Computing Mentorship in a Software Boomtown: Relationships to Adolescent Interest and Beliefs

Andrew J. Ko
The Information School
University of Washington
Seattle, WA
ajko@uw.edu

Katie Davis
The Information School
University of Washington
Seattle, WA
kdavis@uw.edu

ABSTRACT

Prior work on adolescent interest development shows that mentorship can promote interest in a subject while reshaping beliefs about the subject. To what extent do these same effects occur in computing, where interest and beliefs have traditionally been negative? We conducted two studies of the Puget Sound region in the United States, surveying and teaching 57 diverse adolescents with interests in computing. In the first study, we found that interest in computing was strongly related to having a mentoring relationship and not to gender or socioeconomic status. Teens with mentors also engaged in significantly more computing education and had more diverse beliefs about peers who engaged in computing education. The second study reinforced this finding, showing that teens who took a class from an instructor who aimed to become students' teacher-mentor had significantly greater positive changes in interest in computing than those who already had a mentor. These findings, while correlational, suggest that mentors can play a key role in promoting adolescent interest in computing.

KEYWORDS

Computing education, mentorship, interest development

ACM Reference format:
Andrew J. Ko and Katie Davis. 2017. Computing Mentorship in a Software Boomtown: Relationships to Adolescent Interest and Beliefs. In *Proceedings of ICER '17, Tacoma, WA, USA, August 18-20, 2017,* 9 pages.
DOI: 10.1145/3105726.3106177

1 INTRODUCTION

Prior work has shown that diverse adolescent populations around the world continue to view computing as boring, antisocial, irrelevant, male, and competitive [6, 8, 20, 24, 25, 35, 35–37, 44, 52]. To address this, multiple efforts worldwide aim to engage adolescents in computing education with the goal of overcoming these views, developing *interest* in computing.

Sustaining this interest, however, requires more than just initial engagement. Theories of interest development, for example, view interest as something initially triggered by events such as an in-class activity, and afterwards, as something that is maintained over time, first externally by the learning environment and then individually through intrinsic motivation, values, and identity [26]. From this view, developing adolescent interest in computing is first a matter of *triggering* interest through experiences like Hour of Code or a parent enrolling a child in a computing camp and then a matter of maintaining and developing it through coursework, projects, and community engagement.

Mentors can play a key role in both triggering and maintaining adolescent interest. They can devise learning experiences to situationally maintain interest and they can also encourage further learning, connecting learning to adolescents' identities, shifting situational interest into individual interest [26]. For example, studies of technology fluency (which have investigated coding among other technology skills) have found that parents can play mentoring roles in the development of technology skills, advancing their children's learning through parent-child collaborations [4]. These relationships are associated with adolescents' deeper expertise and positive attitudes toward technology.

Parents, of course, are only one kind of mentor. Dawson, for example, conceptualizes mentorship broadly [12], describing it as both formal relationships between younger and older individuals, but also relationships between individuals of all ages with widely varying formality and levels of engagement. Lave and Wenger treat mentorship similarly, describing apprenticeship as a form of peer mentorship that develops skill and knowledge [34].

Unfortunately, there is little prior work on informal mentoring [1]. Prior work has shown that many youth have some kind of informal mentor [5], and that these mentors can be central to youths' lives [32], but only a few works investigate computing. For example, Barron et al.'s investigation of parent-mentors only considered a few adolescents learning to code [4] and Ko's study of computing autobiographies only reported a few students mentioning mentoring relationships as part of their developed interest in computing [33]. Most prior work has instead focused on *formal* mentoring in the workplace (e.g., [28, 29]) and in educational contexts with at-risk learners (e.g., [38, 51]. This trend is also true of research on mentoring in computer science in higher education, which has found that strong communication skills and an appropriate personality fit are key [39, 42, 49], mirroring more general research on formal mentoring [23, 30]. By following these practices, CS faculty can influence enrollment decisions [45], increase retention [10, 21], and produce more effective learning [16]. Formal peer mentoring, in contrast is fraught with challenges [27], but can improve learners' sense of community [14].

