their cultural backgrounds and wear white identity to survive (Fanon 119). Both double consciousness and white mask pathology persist for Black people today (Smitherman 10 – 11; Edwards et al. 55).

African Americans must perform Whiteness to acquire academic and professional achievement, a performance that draws on interlocking relationships among language, bodies, and an awareness of what Ersula Ore calls “white racialized space” (257). These are often public spheres rhetorically designated for Whites Only. A Black person crossing into this space risks “being delegitimized, satirized, warned to stay in their place, and, finally, when the threat of transgressing the colorline is at its greatest, lynched” (259). Vorris Nunley draws similar observations, writing that African Americans must ward off “Black social death” in public spaces (24). To avoid these consequences, they must have “an acute understanding of racial hierarchy, racialized codes of conduct, and creative methods for usurping these rhetorical constraints to both safely navigate as well as demonstrate that qualities conventionally associate with whiteness . . . are . . . commonsense by design” (Ore 258). African Americans use a variety

of rhetorical strategies with their speech and bodies that demonstrate Whiteness to survive these spaces. One most well-known strategy is code-switching, using “standard” English in educational and professional settings, as that will supposedly lead to economic success (Delpit, *Other People’s*, 68) but at the expense of undervaluing other kinds of Englishes (Young, “The Costs,” 67).

A key site for opportunities with coding literacy, the software profession is implicated in cultures of whiteness and white racial performance for African Americans. According to the U.S. Equal Opportunity and Employment Office 68.5% whites participated in nationwide tech companies while just 7.4% African Americans are represented (2). This suggests that software development is a white racialized space. Systematic documentation of actual experiences in tech further solidifies this notion. For example, a 2017 study found that 37% of participants across ethnic groups left their tech job due to unfairness or mistreatment. As one Black engineer explained in the study, “My employer assumed I knew Ebonics because I was Black. She also assumed I would be okay with coded language around affirmative action and implying that people of color get in to schools more easily than white people” (Scott, Klien, and Onovakpuri 10). In a separate report on African American experiences in Silicon Valley, Howard University interns for Google found they did not share many of the cultural values and practices of their white and Asian co-workers, feeling isolated from their co-workers (Vara).

The legacies of racial desegregation re-appear in cultures of the tech industry. Given that white supremacy cuts across all institutions and communities, it is no surprise that the tech industry—and software development more specifically—have inherited whiteness. However, I find space to extend these observations further through the sociocultural lens of literacy. In this perspective, scholars understand that literacy as the *use* of reading and writing in particular contexts by different communities. In addition, communities display their values and traditions

about their practices that play into establishing who does and does not have power in the community. Here I explore the consequences of failing the white litmus test and the implications these consequences have for emerging digital literacies and race. In this chapter, I show how whiteness limits the participant's coding practices and access to opportunities often associated with programming in the workplace. Then I highlight how a few participants found new pathways of success in activating multiple literacies alongside coding literacy outside the workplace.

Context of the Study

Clearwater Academy is based in a Midwestern city that has one of the fastest growing tech sectors in the United States; the number of startup tech companies founded in the city and the number of technology specialists employed overall increased significantly within one year. This growth signals the tech sector's increasing contribution to the city's economy. Part of this boom stems from a major research university located in the city; its young college graduates are often recruited and hired into the city's tech sector, or they go on to use the sector's wealth of social capital to begin businesses of their own. However, this same sector has a minor brain drain. With their careers just starting, many young tech workers use their first jobs in the city as springboards into better opportunities in other geographical locations. The combination of economic boom and leaky tech talent swirls in the midst of a predominantly White city that suffers from *de facto* racial segregation.

Provided through a non-profit organization committed to equity in the community, Clearwater trains low-income women and people of color in web development and workplace skills for three and a half months. Staff help students apply for paid internships or full-time jobs with Clearwater's tech sponsors or with a different employer; other students may go on to run their own businesses as freelance web developers. Whatever the career path chosen, Clearwater

hopes to help some students join the booming tech sector, slowing the exodus of tech talent, diversifying tech companies, and promoting students' social mobility.

Instructors at Clearwater Academy know the potential mismatch between students' cultural backgrounds and the work culture they enter. While students finish up to 400 hours of programming exercises through an interactive website and complete web design projects, two-thirds of the curriculum teaches soft skills. Most Clearwater students often don't have college degrees and, according to Richard, the technical skills instructor, most students had "gas station and retail" as their "best [career] opportunity." Seeing the lack of formal experience working in a tech office, Clearwater's ultimate mission, according to Janet, Clearwater's case manager and lead instructor on soft skills, is "[create] employees who are ready to work. Especially in the tech field . . . We want to create employable people, and that is what we care for." Janet emphasized, "We are not a school; we're an *employment and training program*."

In previous years Clearwater did not include in its curriculum workplace preparation. Instead, instructors focused on teaching students web programming. As a result, students who graduated and found internships performed poorly in the office. Employers explained to Richard, "There's no way these people should be in our office working." He went on to explain, "These people didn't know how to work, be employees. They had very bad habits. Social skills." We can glean from a work etiquette quiz that Clearwater has sensed retired to prepare students for work these social and cultural tensions between Clearwater graduates and tech companies, as each question was based on actual incidents in previous internships. In the directions, students were asked give their "honest answer [on] how you would act in the work environment." For example:

2. While you're waiting for your computer to be fixed . . .
 - a. It'll take a while. Kick back, relax and take a quick nap.
 - b. Pull out the employee handbook, prop your feet on the desk and read it.

- c. Find your supervisor and ask them what you should do in the meantime.
- d. Ask your coworker if they need help.

Other incidents the quiz covered include negotiating salary, how often to take a smoke break (the first choice is “8 – one every hour”) and having a consistent work schedule. By the start of my study, Clearwater had revamped its curricula. One change was communicating in the syllabus that class would be treated as a worksite, so instructors teach students explicitly how to behave as professionals sometimes through lectures every day classroom interaction among their “co-workers” and instructors. The stakes, Richard thought, were high for these students because “if we don’t teach those skills, they cannot survive.” The increasing number of refugees and migrants further complicate Clearwater’s workplace training as instructors must help them learn English and understand the contours of Western cultures in the workplace at the same time.

A specific concern for Clearwater instructors was how to prepare students for navigating white racialized spaces. Richard and Janet often discussed with their students the gender and racial dynamics of working in tech. The day before a morning of mock job interviews that would take place downtown, for example, they cautioned a Muslim student to consider how passersby or employers during the interview might react to their wearing a thawb. In another instance, Janet tried to help an African American woman code-switch and speak clearly because her accent would be seen as a barrier to successful interviews. Richard himself shared his own strategies to show clients that they can trust him. Being a tall and large African American man, Richard knew he could appear intimidating, so he acted friendlier and wore a suit and tie during business meetings. Janet knew that she was asking her students of color to conform with whiteness but that’s how they must live in order to achieve some level of social mobility. During an interview with her and Richard in the summer of 2017, she explained that, “It’s hard for me as a white person because I ask people, ‘Come to my side.’ It’s not fair and I hate it. But at the same time, we have to make decisions. Do you want a job or do you not want a job?”

Clearwater's methods address criticisms that the local tech companies hire from top universities who culturally fit in. My goal in this chapter is to closely examine how African American adults carry out their coding literate lives in this local context after having some accelerated training in both coding and workplace cultural practices. How do they interact with Whiteness in such a way that "the door of Opportunity," as W.E.B. DuBois describes African American success, doesn't close in their faces, and what happens when that door is closed, or when they never even approach the door?

Data Collection and Methods

My argument draws from interviews with African American adults who participated in a year-long ethnographic study at Clearwater in the Spring and Fall of 2017 and one interview with a highly successful full-stack developer who graduated from Clearwater two years before the start of my study. Clearwater Academy intends for students to find work in tech, whether as a paid web developer intern, as a full-time web developer, or as a freelance coder. In other words, students would go on to apply their coding literacy knowledge in different contexts to improve their lives. The conclusion of my study would track how participants were using their coding knowledge in the workplace to navigate racial inequality.

All twelve adults initially agreed to do two follow-up interviews: one interview three months after graduating and a second interview an additional three months later. To be respectful of their life circumstances, I gave them two options for interviewing: an in-person interview or a video chat. Only one participant was available to meet in-person for both interviews. I interviewed all seven participants from the Spring 2017 semester but only two participants from the Fall 2017 semester. My argument also draws from chance encounters with two participants in public locations who did not respond to my emails, text messages, and calls asking formally for an interview. Although brief and unstructured, learning about what happened to them after

graduating complements my longer, semi-formal interviews about how their experiences at Clearwater informed, or did not inform, their own lives. These follow-up conversations were made possible because I established rapport with participants during my visits to Clearwater Academy. I was not merely a detached interviewer and observer, but rather a student of study participants' life experiences and reflective learning while offering my own expertise in writing resumes, cover letters, and other job-related documents. Throughout my year of class visits, I came to understand the practices these students learned from Clearwater, and their expectations and motivations for applying the knowledge gained. However, to broaden my understanding of student experiences, I also asked Janet if she might connect me with previous graduate students. She identified four successful students; however, only one responded to my requests for an interview.

My initial interview protocol focused on participants' experiences in their internships; as I conducted follow-up interviews, however, I learned that between the two cohorts only two participants received internships. Other participants' lives diverged away from any career related to computer programming. For example, a year after graduating, one participant lived out of his sister's truck and worked a job installing Wi-Fi. Three participants decided to attend a local community college, while a fourth found work repairing computers back home in the Southwestern region of the United States. For participants who did not receive an internship, I asked one open-ended: "What has happened to you since graduating from Clearwater Academy?" For my six-month update, I asked: "What's the update on your life now since we last spoke?" I discussed with participants only topics from the original internship questionnaire that seemed to relevant to the study, such as the value of programming to their lives, if the realities of post-graduate life met their expectations, and was attending Clearwater a good investment of their time. Both follow-up interviews lasted from 35 – 60 minutes.

I used grounded theory to understand how coding literacy was integrated into participants' lives after completing Clearwater and what opportunities that education produced. After listing all possible codes in each transcript, I did a second round of coding to reduce my initial list of codes into categories. From that second round, I discovered multiple themes: multiliteracies, isolation, stereotype threat, stereotypes, whiteness, and job/financial security. After re-coding participant interviews for accurate themes, the data slowly revealed how social and financial contexts influenced the ways coding literacy did and did not get used and the alternative strategies taken to advanced forward in life or maintain some form of sustainability.

For those with whom I spoke, coding literacy appeared to have the most value in the workplace when participants followed Discourses of whiteness to access opportunities for social mobility. Outside of this Discourse, however, participants re-purposed coding literacy, or knowledge associated with coding literacy, to work in their own lives. In this second case, participants adopted two strategies so that coding still resonated in their lives. First, participants found a means to keep practicing coding or place coding on reserve in anticipation of one day using it. I call this strategy *coding maintenance*. Second, participants traded coding knowledge for well-established literacies that had better chances of their receiving rewards. Programming competes within rather than dominates over other literacies, suggesting that coding literacy, at this moment, may be yet another option in our repertoires of literacies. I call this second strategy *trading literacies*.

In the following section, I discuss 23-year-old Isaiah's internship experience. In the first half of the analysis I focus on how Isaiah encountered expectations of whiteness despite his company's interest in diversity and inclusion. I show that middle and upper management believed Isaiah's coding literacy knowledge and practice was a deficit, which increased his frustrations and ultimately led him to leave the internship. Then I shift attention to Isaiah

adopting coding literacy into his own set of literacies to reactivate his journey toward social advancement. I highlight the process of coding literacy's diminished promises as a prestigious resource for mobility in the face of pressing demands for financial security but that it's no less valuable for Isaiah. As I walk through Isaiah's story, I draw on the voices of other participants to solidify my argument on whiteness and the ways coding literacy operates outside the tech industry.

I choose Isaiah as my focal participant because he encountered the most severe and detailed forms of Whiteness of the two participants from the 2017 study who did internships. I place his story in conversation with other participants, including Sufan, the only student in my study to have started a career as a full-stack developer, to further highlight the variations of experiences that help build this chapter's theory. Thus, I push a theoretical argument that's as grounded in and true to participants' coding literacy practices as possible.

Perceptions of Failure According to Standards of Whiteness

Isaiah started interning for Wilburt Credit Financial Assistance (WCFA) a few weeks after graduating from Clearwater Academy in May 2017. He was one of six African Americans among the 72 interns WCFA hired for the summer. Isaiah was also the only intern who wasn't a college student. Unlike his majority white, senior undergraduate counterparts, Isaiah did not know any programming languages for managing the servers and databases that held users' and the company's data. He worked with a team of experienced developers and had a peer mentor to teach him C# (pronounced C sharp). But the combination of whiteness and his lack of coding literacy hindered Isaiah from using coding literacy for social mobility.

The internship seemed to be a productive location for learning coding literacy before Isaiah encountered race-based perceptions of deficit. He enjoyed learning from his peer mentor, who for the first two weeks met with Isaiah one-on-one to prepare him for backend projects and

assign him work in HTML and JavaScript. She gave Isaiah videos and articles on C# to study at home. If he didn't understand a concept, the peer mentor would clarify or answer questions the following morning. However, after she returned from a week-long family emergency, the peer mentor "started falling off really heavy but in like little small portions." She cancelled most of their morning meetings. Despite no longer meeting in person, the peer mentor sent e-mails to ask if he had enough work, and if not, would assign Isaiah additional projects. But this soon stopped, too. Instead, Isaiah learned, the peer mentor placed more interest in teaching her other intern, a white college student set to graduate next year; the meetings she cancelled the individual meetings with Isaiah and replaced that with meetings with the intern. Occasionally, the peer mentor would have joint meetings with them both. Nevertheless, the peer mentor showed more concern about his whereabouts in the office than Isaiah's own learning needs. During a demonstration of objects and classes, two crucial features used to program software, the peer mentor spoke directly to the intern, as if Isaiah was not in the room.

The relatively swift deterioration of Isaiah's coding literacy education while on the job stemmed from perception of his poor coding knowledge and language practice. The peer mentor sent him to see Isaiah's manager, who quickly summarized the shortcomings his mentor observed: "You haven't really learned backend or 'you haven't zero experience,'" said Isaiah. "They just basically said you don't go home and study. You don't do this. You don't do that. You're not as articulate enough as the other interns." The peer mentor's assessment of Isaiah's coding showed in joint meetings with the white intern several weeks later. At the start of one meeting on C#, the peer mentor said to Isaiah with dismay, "Look, I was actually skeptical of inviting you to our meeting because I didn't think you'd understand anything relative to the subject. I don't know if I can dumb it down anymore for you." Nine weeks into the internship, the peer mentor told Isaiah that she didn't want to teach him.

These observations and interactions evoke unstated expectations of white performance in coding. By recruiting mostly white college students, WCFA creates a white racialized space. These students' past experiences become the baseline criterion for judging *all* interns' coding performance and office cultural knowledge; because white interns make up the workforce, this standard root itself in knowledge and language practices of whiteness. Management has little room to adapt their standards to Isaiah's education from Clearwater or his "background of thugs," as Isaiah described his growing up. They expected Isaiah to adapt *his own racial performance* according to their default expectations—talk about coding and other general topics well, know the same kind of coding as privileged college-education interns—or fail. Isaiah has been aware of this pressure to clean up his African American English for white spaces since before the internship. In high school, Isaiah's close friend, and the only Clearwater graduate in my study to have a successful career in software, Sufjan warned him, "Look, you can't talk like this everywhere you go. You're not going nowhere." Learning from Richard in Clearwater, Isaiah knew he had to indicate his social position using language. Still, his performance in the internship was not enough; whether it was his lack of backend knowledge or he isn't "articulate" enough in his communication, Isaiah failed the whiteness test. This caused fractures in his relationship with his peer mentor and manager, but it also denigrated Isaiah as an African American and as a code bootcamp student; he was less than the white college educated students Wilburt Credit Union hired. Hence, his peer mentor and manager thought Isaiah came to the internship with a deficit and was too troublesome to teach.

Performing for whiteness persisted for Sufan and Zelda. Zelda was the only other graduate in the 2017 to have an internship, and Sufan graduated from Clearwater in 2015 and had become a highly successful full stack developer over three years. For Sufan, the consequences of external racism prevented his enjoying his first internship and practicing

coding. He loved his diverse classmates in Clearwater, but Sufan admitted he was unprepared for working in a small tech consultancy who employed majority white workers. For the first three months, Sufan experienced stereotype threat, or the fear that one's actions or characteristics "conform" to making a stereotypes "more plausible as a self-characterization in the eyes of others, and perhaps even in one's own eyes" (Steele and Aronson 1997). Reflecting on those first few months, Sufjan recalled telling himself, "'Oh you had to be on point or you suck.' Because of [the] whole stigma of being black and you had to work two times better . . . I was trying to fit those shoes." While Sufan felt stereotype threat and Isaiah encountered microaggressions, Zelda found herself appeasing her white counterparts so they might feel safe with her presence. Zelda worked data analytics for a small company that sold and maintained patient management systems for hospitals. Zelda admitted that making connections with her co-workers went faster than most because she had a young daughter that could pass for white. When co-workers noticed a photo of Zelda's daughter on her desk, they immediately warmed up to her. "I feel like, it just makes people feel more comfortable," Zelda said. "It's like a level of—here's this level of acceptance . . . Like, 'Oh, this is a Black person who's been around white people and happens to have a white baby.'"

These participants' experiences highlight the different ways whiteness appears in spaces that rely on coding literacy practice. Programming, like reading and writing, fold themselves into sociocultural contexts that shape how other coders of color get judged and treated. Perceptions of these marginalized people can have significant impact on the actual coding practice done. In the following subsection, I give further detailed description and analysis of how a persistent focus on failure impacted the coding literacy practice of Isaiah and the trajectory of his life after departing from the internship.

Invisible Black Coding Literacy

The perceptions of participants' so-called deficit in programming knowledge and work ethic made invisible or unacknowledged the knowledge they did possess and use; this troubled participants' coding literacy practices. Learning C# was new for Isaiah, but he was willing to learn. But he spent more time defending his ability to learn than actually learning. For example, he believed strongly that the student should be responsible for telling their teacher that they don't understand something. When the peer mentor said she had to dumb down concepts, Isaiah protested immediately. He explained to her, "It's not about dumbing it down, it's about talking to me too, not just to him [the white intern]. Cause I'm not understanding when you're going super fast explaining this, for one. And then jumping from thing to thing." Isaiah defended his ability to learn again when the peer mentor confusingly compared objects and classes to drugs and medicine. The peer mentor thought the analogy fit, but this visibly frustrated Isaiah. The peer mentor misinterpreted Isaiah's anger as his being "passionate" for the subject and suggested that he just needed "a challenge."

However, Isaiah was already working hard to "grasp" C#. Recall from the previous section that Isaiah's manager said he didn't practice code at home. That wasn't true at all. During his mentor's week-long absence, Isaiah attempted to learn C# on his own while doing projects in HTML and JavaScript at work. In the middle of the internship, Isaiah was in the process of moving apartments, so he didn't have Wi-Fi at home. As an alternative, ten minutes before and after official clock-in and clock-out time, Isaiah would read about C# at his desk. He worked on additional code to the point that he and his girlfriend got into arguments over his doing too much. He codes and codes "until I'm ready to go to sleep." Isaiah was insulted, then, that management didn't understand this, and he didn't appreciate that they treated him as if he were "dumb."

The criticism that he wasn't articulate further made invisible his labor at communicating effectively in a white-dominated profession. Isaiah admitted that he needed to work on his word

choice at work, but this self-observation didn't prevent the impact of his being "inarticulate" had on his interaction with his peers. "I stumble over my words every time I speak to her [the manager]," said Isaiah. "I'm that much more conscious. And then every time I speak, I either stumble on my words or stutter. And it's because she sat there and told me that." Isaiah had made several leaps in his language practices to hide his African American English. But the perception of his poor ability to speak, whether about coding or speaking in Standard English alone, reduced his language to stumbling and mumbling through conversations with management. And he could not ask for help or show that he understood coding well for fear of confirming the racial stereotypes already placed on him.

Despite Isaiah's efforts to mostly self-learn coding and maintain a good rapport with management, he never earned the three-month extension that he needed to financially support his son who was due in September. Denying him the extension for being unfamiliar with coding that the company themselves had promised to teach him was "unfair"; it wasn't fair that Isaiah never "got a chance" to show them his worth. When WCFA offered him to leave two weeks early with pay, Isaiah accepted it because "I didn't like my experience there; I didn't really want to go and keep going through that stuff." His son was due the following week. I just said, 'All right. Cool. Sounds great.'"

These events demonstrate how whiteness mediates relationships and coding practice. How one applies white ideology to coding literacy determines the kind of assessment applied on Black coders. For a literacy sponsor that judges worth according to whiteness, this assessment is a process of searching for the *familiarities of whiteness* within a Black coder. But that comes at serious cost: failure to acknowledge Black coders' unique thoughts and ways of learning can make them feel unwelcome or out of place. As Isaiah's experiences indicate, disrespect is a social act that derails a talent's will to practice coding or belong to a Discourse. What is

supposed to be a space for developing Black academic achievement and freedom, turns out to be yet another roadblock to meaningful social advancement and liberation. The culture of coding literacy sets the fence too high and the ladder to navigate that fence too broken. The challenges stacked immediately against them, Black coders end up doing coding literacy differently or, as we'll see in the next section, not at all.

External, concrete manifestations of whiteness can make invisible the coding practices of racially marginalized people in other ways. Both Sufan and Zelda seemed prepared to accept that coding would not benefit them. Sufan's stereotype threat re-shaped his approach to coding and learning coding at his all-white tech consultancy internship. He worked hard to avoid appearing coding illiterate. "I was always thinking, 'Oh folks are going to think I'm stupid. And X Y and Z.' So *I didn't ask a question* for like three months while was I there. I would suffer in silence, which you don't do that." Instead of asking questions or asking for help, Sufan pretended to know exactly what he was designing and coding. Withholding the gaps in his coding literacy knowledge due to his fear of coming "off as this Black dude, you know, that BS," only put more pressure on Sufan's sense of self, as he found his identity slowly slipping away. Trying to play an atypical Black man performing as a white coder, Sufan thought, actually made him unsuccessful for those three months. Sufan only felt better when he focused on trying to do good work and getting to know his team of developers during a road trip to Texas.