ICER '17, Tacoma, WA, USA

978-1-4503-4968-0/17/08…$15.00
DOI: 10.1145/3105726.3106177

While all of this prior work *suggests* that mentorship may be an effective contributor to developing adolescents' interest in computing, there has been little systematic investigation into the relationships between computing mentoring, interest, and beliefs among youth. Specifically:

> RQ1: To what extent is mentorship related to positive beliefs about computing, interest in computing, and engagement in computing education?
> RQ2: What relationship does acquiring a computing mentor have with interest in computing?

Studying these question is complex. As Renninger and Hidi discuss in their review of conceptualizations of interest, researchers have only just begun to develop systematic, consistent conceptualizations of interest [43], providing few methods or measurements that control for interacting roles of age, identity, cognitive development, and learning context. Moreover, conducting longitudinal controlled experiments on the effects of mentorship is not only challenging, but premature, without first establishing the potential effects of computing mentorship. Therefore, our goal was to measure these potential effects and do so in a community near universal access to computing education and an abundance of potential computing mentors, partially mitigating confounding factors such as access to computing education and availability of mentors. While these methods cannot show that mentorship was a *cause* of the beliefs and interests we observed in either study, we hope that they provide insight into the potential effects of mentorship in computing education, laying a foundation for further investigation.

2 APPROACH

Our approach to investigating these two questions was to study adolescent experiences with computing mentorship in the Seattle metropolitan area in Washington state in the United States (also known as the Puget Sound region). Seattle is unique in that while it only has tens of thousands of students, it also has over 90,000 professional software engineers, possibly providing most students with access to potential computing mentors. Moreover, public and private schools in the region also provide near universal access to both formal computing education in secondary school, each offering at least one if not more computer science courses, some with dedicated CS instructors, some who primarily teach math, science, or technology, and some who are industry volunteers through programs like TEALS (Microsoft's Technology Education and Literacy in Schools program). The city also is full of *informal* learning opportunities, including adolescent coding camps designed to develop interest. Dozens of companies including Microsoft, Google, and Zillow also sponsor frequent day or week-long computing education events for teens across the region. Finally, Seattle is unique in that it is also one of the most highly educated and wealthy cities in the U.S. and yet still contains socioeconomic diversity due to the significant influx of immigrants and refugees relative to its size. These conditions made it possible to explore more idealized conditions for mentorship, mitigating structural inequities in access to learning and mentorship, while also preserving socioeconomic diversity and a degree of racial diversity and segregation. Therefore, our exploratory study is as much an investigation into the unique conditions in Seattle as it is a study of mentorship.

In this context, we conducted two studies. First, we surveyed two socioeconomically and racially diverse groups of high school students. One group of students enrolled in a 1-week, half-day web design course through a fee-based summer camp program sponsored by the University of Washington, which tends to attract upper-middle class White and Asian students from Seattle's wealthier neighborhoods and suburbs. The other group was enrolled in a federally-funded Upward Bound program, a first-generation college preparation program that recruits from three of south Seattle's public high schools, which tend to enroll racially diverse immigrant youth living in neighborhoods with low socioeconomic status. We asked students about beliefs, interests, and mentorship relationships. We then taught the Upward Bound students in a six-week course that framed the instructor as an informal computing mentor, investigating the potential for his mentorship to strengthen students' situational interest in computing.

3 STUDY 1: MENTORSHIP, BELIEFS, ENGAGEMENT, AND INTEREST

The goal of our first study was to investigate adolescents' computing mentoring relationships, their beliefs about computing, and their interest in computing.

3.1 Method

3.1.1 Population and Sampling. Our target population was adolescents aged 14-18 living in Seattle or the broader Puget Sound region. Our objective was to recruit as diverse a group of teens as possible, across race, age, gender, socioeconomic status, geography, and school. To do so, we offered two types of summer coding classes. One was a university-based Upward Bound program. Upward Bound (UB) is a federally funded college preparation program that helps high school students who are low-income and/or have no parent with a bachelor's degree enter college. There are currently 826 programs in the U.S., many of which have existed since the 1964 Economic Opportunity Act that founded them. The program we worked with serves three urban south Seattle public high schools. The program serves about 125 students per year. The program is free; students receive lunch money and a stipend to attend. In 2016, 79% were both low-income and first-generation immigrants, 50% identified as female, 35% as South Asian, 19% as African, 16% as Asian, 14% as Hispanic/Latino, 10% as Black, 4% as two or more races, and 2% as White. The program's high school graduation rate is 98% and its college graduation rate is consistently above 60%, with many alumni pursuing graduate studies. Therefore, the program primarily serves the high-achieving end of Seattle's lower income schools.

The UB program offers a full summer curriculum that includes afternoon electives. We offered a course titled "Web Design" with an enrollment limit of 25. The course description made no mention of "coding." Students could choose between it and electives in ballroom dancing, swimming, or music. Program administrators solicited student preferences and then randomly assigned students to their 1st and 2nd choices, ensuring a balance along racial and gender lines; however, administrators also encouraged students to focus on classes that would satisfy graduation requirements, and so most students who enrolled in the class had an existing interest

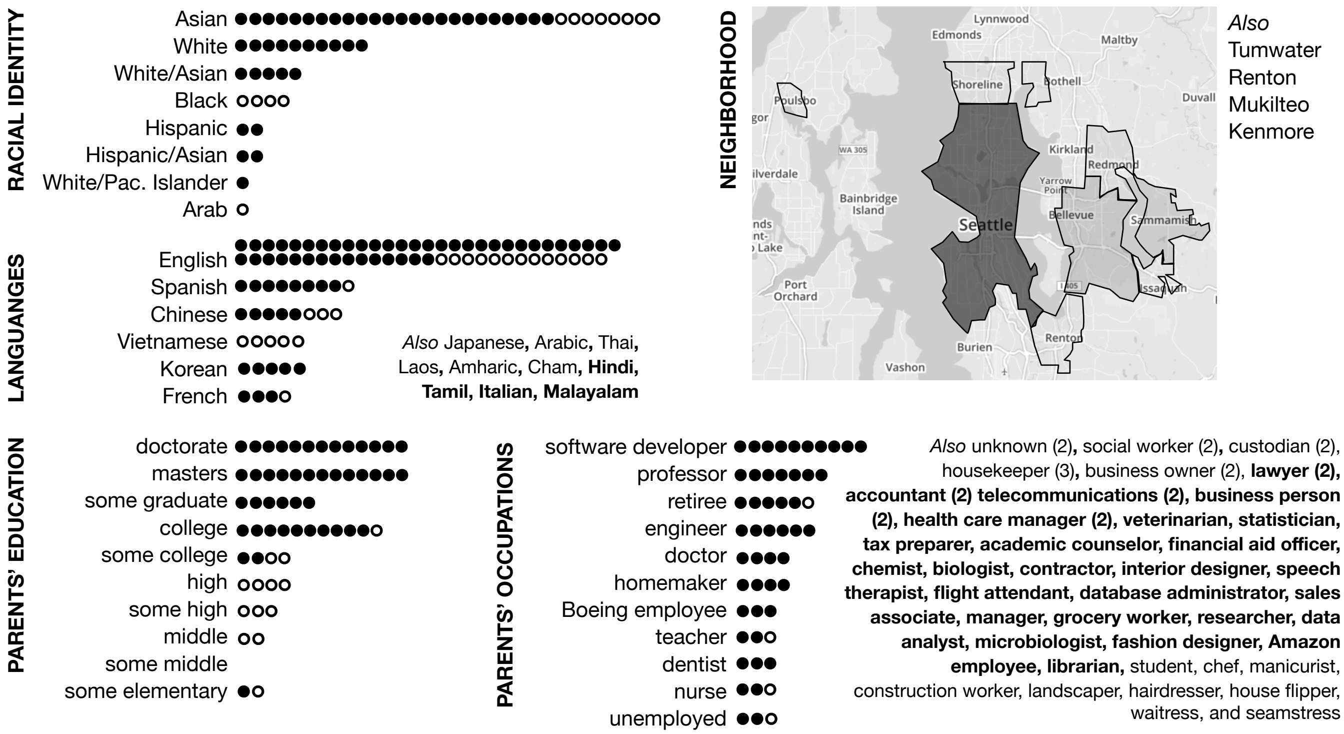

Figure 1: Student demographics. Students could choose multiple races and languages, one parent education level, one city, and up to two guardian occupations. Dark circles and bold text are SY students, white circles and non-bold text are UB students.

in computing. The UB class was 6 weeks, 4 days per week, and 36 hours total. We describe the class in detail later.