Zelda didn't feel stereotype threat because she focused on meeting her deadlines to counter those feelings. She worked in a company small enough that the CEO and vice president held meetings with the entire staff. As her six-month internship progressed, Zelda's relationships with the vice president worsened, partly because the vice president, and other managers, failed to acknowledge her work, as if she didn't exist. The most grievous of these instances was after Zelda re-designed a conference website; the vice president said the video on the website looked

very impressive. When Zelda asked for further feedback, or at least acknowledgement of her work, the vice president responded that links to the conference speakers' biographies were missing. That feature, Zelda explained, was someone else's responsibility. By the end of the internship, Zelda was assigned little work and remained largely invisible to others; a waiting game to the end of her internship (while still getting paycheck) and the start of her vacation ensued. Although Zelda thought her treatment had less to do with race because the vice president treated white employees the same way, the implications of ignoring Zelda's work stands. Ignoring the coding literacy of African Americans inadvertently undoes the cause of diversifying tech.

This section has shown how ideologies of whiteness and deficit thinking influences the coding literacy practices of African American computer programmers. The rules of doing, learning, and talking about coding literacy have traditionally been under the purview of white computer programmers. Indirectly, then, their ownership of these tools have helped create a culture that valorizes their backgrounds. African American participants in this study found themselves either needing to or feeling the pressure of needing to assimilate into this Discourse of whiteness in order to do programming. Those pressures come out of interactions with sponsors who signaled how the Discourse on coding works; how well African American coders ride this wave shape their approach to coding and determine to what extent they drop out of tech. In the next section, I investigate the consequences of stepping away from coding literacy and its opportunities through the software profession. I describe how a different set of social inequalities outside the tech workplace fueled two kinds of literate strategies: *coding maintenance* and *trading literacy*.

The Coding Life After Clearwater Academy

Coding Maintenance

In this section I describe coding maintenance—one’s effort to continue to learn and maintain knowledge of programming outside the profession. This literate strategy places coding literacy in a vault, so to speak, where its tended to in the event one needs to use it. In this instance, coding never gets lost despite not being used for a career. I continue to describe Isaiah’s story to demonstrate this concept. I also supplement his experiences with DeAndre, a 21-year-old Clearwater classmate of Isaiah’s.

Isaiah’s son was expected to be born in September; he had hoped to do well enough on his internship that he would get a six-month extension. Then Isaiah would continue to pay rent and provide for his new family. WCFA ending his internship in August, however, left Isaiah open to the risks of financial precarity. He needed a job quickly. Isaiah preferred working in IT, but the difficulty of accessing or re-accessing opportunities in software development and life responsibilities put the actual worth of coding literacy to the test.

Stepping away from the structures of opportunities available in software often led to coding maintenance—practicing coding literacy in hopes that someday it will be utilized for work. Despite his rough time at WCFA, Isaiah still valued coding. As he began his exit from this internship, Isaiah applied for frontend developer JavaScript developer internships, and he learned from a connection that he could possibly work a six-month internship that paid fifteen dollars an hour at another company. As explained before, this money would be timely and useful for supporting his family. Isaiah also hoped to learn more coding than he did at WCFA to increase his chances of getting a fulltime job. While waiting to hear back from employers, Isaiah met with a group of friends to practice coding. These attempts at maintaining coding knowledge reflect a mission: to code until Isaiah got rich from his own clothing line, built largely through the combination his graphic design knowledge and frontend development skill. The company would bring in “significant income” that he could use to give his son a better life.

Similar to Isaiah, DeAndre, a 21 year-old Midwestern native who graduated from the Spring 2017 cohort, practiced code in anticipation of securing work in software. Attending Clearwater while making little money was “very stressful,” so he took a break from coding for a few weeks after graduating. DeAndre had applied to an internship but never got a call back, so he found work as a cook at a Southern-themed restaurant. Then his cousin hired him to write the shell for an adult website; DeAndre would code the rest of the website once his cousin produced the videos, but those videos were two months overdue when I interviewed DeAndre. In his spare time DeAndre also practiced learning PHP, SQL (pronounced “sequel”), Swift, and Python. Unlike in Clearwater where he had to learn HTML, CSS, and JavaScript all at once, DeAndre was learning each language one at a time, and instead of learning coding through the interactive website Clearwater used, DeAndre learned from “doing a whole bunch of shit--just making random websites. Fuck up, fix it and then find some new shit . . . You never know.” DeAndre intended to leverage his coding skills into a fulltime job one day; he would have to do further research, he said, but he would build his reputation as a coder and then use that reputation and experience to work for himself as a freelance programmer.

However, roadblocks to transforming coding maintenance into a resource that generated income persisted. Isaiah got a last-minute invite to do a code interview the next day. Because he still worked for WCFA, however, Isaiah only had the lunch hour to study for the upcoming test; the interview didn’t go well, Isaiah said. It was the same for other interviews. “I was rattled because a lot of places didn’t want me because I didn’t know certain things that I was supposed to gain from [WCFA] and I didn’t,” said Isaiah. He considered being a freelance web designer but found that customers wouldn’t pay enough for his services, especially when most people could create their own website using a template from Wordpress. Kevin, a graduate of the Spring 2017 cohort, also tried applying to software developer jobs when he returned home in the

Southwest. His infectious personality helped him build a good rapport with interviewers. Still, they could not hire him because he lacked backend programming. “Literally . . . every coding job” told Kevin the same thing. He could apply again in four to six months; if he would “get some more projects under your belt” during that time, Kevin would have a better chance of being hired. There were plenty of software jobs available in his home state, and Kevin could easily talk his way through interviews, but none of that mattered without the proper coding knowledge.

As multiple avenues for computer programming closed, coding literacy’s utility depreciated in value because it failed to meet needs for security and mobility. Consider the shift in expectations and goals between my second interview and third follow up interview with Isaiah six months later. In August, he insisted on working in IT but he did think non-tech work was a possibility. “At the end of the day,” said Isaiah, “I would like to be in IT, but if the job I get is not IT, that’s what I got to do. Because I got to provide. At the end of the day, *I just need to get a job.*” Later in January of the following year, Isaiah’s limited knowledge in coding, according to employers, discouraged him from continuing to code: “I still know how to code; I still mess with it sometimes, but as far as like do I still *code*? No.” Isaiah had learned that coding had only one sure location to meet his financial needs, and when that pathway led to nothing, he relinquished it as a resource.

Similarly, coding literacy had no utility for Kevin’s circumstances because he didn’t have enough experience. After graduating from Clearwater, he intended to stay in the Midwest to complete a Swift code bootcamp, but he felt something was wrong at home. So Kevin cancelled his plans and returned home to find “chaos”: his house was a mess, his truck was broken, and his mother’s health “had gone down.” He needed to put his life back together. With no luck finding work in software, Kevin set coding literacy aside, explaining, “The thing is, when I think about coding it’s just like, ‘What do I want to do with this?’ I just really haven’t found a place for it.”

Finding a place was also how Rosie described coding literacy in her post-graduation life. Also a graduate of the Spring 2017 cohort, Rosie was diagnosed with lupus prior to attending Clearwater Academy. At the suggestion of her doctor, she retired early and took care of her mother and brother who had cancer. Rosie asked her other siblings to look after them while she attended Clearwater. But after graduation she resumed her caretaking duties. Rosie just “hadn’t been able to move it [coding] up on any kind of scale . . . *It’s just my life.*” The closest Rosie came to coding was writing web content: she input a Black Women’s conference schedule into a pre-coded app, she helped write a friend’s web portfolio, and she regularly updated her lupus support group’s website with links and other information. Rosie felt that she had put coding literacy, and Clearwater’s resources for starting her own business, on “reserve” until an opportunity came to use coding. She joked and still dreamed of one day working remotely from Tahiti. Until then, her family was top priority.

Coding literacy has power if coders’ sociomaterial conditions are already somewhat secure. The ways of accessing jobs that reward coding literacy with mobility require careful research on where one can meet the highest of literacy standards, whether that’s freelance work or a fulltime job with a company. But the process of eventually activating coding literacy can take too long, or the process doesn’t keep pace with the speed of social inequality in racially marginalized people’s lives. Racially marginalized people, at least in this study, hoped that *coding literacy would provide* job security and mobility, not that they needed security to wait for the return on their investment in coding. Poverty and health ultimately overwhelm the process of using coding literacy for mobility. With no real results or even a hint of its utility, coding literacy can diminish as a viable resource for African Americans computer code bootcamp graduates. This observation leads to the second literate strategy that makes up for coding literacy’s potential weakness—trading literacies.

Trading Literacies

In this section, I describe *trading literacies*—switching emerging digital literacies such as coding for more established and traditional literacy practices to better adapt to sociomaterial conditions. This literate strategy indicates that the value of coding literacy depends its affordances compared to other kinds of literacies. This strategy suggests that even dominant literacy practices (such as writing) belong to a larger ecology of literate practices; any one practice can lose its dominance within this ecology based on sociomaterial conditions. To further show this phenomenon through the lens of Isaiah, supplemented with three other participants: Cherri, Kevin, and Zelda.

With structures of opportunities for coding literacy limited and the need for job security and financial stability deepening, other established literacies became a safe and certain option. Thanks to his uncle, Isaiah found work cleaning corporate offices in the city. However, he had to clean more floors than usual after the company fired his co-worker, making Isaiah's returning home on time to care for his family difficult. Luckily, his girlfriend got a job in pharmacy which paid enough to support the entire family. At the insistence of his girlfriend, Isaiah quit his cleaning job to pursue his dreams. In January, nearly a year after starting Clearwater, Isaiah explained, "I want to make it as an artist is really what I want to do. So I'm a full time artist as well as full time dad, doing my art." He planned to create a website to sell his art and had returned to his community college to complete his degree in graphic design. It was a two-year program, which isn't quite as fast as he wanted, but Isaiah believed re-investing in college would pay off: "At the end of the day, I look at it like, I want to do something I'm going to be happy about, passionate about." He would still start a clothing line and, who knows, make enough money to spend time with his son and travel to countries in East Asia.

Both Cherri and Kevin, classmates from the 2017 cohort in the study, found themselves working closely with hardware and software as alternatives to coding literacy. Cherri needed quick financial and job security to take care of her two daughters; an internship did not guarantee long-term employment, so Cherri returned to her prior work in banking as a loan liner for Wilburt Credit Union. Knowing JavaScript helped Cherri get this job, although in reality she rarely used JavaScript for work. Clearwater gave Cherri “a nice understanding of what it is and what it’s used for kinda,” but she “knew right away that coding really wasn’t my thing . . . that part of learning computers just didn’t interest me as much.” Still, she was prepared to use JavaScript if they needed her to, a kind of placing coding on reserve, which I’ll return to later. Cherri was interested in help desk; after speaking to a community college instructor at a job fair, Cherri enrolled in a Systems Administration certificate program that would train her in troubleshooting common software and hardware problems. Meanwhile, Kevin returned to the Southwest only to work in the same cooking job he had before going to Clearwater, “doing exactly what I didn’t want to do for the rest of my life and now I’m doing this for the rest of my life.” His managers suggested Kevin leave the job and figure out what he wanted to do. After trying to get a few software development jobs, he found work doing service repair for a major computer retail company. Like Cherri, Kevin rarely needed to repair customer’s computers using coding, but Clearwater’s lessons on learning how to learn made him highly successful at his job.

Other literacies had clear pathways toward mobility, unlike coding literacy. Cherri planned to transition into Wilburt Credit Union’s IT department after getting her certification; she had arranged a meeting with someone in the department to discuss how her education could land her position. Kevin’s quick self-learning and problem solving helped him teach his own instructor something. The instructor was so impressed and shocked that he invited Kevin to help train new employees. Still, Kevin thought his “ultimate skill” wasn’t tinkering with computers

but knowing “how to talk to people, I know how to make people comfortable with me . . . I’m very perceptive and intuitive. Those are probably my—personal skills—those are probably my two strongest skills.” Kevin set his sights on doing work in child welfare but first he needed to get his credit card debt under control.

The most striking literacy trading was Zelda who, even as an intern, searched for a programming language more interesting than JavaScript, something like Python, Ruby, or Swift, but poor interactions with management—being ignored during a meeting, seeing other co-workers get disrespected, and not receiving recognition for her work—made Zelda re-consider her options. The good thing to come out of her internship was discovering user experience (UX) design and research. Related to coding, UX designers design clean interfaces so users navigate a website or app easily; Zelda learned that UX design was in high demand, so there was plenty of money to be made. She would look into starting her own company and build her mobility from there.

The systems of reward for coding literacy has standards that may be too difficult to meet, standards that may implicate whiteness if African American adults even get passed the interview. This makes efforts to leverage coding into a better social position difficult for a pool of students who don’t come from or haven’t learned that Discourse. As a new emerging literacy, computer programming seems to have limited structures of opportunities that participants can’t access fast enough to address their social inequalities. Coding literacy competes with other traditional literacies that have fewer institutional roadblocks and a wider range of opportunities for possible mobility. These literacies have more utility for participants’ sociomaterial conditions compared to coding literacy. They’re more trusted, more familiar, more approachable. This shows how the position of coding literacy as a writing activity for social good is contingent on the social

resources racially marginalized people have access to and the severity of the social inequality they encounter.

The combination of whiteness and social inequality may overwhelm whatever benefits coding literacy may offer. Yet coding literacy resonates throughout people's lives in various ways even as white supremacy forecloses opportunities to make a living from it. As seen in the examples above, coding maintenance and trading literacies offer alternative means to using prestigious emerging literacies in a variety of ways. If participants do not come to rely on coding literacy as a primary means for social mobility, they may add it to their existing set of literacies, or coding literacy is set aside and tended to in anticipation of an opportunity to use it. As explained earlier, some participants found that they needed some stability in place while searching for ways to use coding literacy for social mobility. The two strategies identified here are more grounded in the kinds of security participants do have—they demand only as much as participants are able to give, and participants show more agency over these strategies rather than behold themselves to the long process of hiring and then learning how to navigate white racialized spaces. A computer code bootcamp may be useful in evolving the repertoire of literacies marginalized communities have or help them re-see the possible ways they can use their set of literacies to navigate their world. This suggests that computer code bootcamps may consider how the introduction of coding can shape students' literacy practices for a variety of needs in their own lives.

Conclusion

Computer code bootcamps ride on the rhetorical waves of broadening opportunities to learn computer coding; some wish to flatten access to coding literacy with the specific intention of broadening participation in software development for racially marginalized people. This research study described how African American adults live out this rhetorical dream and attempt

to leverage coding literacy for mobility and greater social position. Participants may avail themselves to invisible standards of whiteness that surround coding literacy practices. Those standards allow for accessing to the opportunities that Clearwater hope students get. However, navigating white racialized spaces may trouble their ability to practice coding. Certain contexts maintain rather than breakdown racial hierarchy through software production. More pressing is life after leaving this context, participants come to experience coding literacy, as not fitting, not having a place, or being something, they do on the side. This calls into question how exactly the promises of coding literacy happen for racially marginalized people.

Findings suggest that while people of color may get opportunities to learn coding or computer science, only a few African Americans will ever join the class of professional coders. Coding for all may lead to a general use of coding, but even in this context, certain coders will control the ways technological systems work for other kinds of coders. Success in these spaces implicates how well a person of color fits a culture of whiteness, stated and unstated, and whether or not they find support elsewhere in the midst of these white racialized spaces.

Nevertheless, this study reveals that participants have a rich tapestry of literacies, and when one literacy loses its value participants shift horizontally to something else and then get savvy with those literacies, if they are able. The value of African American literacies, after all, has been in using African tradition to ground the community's social position and fight for civil rights. That is, they used their literacies according to their rules and not the rules of white society that have been historically stacked against them. The strength of African American culture, language, and knowledge, as they circulate across digital media platforms and, one day, deep into programming, adds, in the words of Isaiah, "some flavor" to our technocultures. Perhaps to the benefit of us all.

Chapter 5

Conclusion: Critical Imaginings of Coding Literacies

“See here’s the thing, you talk about people of color. 2040 is coming. White people will be . . . the minority. So, we’re about to be a brown country. 2040 is coming, so all in all, you know, I’m not going to make this about Black people, because like, everybody should [learn to code]”

—Zelda

During my interviews with participants about life after graduating from Clearwater Academy, I asked if they recommended that more Black people learn coding. In response, Zelda hints at a pressing concern for the future of coding literacy. Since the conception of the United States, white supremacy has gathered literacy and its rewards for white people, driving much of the infrastructure of the United States to its benefit. But the deliberate and systematic neglect of sharing such powers with people of color may have dire consequences as White recipients of the country’s rich economy and global political domination recede. Having no equal or equitable access to literacy, it seems that people of color would be unable to carry forward competition and innovations of late capitalism as an aging White population persists (Frey). Even though Zelda calls for everyone to learn coding, the gradual “browning” of United States’ population underscores the point that all people (of color) should take advantage of this moment in our digital literacy history.

However, that depends on the work afforded to us now as the early stages of coding for all continues. Questions about this process remain unanswered: how do we train computer science teachers? How do we design appropriate curricula for marginalized people in different contexts? What is the desired end goal of coding literacy education? Open-ended questions like these should be grounded in our understanding of race to inform the ways coding literacy practice gets taught. In looking ahead, how might we use the infancy of the computing for all movement to

construct ways of resisting reproducing social stratification through computer programming?
 What should we do next?

“What should we do *next*?” seems apt because it typifies African American literacy and rhetorical tradition. Looking ahead follows the *hope* that reading and writing will mediate liberation from slavery and second-class citizenship. Among other kinds of writing, Afrofuturism participates in that hoping, tossing science fiction and African and African Diaspora lived experiences into a stew to “re-contextualiz[e] and [assess] history and *imagin[e] the future* of the peoples of the African Diaspora” (Castro, emphasis mine). Countering notions that literacy, science and technology, and education guide (White) Western progress, African American literacy and rhetoric seems, in general, to be more focused on a different kind of progress: making the imagined futures of liberation real. Working toward a future of liberation recalls the feminist rhetorical method critical imagination. Using this method, the scholar gathers and “configures” the evidence they do know, and then they “think between, above, around, and beyond this evidence to speculate methodically about probabilities, that is, what might likely be true based on what we have in hand” (Royster, Kirsch, and Bizzell 71). Critical imagination makes sense of who and what may have been in the past, *of what is possible*. In my conclusion to this study, I practice critical imagination to consider how we might conceive of coding literacy’s role in addressing racial and digital inequality.

First, I briefly summarize what has been discovered from this study at Clearwater Academy: I argue that the findings from the previous chapters demonstrate that coding literacy education allows a community of historically marginalized people to continue practicing sustainability and resilience in our moment of technological change and racial formation. Based on community practices of sustainability, and in looking forward to possible futures, I suggest that there’s a possibility that we understand coding literacy as a sociocultural and embodied

practice. This interpretation, I argue, helps broaden and complicate the function of computer coding. Then I describe two possible functions drawn from the experiences of participants that computer code bootcamps and scholars in Writing Studies may consider for further research and teaching. First, that coding literacy is a tool to be integrated into our everyday lived experiences for problem-solving. Second, coding literacy is a resource to be shared widely within a community and then widely used for addressing that community's needs. On one hand, then, coding literacy functions to enhance individual lives; on the other hand, professional coders, especially coders of color, may distribute their knowledge of technology to support their own people. In both cases, survival or sustainability serve as the philosophical goals for computer code bootcamps that apply a social justice framework.

Using literacy as a tool for social mobility is complicated as it requires students' emotional and physical labor to work through racist political policies that regulate their process of social advancement. The two possible functions presented in these last few pages mean to address the codes that white supremacy has programmed to control the distribution of capital and dignity among people. For this reason, a coding literacy education with a sociocultural understanding of coding practice allows for a stronger response to racial inequality both in the culture of a majority white profession and, most important, in our public digital cultures. Emphasizing survival or sustainability for communities raises the stakes of coding literacy practice while humanizing its work. In thinking towards this implication, I describe the limitations of my study and show that these limitations lay a foundation for further research.

Summary of the Study

The main aim of this dissertation has been to trouble claims about the benefits of coding literacy for racially marginalized people. As a new addition to the rubric of writing (Vee), computer coding seems to bring multiple social benefits to its users, such as new methods of

problem solving, a medium for creative expression, political and social empowerment, and participation in the labor market and economy. Such claims recall, and build on, what Harvey Graff calls the literacy myth, the historical and persistent belief that reading and writing is a precursor for the progress of Western civilization and individual transformation in morality, financial stability, citizenship, and political participation. Belief in the power of literacy has galvanized efforts to democratize literacy education, although that process is fraught with exclusionary practices.

The rhetorical power of the literacy myth extends into coding literacy. Sensing an urgency to increase digital literacy skills in response to our current political and economic moment, a range of literacy sponsors—educators, school administrators, non-profits, for-profit tech companies, and the federal government—have helped expand opportunities to learn computer programming in public schools. The computer code bootcamp industry came out of this eager energy for learning computer programming. Taking up the economic benefits of coding as one of the justifications for their existence, these technical career training programs recruit and rapidly train adults in coding to meet increasing demands for more software developers. Some programs invest in teaching racially marginalized people to help their achieving social mobility and to diversify a majority white male tech industry. Recent survey results demonstrate that these camps graduate nearly as many students as computer science undergraduates, and one percent of those code bootcamp graduates are African American, and some of these graduates report increasing their salary.

Still, structural and systemic inequality unevenly distribute access to literacies and opportunities along race, class, and gender. Given this historical and present issue, this study describes what exactly happens on the ground for African American adults as they interact with systemic racism while learning computer programming. The circumstances of learning coding

results in the following research questions: In what ways do students' experiences of code bootcamps help them overcome (or not) the effects of systemic racism on their literacy learning? And what do such experiences reveal about the relationship between race and writing in a moment of technological change?

To answer these questions, I designed a small-scale qualitative study of twelve African American adults attending Clearwater Academy, a Midwestern-based computer code bootcamp. Clearwater Academy belongs to a non-profit organization that provides services to the local community to combat racism, gender inequality, and poverty. To this end, Clearwater Academy trains low-income women and people of color in web programming and job skills (resume and cover letter writing, mock job interviews, elevators pitches, etc.). The preferred goal for Clearwater is that students graduate into paid internships or full-time employment within tech, although any full-time job in any field counts as success.

I conducted participant observation in the Spring and Fall semesters of 2017. My weekly visits included one-on-one interviews on participants' digital literacy histories, the kinds of support they had that mediated their learning coding literacy with challenges associated with structural racism, and their experiences in the class itself. I also conducted group interviews throughout both semesters to see how their expectations of Clearwater matched or did not match the realities of their education and if, given their experiences in Clearwater and experiences as second-class citizens, African Americans as a people should actively pursue opportunities like computer code bootcamps. I followed participants from the first day of class to their graduation night. And then I followed them beyond graduation: I attempted to check-in with participants three months and then six months after they completed their education at Clearwater Academy to understand how coding literacy changed their lives or didn't, what they encountered, especially at their internships, the end goal after graduating from Clearwater Academy.