Our 2nd class was two sections of a 5-day, 3-hours per day coding camp (also titled "Web Design" and using the same UB course description) offered through a university-based summer youth (SY) program. This program markets to teachers in Seattle's wealthier north suburbs and the eastern suburbs of Bellevue, Redmond, and Kirkland. These classes tend to attract youth from upper-middle class families. Teachers distribute camp information to parents, who enroll their teens. The course had a registration fee of $275 for the week, providing a barrier to enrollment for lower-income students (though the program had a limited budget for financial assistance). Our class's fee was comparable to other half-day camps in the city. We offered two sections: one in the morning and one in the afternoon, with an enrollment limit of 25 each.

3.1.2 Participants. We successfully enrolled 57 teens across both classes. We filled both SY camps, but had several last minute cancellations, resulting in 44 total enrolled students. Our UB class had room for 25, but we only enrolled 13, two of whom dropped the class in favor of ballroom dance, but still filled out our survey.

Despite the enrollment challenges, our sample was still diverse along many dimensions. Students were 39% female and aged 14-17 (median 15). As shown in Figure 1, students were all fluent in English, but most were bilingual (12 reported never speaking English at home), most identified as White or Asian, many had highly educated parents but across a diversity of occupations, and students came from across the region, but primarily Seattle.

As with any sample, there were several biases. All students reported wanting to go to college, despite Seattle Public's 2015 high school dropout rate of 22%. Over 75% of students had a parent with a bachelor's degree, which is higher than the city's 2015 count of 54% of residents over age 25 with a bachelor's degree. (Seattle has one of the highest education rates of any U.S. city, and so the region already skewed toward high educational attainment.) Seattle's racial demographics are also unique, with many low-income recent Asian immigrants, and many affluent Hispanic families. The resulting sample is therefore reflective of the city's mix of educated professional families and its recent immigrant families.

3.1.3 Data Collection. We began all classes by having students fill out a web-based survey. Both classes were held in adjacent computer labs on a university campus, roughly identical in layout and identical in hardware. Students took approximately 30 minutes to complete the survey. It began with demographic information (age in years, neighborhood they lived in, school and grade they would enter in the fall). Next, to obtain data on students' interests broadly, we asked students about their academic plans, asking whether they planned to go to college and measuring students' possible selves [41], responding to the prompt *"Describe your vision of your life at the age of 25, assuming everything goes well. Who are you? What will you have achieved? What will your goals be?"* This question allowed us to understand students' interests, identities, and motivations. The survey then probed for interest in computing (detailed shortly), followed by beliefs about peers engaged in computing education and about software developers.

The end of the survey probed identity (to avoid potential stereotype threats [47], as questions about identity can influence how respondents describe their interests, prior knowledge, and ability). We asked for students' languages that they speak fluently, ordered from most to least fluent. We also asked them to note which language they spoke most at home. We then asked for racial identity (following the latest U.S. census recommendations found at *census.gov/topics/population/race/about.html*), gender, and socioeconomic status. Measuring socioeconomic status (SES), which is viewed as a combination of education, income, and occupation, is non-trivial [19]. However, there is evidence that parents' educational attainment is highly predictive of youth SES in adulthood [17]. Therefore, following the latest best practices on SES measurement [31], we asked students *"How many years of school has your most educated parent or guardian completed (0-25 years)? (0=no school, 5=finished elementary school, 8=finished middle school, 12=finished high school, 16=finished college, more than 16 means graduate school)."* We also asked students to identify their parent and/or guardians' occupations to help us understand their parents' occupational proximity to the software industry.