The research project I have described in these pages, then, offers an in-depth look at the lives of a few adults who attended a computer code bootcamp. In Chapter 2, “Like Coming Home: African American Adults Playing and Tinkering Toward a Computer Code Bootcamp,” digital literacy life history interviews revealed that students draw on personal legacies of informal “tinkering” (and not their experiences of racially biased schooling) to learn computer coding. This chapter suggests we consider how “making” in informal, supposedly resource-poor environments can inform current interest in maker-based pedagogies in education, composition studies, and professional and technical communication. Meanwhile, Chapter 3, “Between Learning and Opportunity: A Study of African American Coders’ Networks of Support” shows that success in a computer code bootcamp in a context of white supremacy necessitated a complex and often ad hoc network of social support that generated different kinds of resources for their continuing coding. Educators are called to look outside of the classroom to the ways structural racism shapes the lives of students and their access to prestigious literacies. Finally, Chapter 4, “Making the Pivot with Coding Literacy,” demonstrates that even after successfully completing the bootcamp, systemic racism hindered their ability to succeed in the tech workplace. Despite limited access to coding careers, participants use coding literacy in new ways to reach other life goals. This chapter reveals that African American adults’ literacy does not automatically lead to upward social mobility nor does coding literacy cease to work as a tool. However, participants are building on the African American legacy of literacy for personal liberation that looks different for different circumstances.

My dissertation makes a Black digital update to the literacy myth by placing a tight focus on the role computer code bootcamps serve as sponsors of African American students who may use coding literacy for social mobility. The path to coding literacy is fraught with material, social, and cultural disparities, but these findings collectively show that throughout their journey

students draw on funds of knowledge that enhance or supplement their access to coding literacy through a computer code bootcamp. We may draw on such knowledge and personal experiences to revise coding literacy's value from an economically-powered and cognitive skill to a sociocultural and embodied practice. Addressing computer programming as a social and cultural practice helps highlight its *ideological functions* and further underscores the role coding literacy plays in addressing, and creating, racial inequality. Two functions I consider here are 1.) integrating coding literacy as a practice for everyday use to strengthen the lives of marginalized individuals and 2.) using coding and the types of social positions it helps create within marginalized communities for re-shaping our digital cultures. In the next section, I describe in more detail what these two practices may look like in a future computer code bootcamp.

Imagined Futures for Two Functions in Coding Literacies

Integrating Coding Literacy into the Everyday

Some computer code bootcamps invest in sending racially marginalized people into the tech industry, filling apparent massive gaps in software development job positions and diversifying the tech industry's predominantly white male workforce. Coding careers of any kind are not guaranteed, given that systems of inequality can impact even the so-called merit-based software industry; however, participants' post-graduate use of coding literacy, as seen in Chapter Three, suggests adjusting instruction for a more individualized use for coding. In this way, coding literacy is an everyday practice. For example, Kevin had not come to use computer programming for a career. Despite getting interviews with many companies, each of the employers gave him the same feedback: Kevin didn't have enough software projects to demonstrate he could be a reliable coder for their clients. However, Kevin leveraged "learning how to learn" into a good paying job repairing computers, and his training with the software

program Sketch led to his creating an automated chore chart for his household. Kevin cited these skills as essential ways to improve his quality of life.

This personal integration of coding literacy enacts a self-intervention method. Applying computer programming to common everyday problems ensures another level of protection to racially marginalized people's families and communities in the face of racial disadvantage. This study suggests that instead of teaching a general, industry-specific use for coding literacy, computer code bootcamps may have students brainstorm the multiple ways coding literacy can help improve their lives. The goal to democratize emerging technologies in other educational contexts, such as in public schools, should embrace the diverse ways one can use coding literacy to ensure sustainability and resilience. Success may rely on how racially marginalized people use coding literacy to strategically engage the personal responsibilities (e.g. coding to locate housing, job opportunities, and childcare services) that make up the contours of their lives.

Coding Literacy as a Community-Based Resource and Practice

The presence of racially marginalized people in the tech sector encourage possibilities for transforming digital cultures more broadly and contribute to sustaining marginalized communities themselves. Richard and Janet, during my observations and in interviews, explained that technologies need to serve the needs of marginalized people. These observations match with reports on a number of automated technologies that actively exclude people of color, such as facial recognition software that doesn't recognize darker skin complexion or if they do, they match those faces with apes; and algorithms that call up majority white people when images searching for STEM careers while presenting images of Black women as sexual objects (Noble, *Algorithm*).

Thus, an additional track teaching a practical sociocultural understanding of coding literacy for careers may include first recognizing that coders of color may gain significant social

positions where they can add their knowledge to the databases and logics of software design and development. Or computer code bootcamps may highlight how students can bring the tools and knowledge gathered from working in majority white tech sectors to their own communities.

Sufjan, mentioned in Chapter Three, described plans to build up communities through the tech hubs in Kenya. He has heard about the ways tech companies in Silicon Valley has pushed out and decimated communities of color. How might Sufjan, and a coalition of other coders of color, approach building tech hubs that support neighborhoods, becoming an extension of their own efforts to survive?

Investment in these two functions for coding literacy practice also help continue a longer tradition of using literacy, broadly conceived, for survival and sustainability. A focus on entrepreneurship and career for social mobility improves a quality of life that is deeply reliant on the strength of white supremacy. As our country invests in computer programming as a type of writing, training programs like Clearwater Academy may sidestep the so-called coding for prosperity rhetoric in the interest of coding for sustainability and transformation through the agency of marginalized communities. True empowerment, it seems, is in adapting coding to their lives and not vice versa. Thus, an ongoing question that should keep scholars and teachers of end-user programming awake at night should be the following: How do we help communities obtain control over coding literacy and digital technology? How might our programs than teach coding literacy practices for protection, sustainability, and survival? What role might community coding literacy programs like Clearwater Academy play in this sustainability work? These questions call up the explicit need to engage with the institutional and structural list of oppressions as a lifelong goal of teaching and using emerging technologies.

Limitations

This dissertation provides a foundation for an ongoing research agenda that explores how racially marginalized people learn and use emerging digital literacies to create characteristics of social inclusion that fits their sociomaterial conditions. For this reason, this dissertation is a first draft of findings, marked by a few limitations. In this section, I'll name two of those limitations and discuss afterward how they figure into additional chapters for a book manuscript.

Focusing on Race as a "Single-Issue": A Call for Black Feminist Perspective

This dissertation uses what Rachel E. Luft calls "a single-issue tactic." In other words, this study largely centers racial inequality and how it may influence African American adults' learned coding literacy in a computer code bootcamp, and how racial inequality shapes their use of coding literacy for social mobility. Luft makes a compelling argument for taking a "single-issue tactic" in educating others in social justice workshops. She notes that Kimberle Crenshaw's "intersectionality" necessarily captures the complexities of multiple identities; however, intersectionality may also hide the unique barriers societal structures pose against each identity. Challenges to racial identity are dissimilar, in some ways, from challenges to gender identity. The choice to present a single identity depends on context. For example, if a teacher addresses a majority white class in a workshop, beginning with one identity—race—maybe more useful than immediately jumping into intersectionality and social justice. It's relatively easy to center race in literacy studies, as white supremacy has been the most visible basis for structuring our institutions, activating a long tradition within Writing Studies to tackle what seems to be the greatest social disease facing the United States.

Having taken a single-issue tactic in this study, I see potential for a much-needed shift toward examining the dynamics among race, gender, and coding literacy in a future book manuscript. A range of research on girls and computing science education exist, especially on how to recruit more girls into computer science classes and undo cultural assumptions that they

“don’t do STEM.” A related area of research includes how adult women navigate patriarchal cultures of tech. Feminist approaches to studying gender and coding has immense value, but this heuristic can neglect the experiences of not only Black women (Crenshaw 154) but also the experiences of Black transgender and gender nonbinary people. In other words, we are met with challenges to trying to universalize encounters with coding literacy and the systems and structures that regulate their distribution across people.

The combination of race and gender presents narratives that not only are Black girls not interested in coding, they must become interested in coding to bring them into civilization. An intersectional approach reveals this and other ways ideologies shape the motivations for coding literacy education and what kinds of programming Black girls learn. More important, an intersectional lens uncovers the ways white privilege complicates our understanding of patriarchy within software development. What kinds of values or traditions are distributed and exposed through the daily interactions of Black women? And how do they draw on their own knowledge to navigate spaces of both racism and sexism? In addition to this, a Black feminist perspective points directly to how coding literacy itself operates as tool for Black womanist liberation and for fostering communities of Black women coders. Scholars then may draw a more nuanced picture of coding literacy’s use and status across race and gender. These two social constructions are implicated through education policy and workplace practices, and we cease to neglect the unique affordances given to Black women.

Short time span

This study covers roughly eighteen to twenty-two months of African American students’ experiences with a computer code bootcamp and their post-graduation. Doing a follow up three to six months (and in some cases nearly a year later) only captures in a small way the long-term uses of coding literacy. As seen in Chapter Three, the lives of participants diverged in different

directions, not all indicating a straightforward account of coding literacy's power in a racist world. Long after I declared data collection complete, I (surprisingly) met participants at random moments—on the street, on campus, and in bars and coffee shops. Although most of these encounters were brief, the conversations indicated rich details on their lives beyond the six-month check-in.

Participants whom I thought I had not obtained social mobility I later discovered did indeed improve their lives. Consider Alex, a student from the Spring 2017 cohort. After graduation, he spent the summer in jail, and then some months again in poverty, living out of his sister's truck. However, on the night of the 2018 Congressional election results, I randomly met Alex in a gay bar. During this encounter, I noticed Alex was smartly dressed in a tan peacoat and suit. He had become a partial investor in a telecommunication company and now worked as the company's communications director. His life post-graduation brought a beautiful conclusion to a life story that I had assumed ended with poverty and a truck.

Given that coding for all is still in its early stages and that computer code bootcamps seem to have become a legitimate means for learning coding (i.e. some employers will send their coders to learn new programming languages from a code bootcamp; the University of Texas at Austin now has a computer code bootcamp in their continuing education program), there's reason to follow recent calls in literacy studies for lifespan research. Couple a digital literacy life history with a longitudinal study on marginalized coders, from classroom to career and beyond. What shifts are made in their lives that either directly implicate coding literacy or suggest alternative influences of coding literacy education on participants? How are other literacies, which are reliant on computer programming in our digitally multimodal communication ecosystem, shift in light of changes to how computer programming gets done for these participants. And how might ideologies on race re-write their efforts for liberation?

An Introduction to Future Work

The rhetorical situation and time constraints of researching and writing this dissertation has inevitably left other chapters on the cutting room floor. In this final section, I want to briefly describe two additional chapters and a revision of another chapter in this dissertation. These additions and revisions would stem from analyzing field notes as well as additional observation and interviews with computer code bootcamps instructors and students, at Clearwater Academy or elsewhere.

Chapter 2 Revision

My initial intent for Chapter Two was to explore the places African American students practiced coding. I noticed that a few participants would discuss how they negotiated use of space and time to get work done outside of Clearwater Academy. For example, Zeus, a 25 year-old Afro-Latinx student, would go to a classmate's arcade and play video games. The classmate would offer Zeus advice on life and challenge Zeus to complete his coding assignments. Kevin's experiences in Chapter Two is actually a snapshot of this larger intent to possibly interview and then visit the places students go to code. This chapter contributes to interests in space, place, and literacy and highlights specifically the understudied locations where racially marginalized people practice literacy learning. What kinds of rhetorical moves must they make with people, objects, and invisible policies that govern those spaces to get work done? How might their own digital environment (video tutorials, Stack Overflow forums, the integrated developer environment, in-person and online conversations with other coders) play a role in their coding? What's the link between the physical environment and the digital screen that informs their coding labor? This chapter would seek to theorize Black digital coding spaces – the environments Black coders create to engage digital cultures.

Chapter 4

The fourth data chapter would put in conversation participants' expectations and perspectives of the course curricula with the instructors' expectations and perspectives of course curricula. While this dissertation focused on the stories of students before, during, and after attending Clearwater Academy, this chapter helps provide a broader context, as it unpacks the institution's relationships with its students and vice versa. The group interviews with participants and the interviews with the instructors will help highlight the tensions between the two groups as they describe what coding literacy and technology should be doing. Analysis, I argue, may reveal how the coding for all movement might develop a critical coding literacy approach to curriculum design and pedagogy.

Chapter 5

While the fourth data chapter develops a critical coding literacy based on the combined conversations of instructors and students, the fifth data chapter brings together conversations with the instructors and the tech companies that sponsor computer code bootcamps. On the surface, the partnership between a non-profit intending to promote equity and reduce poverty and a tech company intent on drawing a profit seems admirable. The mutual benefit relies on social mobility and visibility of marginalized people once they become paid interns or full-time employees of the tech sponsors. But this chapter looks at how exactly these two co-literacy sponsors are framing diversity and inclusion practice around coding literacy. Moreover, this chapter investigates how exactly a computer code bootcamp that centers social justice shapes the discourse about race within its tech company sponsors over time.

Works Cited

- Alexander, Michelle. *The New Jim Crow: Mass Incarceration in an Age of Colorblindness*. The New P, 2012.
- Alexis, Cydney. "The Material Culture of Writing: Objects, Habitats, and Identities in Everyday Things." *Rhetoric, Through Everyday Things*, edited by Scot Barnett and Casey Boyle. U of Alabama P, 2017, pp. 83-95.
- Anderson, James D. *The Education of Blacks in the South, 1860-1935*. University of North Carolina P, 1988.
- Banks, Adam J. *Race, Rhetoric, and Technology: Searching for Higher Ground*. NCTE, 2005.
- Bannon, Jessica L. "Capitalizing on Adult Education: The Economic Imperative for Literacy in 1960s Federal Policy Discourse." *College English*, no. 78, vol. 4, 2016, pp. 316-339.
- Bargotti, Stephen, et al. "Network Analysis in the Social Sciences." *Science*, vol. 323, no. 5916, 2009, pp. 892-895.
- Bazerman, Charles, et al. "Taking the Long View on Writing Development." *Research in the Teaching of English*, vol. 51, no. 3, 2017, pp. 351-259.
- Beckett, Katherine, and Steve Herbert. *Banished: The New Social Control in Urban America*. Oxford UP, 2010.
- Bessen, James. *Learning by Doing: The Real Connection Between Innovations, Wages, and Wealth*. Yale UP, 2015.
- Bramson, Kate. "R.I. Initiative Exposes More Students to Computer Science, But It's Got a Long Way to Go." *Providence Journal*. Gate House Media, 13 Dec 2017.
<https://www.providencejournal.com/news/20171213/ri-initiative-exposes-more-students-to-computer-science-but-its-got-long-way-to-go—video>. Accessed 8 Jan 2018.

- Brandt, Deborah. *The Rise of Writing: Redefining Mass Literacy*. Cambridge UP, 2015.
- . *Literacy in American Lives*. Cambridge UP, 2001.
- . "At Last: Losing Literacy." *Research in the Teaching of English*, vol. 39, no. 3, 2005, pp. 305-310.
- Brock, Kevin. *Rhetorical Code Studies: Discovering Argument in and around Code*. University of Michigan P., 2019.
- . "The 'Fizzbuzz' Programming Test: A Case-Based Exploration of Rhetorical Style in Code." *Rhetoric and Computation*, special issue of *Computational Culture*, 5, 2016.
- Brock, Kevin, and Chris Lindgren. "Responding to the Coding Crisis: From Code Year to Computational Literacy." *Strategic Discourse: The Politics of the (New) Literacy Crises*. Ed. Lynn C. Lewis. Computers and Computation P., 2012.
- Brooke, Robert, and Jason McIntosh. "Deep Maps: Teaching Rhetorical Engagement through Place Conscious Education." *The Locations of Composition*, edited by Christopher J. Keller and Christian R. Weisser. New York: SUNY P, 2007. 131-150.
- Brown, James Jr., and Annette Vee, editors. *Rhetoric and Computation*, special issue of *Computational Culture*, 5, 2016.
- Brown, James. Jr. *Ethical Programs: Hospitality and the Rhetorics of Software*. University of Michigan P, 2015.
- Bruckman, Amy, et al. "'Georgia Computes!': Improving the Computing Education Pipeline." *SIGCSE '09 Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, March 04 - 07, 2009, Chattanooga, TN, edited by Gary Lewandowski and Steven Wolfman. ACM: New York, 2009. 86-90.

- Bogost, Ian. "Procedural Literacy: Problem Solving with Programming, Systems, & Play." *Telemedium*, Winter/Spring 2005, pp. 32-36.
- . *Persuasive Games: The ExPive Power of Videogames*. MIT P, 2007.
- Burning Glass Technologies. "Beyond Point-and-Click: The Expanding Demand for Coding Skills." Burning Glass, 2016.
http://burning-glass.com/wp-content/uploads/Beyond_Point_Click_final.pdf
- Carey, Tamika. *Rhetorical Healing: The Reeducation of Contemporary Black Womanhood*. SUNY P, 2016.
- Castleton, Geraldine. "Workplace Literacy as a Contested Site of Educational Activity." *Journal of Adolescent & Adult Literacy*, vol. 45, no. 7, 2002, pp. 556-66.
- Castro, J.G. "Sanford Biggers: Music, Afrofuturism, and Re-envisioning History." Black Renaissance/Renaissance Noire, Institute of African American Affairs, 14 Aug. 2013, <https://www.nyubrn.org/sanford-biggers-music-afrofuturism-and-re-envisioning-history-interview-by-jan-garden-castro/>. Accessed 20 Apr. 2019.
- Clanchy, T.M. *From Memory to Written Record: Europe 1066-1307*. Wiley Blackwell, 2013, pp. 255-285.
- Coleman, Gabriella E. "Ethnographic Approaches to Digital Media." *Annual Review of Anthropology*, vol. 39, 2010, pp. 487-505.
- Cornelius, Janet Duitsman. *When I Can Read My Title Clear: Literacy, Slavery, and Religion in the Antebellum South*. University of South Carolina P, 1991.
- Cooper, Marilyn. "The Ecology of Writing." *College English* vol. 48, no. 4, no, 1986, pp. 364-375.
- Cooper, Steve, Jeff Forbes, Armando Fox, Susanne Hambrusch, Andrew Ko, and Beth Simon. "The Importance of Computing Education Research." *Computing Community*

- Consortium*, 2015, <https://cra.org/ccc/wp-content/uploads/sites/2/2015/01/CSEdResearchWhitePaper2016.pdf>
- Daniels, Jessie. “‘My Brain Database Doesn’t See Color’: Color-Blind Racism in the Technology Industry and in Theorizing the Web.” *American Behaviorist Science*, vol. 59, no. 11, 2015, pp. 1377-1393.
- . “Race and Racism in Internet Studies: A Review and Critique.” *New Media and Society*, Vol. 15, no. 5, 2013, pp. 695-719.
- Delpit, Lisa. “The Silenced Dialogue: Power and Pedagogy in Educating Other People's Children.” *Harvard Educational Review*, vol. 58, no. 3, pp. 280-298.
- . “Acquisition of Literate Discourse: Bowing before the Master?” *Theory Into Practice*, vol. 31, no. 4, 1992, pp. 297-302.
- . *Other People’s Children: Cultural Conflict in the Classroom*, W.W. Norton, 1995.
- Dery, Mark. “Wired Man’s Burden: The Incredible Whiteness of Being Digital.” *AfroGreeks: Beyond the Digital Divide*, edited by Anna Everett and Amber J. Wallace, Center for Black Studies Research, 2007.
- Dinerstein, Joel. “Technology and Its Discontents: On the Verge of the Posthuman.” *American Quarterly*, vol. 58, no.3, 2006, pp. 569-595.
- Dishman, Lydia. “Why Coding Is Still the Most Important Job Skill of the Future.” *Fast Company*, Fast Company Inc, 14 June 2016.
<https://www.fastcompany.com/3060883/why-coding-is-the-job-skill-of-the-future-for-everyone>. Accessed 31 March 2019.
- Driscoll, Megan. “Color Coded: Mendi + Keith Obadike’s Black.Net.Art Actions and the Language of Computer Networks.” *The Black Scholar*, vol. 47, no. 3, 2017, pp. 56-67.
- Drug Enforcement Administration. “Drug Scheduling.” *Drug Enforcement Administration*.

- United States Drug Enforcement Agency. n.d. <https://www.dea.gov/drug-scheduling>
Accessed 4 January 2019.
- Du Bois, W.E.B. *The Souls of Black Folk*. Bedford/St. Martin's, 1903.
- Edwards, Patricia A., Gwendolyn Thomas McMillon, and Jennifer D. Turner. *Change is Gonna Come: Transforming Literacy Education for African American Students*. New York: Teachers College P, 2010.
- Eggleston, Liz. "2018 Coding Bootcamp Alumni Outcomes and Demographics Study", Course Report, 2018.
- . "2017 Coding Bootcamp Market Size Study", Course Report, 2017.
- . "2014 Course Report Alumni Outcomes & Demographics Study." Course Report, 1 Aug. 2014, <https://www.coursereport.com/reports/2014-coding-bootcamp-outcomes-demographics-report>. Accessed July 12, 2018.
- . "2018 Coding Bootcamp Market Size Study." Course Report, Course Report, 21 Aug 2018, <https://www.coursereport.com/reports/2018-coding-bootcamp-market-size-research>, 20 Apr. 2019.
- Eubanks, Virginia. *Digital Dead End: Fighting for Social Justice in the Information Age*. MIT P, 2011.
- Everett, Anna. *Digital Diaspora: A Race for Cyberspace*. SUNY P, 2009.
- Fanon, Franz. *Black Skin/White Mask*. 2008 ed. Translated by Charles Lam Markmann, Pluto P, 1967.
- Farrell, Lesley. "Texting the Future: Work, Literacies, and Economies." *The Future of Literacy Studies*, edited by Mike Baynham and Mastin Prinsloo. Palgrave MacMillan.
- Fasching-Varner, Kenneth, et al. "Beyond School-to-Prison Pipeline and Toward an Educational

- and Penal Realism.” *Equity and Excellence in Education*, vol. 47, no. 4, 2014, pp. 410-429.
- Frey, William H. “US White Population Declines and Generation ‘Z-Plus’ is Minority White, Census Shows.” Brookings, The Brookings Institution, 22 June 2018, <https://www.brookings.edu/blog/the-avenue/2018/06/21/us-white-population-declines-and-generation-z-plus-is-minority-white-census-shows/>. Accessed 20 Apr 2019.
- Fisher, Maisha. *Black Literate Lives: Historical and Contemporary Perspectives*. Routledge P, 2009.
- Flórez, Francisco Buitrago et al. “Changing a Generation’s Way of Thinking: Teaching Computational Thinking Through Programming.” *Review of Education Research*, vol. 87, no. 4, 2017, pp. 834-860.
- Gee, James Paul. “Literacy, Discourse, and Linguistics: Introduction.” *Journal of Education*, vol. 171, no. 1, 1989, pp. 5 -17.
- . “What is Literacy?” *Journal of Education*, vol. 171, no. 1, 1989, pp. 18-25.
- Geronimus, Arline, et al. “‘Weathering’ and Age Patterns of Allostatic Load Scores Among Blacks and Whites in the United States.” *American Journal of Public Health*, vol. 96, no. 5, 2006, pp. 826-833.
- Gilyard, Keith and Adam J. Banks. *On African American Rhetoric*. Routledge, 2018.
- Goodson, Ivor and J. Marshall Mangan. “Computer Literacy as Ideology.” *British Journal of Sociology of Education*, vol.17, no. 1,1996, pp. 65-79.
- Goody, Jack and Ian Watt. “The Consequences of Literacy.” *Comparative Studies in Society and History*, vol. 5, no. 3, 1963, pp. 304-45.
- Graff, Harvey J. *The Literacy Myth: Cultural Integration and Social Structure in the Nineteenth Century*. Transaction Publishers, 1981.

- . "The Literacy Myth at Thirty." *Journal of Social History*, vol. 43, no. 3, 2010, pp. 635–61.
- Graff, Harvey, and John Duffy. "Literacy Myths." *Literacies and Language Education*, edited by Brian V. Street and Stephen May. Springer, 2018, pp. 31-42.
- Greenberger, Martin. *Computers and the World of the Future*. MIT P, 1962.
- Gregg, Melissa. "Hack for Good: Speculative Labour, App development, and the Burden of Austerity." *Apps and Affect*, a special issue of *The Fibreculture Journal*, vol. 25, 2015, pp. 183-201.
- Grenzfurthner, Johannes and Frank Apunkt Schneider. "Hacking the Spaces." n.d. Accessed November 24, 2017. <http://www.monochrom.at/hacking-the-spaces/>.
- Gowen, Sheryl Greenwood. *The Politics of Workplace Literacy: A Case Study*. New York: Teachers College, Columbia University, 1992.
- Guzdial, Mark. *Learner-Centered Design of Computing Education: Research on Computing for Everyone*. San Rafael, CA: Morgan and Claypool, 2015.
- Hannah-Jones, Nikole. "Segregation Now." *The Atlantic*. May 2014. <https://www.theatlantic.com/magazine/archive/2014/05/segregation-now/359813/>. Accessed 16 Mar 2018.
- Hanneman, Robert. A., and Mark Riddle. "Concepts and Measures for Basic Network Analysis." *The Sage Handbook of Social Network Analysis*. Ed. John Scott and Peter J. Carrington. Sage, 2011. 340–369.
- Harris, Cheryl. "Whiteness as Property." *Harvard Law Review*, vol. 106, no. 8, 1993, pp. 1710 – 1791.
- Havelock, Eric. "The Alphabetic Mind: A Gift to the Modern World." *Oral Tradition*, vol. 1, no. 1, 1986, pp. 134-150.
- Heath, Shirley Brice. "The Literate and the Literary: African American Writers and

- Readers--1830-1940." *Written Communication*, vol.11, no. 4, 1994, pp. 419-444.
- Henry, Elizabeth. *Forgotten Readers: Recovering the Lost History of African American Literary Societies*. Duke UP, 2002.
- Honey, Margaret and David E. Kanter. *Design, Make, Play: Growing the Next Generation of STEM Innovators*. Routledge, 2013.
- Hull, Glenda. "Critical Literacy at Work." *Journal of Adolescent & Adult Literacy*, vol. 43, 2000, pp. 648-652.
- Jones, Phyllis. "Confronting Institutionalized Racism." *Phylon*, vol. 50, no. ½, 2002, pp. 7-22.
- Jones, Marian. "Does Race Matter in Addressing Homelessness? A Review of the Literature." *World Medical and Health Policy*, vol. 8, no. 2, 2016, pp. 139-156.
- Kay, Alan and Adele Goldberg. "Personal Dynamic Media." *The New Media Reader*, edited by Nick Montfort and Noah Wardrip-Fruin. MIT P, 2004, pp. 393-404.
- Kirtley, Susan. "Rendering Technology Visible: The Technological Literacy Narrative." *Computers and Composition*, vol. 29, 2012, pp.191-204.
- Kitchin, Rob and Martin Dodge. *Code/Space: Software and Everyday Life*. MIT P, 2011.
- Krishnamurthi, Shiram and Kathi Fisher. "Programming Paradigms and Beyond." *The Cambridge Handbook of Computing Education Research*, edited by Sally A. Fincher and Anthony V. Robins, Cambridge UP, 2019, pp. 377-413.
- Kross, Sean and Philip Guo. "Students, Systems, and Interactions: Synthesizing the First Four Years of Learning@Scale and Charting the Future." *Proceedings of the Fifth Annual ACM Conference on Learning at Scale*, Edited by Scott Klemmer and Kenneth Koedinger London, UK, June 26 – 28, 2018, ACM: New York, n.p.
- Kynard, Carmen. "Literacy/Literacy Studies and the Still-Dominant White Center." *Literacy in Composition Studies*, vol.1, no. 1, 2013, pp. 63-5.

Lachuk, Amy Johnson. *Literacy as Moral Obligation Among African Americans in the Rural Southeast*. Lexington Books, 2016.

Ladson-Billings, Gloria. "From the Achievement Gap to the Education Debt: Understanding Achievement in U.S. Schools." *Educational Researcher*, vol. 35, no. 7, 2006, pp. 3-12.

---. "'Stakes is High': Educating New Century Students." *The Journal of Negro Education* Vol. 82, No. 2, 2013, pp.105-10.

Laquintano, Timothy and Annette Vee. "How Automated Writing Systems Affect the Circulation of Political Information Online." *Literacy in Composition Studies*, vol. 5, no. 2, 2017, pp. 43-61.

Lewis, Colleen M., Nirai Shah, and Katrina Falkner. "Equity and Diversity." *The Cambridge Handbook of Computing Education Research*, edited by Sally A. Fincher and Anthony V. Robins, Cambridge UP, 2019, pp. 481-510.

Lohr, Steve. "As Coding Bootcamps Close, the Field Faces a Reality Check." *The New York Times*, 25 August 2017, Accessed September 9, 2017.

---. "Where Non-Techies Can Get with Programming." *The New York Times*, 4 April 2017, Accessed April 4, 2019.

Lopez, German. "Marijuana is Legal for Medical Purposes in 32 States." *Vox*. Vox Media Inc. 14 November 2018. <https://www.vox.com/identities/2018/8/20/17938366/medical-marijuana-legalization-states-map>. Accessed 2 Jan. 2019.

Luft, Rachel E. "Intersectionality and the Risk of Flattening Difference: Gender and Race Logics, and the Strategic Use of Antiracist Singularity." *The Intersectional Approach: Transforming the Academy Through Race, Class, and Gender*, edited by Michele Tracy Berger and Kathleen Guidroz, University of North Carolina P, 2009, pp. 100-18.

Margolis, Jane. *Stuck in the Shallow End: Education, Race, and Computing*. The MIT

- P, 2010.
- . *Unlocking the Clubhouse: Women in Computing*. Cambridge, MA: The MIT P, 2010.
- Marin, Alexandra and Barry Wellman. "Social Network Analysis: An Introduction." *The Sage Handbook of Social Network Analysis*. Ed. John Scott and Peter J. Carrington. Sage, 2011, pp. 11-25.
- Markoff, John. *What the Dormouse Said: How the Sixties Counter-Culture Shaped the Personal Computer Industry*. Penguin Books, 2006.
- Mason, Eric. "Greening the Globe, One Map at a Time." *Community Literacy Journal*, vol.4, no.1, 2009, pp. 93-104.
- Marino, Mark C. "Reading Culture Through Code." *The Routledge Companion to Media Studies and Digital Humanities*. Jentery Sayers, editor, Routledge, 2018, pp. 472-82.
- Martin, Lori Latrice, Kenneth Fasching-Varner, and Tifanie W. Pulley. "Death by Residential Segregation and the Post-Racial Myth." *Violence Against Black Bodies: An Intersectional Analysis of How Black Lives Continue to Matter*. Ed. Sandra E. Weissinger, Dwayne A. Mack, and Elwood Watson. New York: Routledge, 2017. 90-107.
- Marx, Karl. "Ideology and Class." *Social Stratification: Class, Race, and Gender in Sociological Perspective*, 2nd edition, edited by D. Grusky. Colorado: Westview, 2001, pp. 87-105.
- McGowan, Dan. "Gov. Raimondo Wants Every RI School to Offer Computer Science Classes by '17." *WPRI Eyewitness News*. NexStar Media Group. 7 March 2016. <https://www.wpri.com/news/education/gov-raimondo-wants-every-ri-school-to-offer-computer-science-classes-by-17/1044529189>. Accessed 8 Jan 2018.
- Mignolo, Walter. *The Darker Side of the Renaissance: Literacy, Territoriality, and Colonization*.

- University of Michigan P, 1995.
- Miller, Elisabeth. "Literate Misfitting: Disability Theory and a Sociomaterial Approach to Literacy." *College English*, vol. 1, No. 1, 2016, pp. 34-56.
- Moran, Charles. "Access: The A-Word in Technologies Studies." *Passions, Pedagogies, and 21 Century Technologies*. Ed. Gail E. Hawisher and Cynthia Selfe. Logan, UT: National Council of Teachers of English. 206–20.
- National Urban League. *2018 State of Black America: Saving Our Cities: Powering the Digital Revolution*. National Urban League, 2018.
- Noble, Douglas. "Computer Literacy and Ideology." *Teachers College Record*, vol. 85, no. 4, 1984, pp. 602-14.
- Noble, Safiya. *Algorithms of Oppression: How Search Engines Reinforce Racism*. New York University P, 2018.
- Ohmann, Richard. "Literacy, Technology, and Monopoly Capital." *College English*, vol. 47, no. 7, 1985, pp. 675-89.
- Olson, David. "From Utterance to Text: The Bias of Language in Speech and Writing." *Harvard Educational Review*, vol. 47, no. 3, 1977, pp. 257-81.
- Ong, Walter. *Orality and Literacy: The Technologizing of the Word*. New York: Routledge, 1982.
- Ore, Ersula. "Whiteness as Racialized Space: Obama and the Rhetorical Constraints of Phenotypical Blackness." *Rhetorics of Whiteness: Postracial Hauntings in Popular Culture, Social Media, and Education*, edited by Tammie M. Kennedy, Joyce Irene Middleton, and Krista Ratcliffe, Southern Illinois P, 2017, pp. 256-70.
- Orr, Amy J. "Black-White Differences in Achievement: The Importance of Wealth." *American Sociological Association* vol.76, no. 4, 2003, pp. 281-304.

- Papert, Seymour. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, 1980.
- Perry, Theresa. "Freedom for Literacy and Literacy for Freedom: The African American Philosophy of Education." *Young, Gifted, and Black: Promoting High Achievement Among African American Students*, edited by Theresa Perry, Claude Steele, and Asa G. Hilliard. Beacon, 2003, pp. 11-51.
- Pew Research Center. *On Views of Race and Inequality, Blacks and Whites Are Worlds Apart*. Washington, DC: Pew Research Center, 2016. PDF. 21 Mar 2019.
- Prell, Christina. *Social Network Analysis: History, Theory, and Methodology*. Sage, 2010.
- Prendergast, Catherine. *Literacy and Racial Justice: The Politics of Learning After Brown V. Board of Education*. Southern Illinois UP, 2003.
- Pritchard, Eric Darnell. "'Like Signposts on the Road': The Function of Literacy in Constructing Black Queer Ancestors." *Literacy in Composition Studies*, vol. 2, no. 1, 2014, pp. 29-56.
- Prior, Paul and Jody Shipka. "Chronotopic Lamination: Tracing the Contours of Literate Activity." *Writing Selves/Writing Societies*. Ed. Charles Bazerman and David R. Russell. Fort Collins: The WAC Clearinghouse, pp. 180-238.
- . "Forum: Setting a Research Agenda for Lifespan Writing Development: The Long View from Where?" *Research in the Teaching of English* vol. 52, no. 2, 2017, pp. 211-19.
- Provan, Keith G. and H. Brinton Milward. "A Preliminary Theory of Interorganizational Network Effectiveness: A Comparative Study of Four Community Mental Health Systems." *Administrative Science Quarterly*, vol. 40, no. 1, 1995, pp. 1-33.
- Powell, Annette Harris. "Access(ing), Habits, Attitudes, and Engagements: Re-thinking Access as Practice." *Computers and Composition* 24, 2007, pp. 16-35.

- Reder, Stephen. *Digital Inclusion and Digital Literacy in the United States: A Portrait from PIAAC's Survey of Adult Skills*. American Institutes for Research. *U.S. PIACC Gateway*, 2015.
- Richardson, Elaine. "‘To Protect and Serve’: African American Female Literacies." *College Composition and Communication*, vol. 54, no. 4, 2002, pp. 675-704.
- Rosenberg, Scott. *Dreaming in Code*. Three Rivers P, 2008.
- Royster, Jacqueline Jones, and Gesa E. Girsch. *Feminist Rhetorical Practices: New Horizons for Rhetoric, Composition, and Literacy Studies*. Southern Illinois University P, 2012.
- Rushkoff, Douglas. "Learn to Code; Get a Job." *CNN*, Time Warner, 12 January 2012.
<https://www.cnn.com/2012/01/12/opinion/rushkoff-write-code/index.html>. Accessed 31 March 2019.
- Sample, Mark, and Annette Vee. "Introduction to ‘The Role of Computational Literacy in Computers and Writing.’" *Enculturation*, 14, 2012.
- Schafer, Gráinne et al. "Investigating the Interaction Between Schizotypy, Divergent Thinking, and Cannabis Use." *Consciousness and Cognition* 21 (2012): 292 – 298.
- Schmandt-Besserat, Denise. "The Envelopes That Bear the First Writing." *Technology and Culture*, vol. 21, no. 3, 1980, pp. 357 – 385.
- Scott, Allison, Freeda Kapor Klein, and Uriridiakoghene Onovakpuri. *Tech Leavers Study: A First-of-Its-Kind Analysis of Why People Voluntarily Left Jobs in Tech*. Kapor Center for Social Impact, August 2017, <http://www.kaporcenter.org/wp-content/uploads/2017/08/TechLeavers2017.pdf>
- Selfe, Cynthia. *Technology and Literacy in the Twenty-First Century: The Importance of Paying Attention*. Carbondale, IL: Southern Illinois UP, 1999.
- Sentencing Project. "Black Disparities in Youth Incarceration: African Americans

- 5X More Likely than Whites to Be Held.” Washington D.C.: The Sentencing Project, 2017.
- Sentry, Jentery. *Making Things and Drawing Boundaries: Experiments in the Digital Humanities*. University of Minnesota P, 2016.
- Shedd, Carla. *Unequal City: Race, Schools, and Perceptions of Injustice*. Russell Stage Foundation, 2016.
- Sheils, Merrill. “Why Johnny Can’t Write.” *Newsweek*, 8 Dec. 1975, pp. 58-64.
- Sinclair, Bruce. *Technology and the African American Experience: Needs and Opportunities for Study*. MIT P, 2004.
- Smith, Megan. “Computing Science for All.” *The White House: President Barack Obama*. whitehouse.gov. 30 Jan 2016.
<https://obamawhitehouse.archives.gov/blog/2016/01/30/computer-science-all>. Accessed 8 Jan 2018.
- Smitherman, Geneva. *Talkin’ and Testifyin’: The Language of Black America*, Wayne State UP, 1977.
- Solomon, Cynthia. *Computer Environments for Children: A Reflection on Theories of Learning and Education*. MIT P, 1986.
- Steele, Claude and Joshua Aronson. “Stereotype Threat and the Intellectual Test Performance of African Americans.” *Journal of Personality and Social Psychology*, vol. 69, no. 5, 1995, pp. 797– 811.
- Sterne, Jonathan. “The Computer Race Goes to Class: How Computers in Schools Helped Shape the Racial Topography of the Internet.” *Race in Cyberspace*, edited by Beth E. Kolko, Lisa Nakamura, and Gilbert B. Rodman. Routledge, 2000, pp.191-212.
- Stewart, Lauren. “Changing the Face of the Tech Industry.” Course Report, Course Report, 19

- Jan. 2018 <https://www.coursereport.com/blog/diversity-in-tech-why-tech-is-poised-to-see-a-change-in-2018>. Accessed 20 Apr. 2019.
- Strauss, Valerie. "A Fourth 'r' for 21st Century Literacy." *Washington Post*, January 2, 2012. https://www.washingtonpost.com/blogs/answer-sheet/post/a-fourth-r-for-21st-century-literacy/2011/12/29/gIQAx2BWP_blog.html?utm_term=.f1246c5af66d. Access 26 July 2019.
- Street, Brian. *Literacy in Theory and Practice*. Cambridge UP, 1984.
- Stuckey, Elspeth J. *The Violence of Literacy*. Boynton/Cook Heinemann, 1991.
- Sutton, Rosemary E. "Equity and Computers in the Schools: A Decade of Research." *Review of Educational Research*, vol. 61, No. 4, 1991, pp. 475-503.
- Syverson, Margaret A. *The Wealth of Reality: An Ecology of Composition*. Carbondale: Southern Illinois UP, 1999.
- Telecommunication Development Sector. *Code Bootcamps: A Strategy for Youth Employment*. International Telecommunication Union, 2016, [https://www.itu.int/en/ITU-D/Digital Inclusion/Youth-and-Children/Documents/CodingBootcamps_E.pdf](https://www.itu.int/en/ITU-D/Digital%20Inclusion/Youth-and-Children/Documents/CodingBootcamps_E.pdf).
- Thayer, Kyle and Andrew Ko. "Barriers Face by Coding Bootcamp Students." *Proceedings of the 2017 ACM Conference on International Computing Education Research*, Tacoma, WA, August 18 – 20, 2017, pp. 245-253.
- Teems, Yvonne. "The Writing Ecologies of Older American Activists." *Literacy in Composition Studies*, vol.6, no. 2, 2018, pp. 1-17.
- Treene, Alayna. "Trump to Announce Computer Science Education Initiative." *Axios*. Axios. 25 Sept. 2017. <https://www.axios.com/trump-to-announce-computer-science-education-initiative-1513305746-24c9b0c5-b6e6-4bae-9a85-425ac65e8f24.html>. Accessed 8 Jan. 2018.

- Truong, Alice. "Coding Bootcamp Grads Boost Their Salaries by 40% on Average." *Quartz* 26 October 2015. <https://qz.com/533427/coding-bootcamp-grads-boost-their-salaries-by-40-on-average-2/>. Accessed February 3, 2017.
- Ture, Kwame, and Charles V. Hamilton. *Black Power: The Politics of Liberation in America*. Vintage Books, 1992.
- U.S. Equal Employment Opportunity Commission. *Diversity in Tech*, May 2016. <https://www.eeoc.gov/eeoc/statistics/reports/hightech/upload/diversity-in-high-tech-report.pdf>
- Vara, Vauhini. "Why Doesn't Silicon Valley Hire Black Coders?" *Bloomberg*, 21 Jan. 2016, <https://www.bloomberg.com/features/2016-howard-university-coders/>. Accessed 26 July 2019.
- Vee, Annette. *Coding Literacy: How Computer Programming Is Changing Writing*. MIT P, 2017.
- Veshkin, Alison. "Silicon Valley Is High on Innovation. And Pot." *Bloomberg*. Bloomberg. 28 February 2013. <https://www.bloomberg.com/news/articles/2013-02-28/silicon-valley-is-high-on-innovation-dot-and-pot>. Accessed 4 January 2018.
- Vieira, Kate. "Undocumented in a Documentary Society: Textual Borders and Transnational Religious Literacies." *Written Communication*, vol. 28, no. 4, 2011, pp. 436 - 461.
- Vogel, Todd. *The Black P: New Literary and Historical Essays*. Rutgers UP, 2001.
- Wade, Ashleigh Greene. "'New Genres of Being Human': World Making Through Viral Blackness." *The Black Scholar* vol. 47, no. 3, 2017, pp. 33-44.

- Wang, Jennifer and Sepehr Hejazi Moghadam. "Diversity Barriers in K–12 Computer Science Education: Structural and Social." *47th ACM Technical Symposium on Computing Science Education*, March 8- 11, 2017, New York City, NY, Association for Computer Machinery, February 17, 2016.
- Warschauer, Mark and Tina Matuchniak. "New Technology and Digital Worlds: Analyzing Evidence of Equity in Access, Use, and Outcomes." *Review of Educational Research*, vol. 34, no. 1, 2010, pp. 179-225.
- Warschauer, Mark. *Technology and Social Inclusion: Rethinking the Digital Divide*. MIT P, 2004.
- Weber, Max. "Status Groups and Classes." *Social Stratification: Class, Race, and Gender in Sociological Perspective*, 2nd edition, edited by D. Grusky, Westview, 2001, pp. 132-45.
- Williams, Linda G. "Preparing for the Knowledge Society: Lessons from Detroit's Early African American Female Teachers." *African American Females: Addressing Challenges and Nurturing the Future*, edited by Eboni M. Zamani-Gallaher and Vernon C. Polite. Michigan State UP, 2013.
- Wisconsin Department of Public Instruction. "Wisconsin to Have Computer Science Standards." Madison, WI 5 Jan. 2017.
- Wysocki, Anne, and Johndan Johnson-Eilola. "Blinded by the Letter: Why Are We Using Literacy as a Metaphor for Everything Else?" *Passions, Pedagogies, and 21st Century Technologies*, edited by Gail Hawisher and Cynthia L. Selfe, NCTE, 1999, pp. 349-68.
- Young, Vershawn. "The Costs of Code-Switching." *Other People's English: Code-Meshing, Code-Switching, and African American Literacy*. Teachers College P, 2014.