3.2 Results

3.2.1 Access to Computing Mentors. We first consider the degree to which students reported access to potential computing mentors. Of all students, 49 (89%) reported knowing someone who knew "how to code." This did not vary by SES ($F(1,55)$=.86, p=.36), gender (*Fisher's*, p=.72), or UB/SY class (*Fisher's*, p=.37). Students reported their relationships with these individuals as friends (17), fathers (16), teachers (9), brothers (4), uncles (3), cousins (2), a mentor, a neighbor, a friend's sibling, a parent's friend, a sister, and a tutor.

To analyze the nature of students' relationships with these individuals, we analyzed the students' descriptions for qualities of mentoring. Although the literature on mentorship has not yet agreed upon a theoretical account of mentoring, there are four facets that consistently arise in the mentoring literature [11]: 1) psychological and emotional support, 2) support for goal setting and career selection, 3) teaching of academic subject knowledge, and 4) framing of the mentor as a role model. We liberally operationalized computing mentoring relationships as any relationship that a student described that included one or more of these facets, including descriptions of explicit encouragement, learning activities, including explicit instruction, help enrolling in classes, visits to workplaces, role modeling and the fostering of other mentoring relationships. We treated all other relationships (including those with individuals who knew how to code) as non-mentoring relationships (for example, many described fathers, friends, and brothers who never explained their software development jobs or attempted to teach them anything about computing). Based on this definition, 24 (42%) of the students described having at least one computing mentoring relationship. Whether a student had a mentor did not differ by SES ($F(1,55)$=.87, p=.36), gender (*Fisher's*, p=.17), or UB/SY class (*Fisher's*, p=.2).

Students with mentors described a wide range of mentoring relationships:

"My dad is a professor in biology here at the university and he has been teaching me python recently." (SY, White male, 16)

"One of my best friends is very interested in programming, and has taught me some basics about HTML5" (SY, White male, 14)

"My dads friend sister works in Microsoft and he's a developer we only got to talk about her job a little bit, she took me on a tour of the campus and saw what people do there. it was pretty cool" (UB, Black female, 17)

"In the past few summers, I had a private computer programming therapist... She taught me new programs and new coding. She didn't do it this year because now she's going to college." (SY, White/Asian male, 14)

"My mom is a website developer, she teaches me about the tools she uses. My dad, writes code in java. My friends taught me how to program also" (SY, White male, 17)

"My cousin wants to be a computer science major in college and runs a club at her school called "Girls Who Code." She encourages me to try coding. " (SY, White/Asian male, 16)

"My dad is a software engineer and he frequently talks to me about his job. He has enrolled me in several classes and in our free time, he often teaches me. " (SY, Asian female, 14)

"My neighbor Laura who did APCS and really enjoyed it and introduced me to it." (SY, Hispanic female, 15)

3.2.2 Beliefs about Computing Education and Software Developers. Next, we turn to the beliefs that students reported about computing and computing education. We first asked students, *What kinds of students take your school's technology courses?* In their responses we saw few of the beliefs reported in prior work (e.g., [3, 36]), in which adolescents attributed student interest in computing to gender or intelligence. Organizing their free responses by the specific words they used, we found that students described students who engaged in computing education as "anyone" (17), "people who love computers" (9), "overachievers" (6), "people who want good jobs" (5), "people who are required to" (5), "slackers" (3), "Asians" (2), "boys," "girls," "cool people," "game lovers," and "loners." Eight students said they were unsure about who takes CS because they were new to their school.

When we considered the beliefs reported by students with and without mentors, there were only minor differences. Of students without mentors, 33% described students who took computing courses as "anyone" and 18% were unsure, whereas of the students *with* mentors, only 25% reported "anyone" and 8% were unsure. However, the students with mentors had more diverse beliefs that differed from those without mentors, including "boys," "girls," and "cool people."

In addition to beliefs about their peers who engaged in computing education, we also considered students' beliefs about professional software developers, asking students *"What characteristics do you think someone must have to be a software developer?"* Adjectives used by more than 10% of students *without* mentors included "creative," "patient," "smart," "hard-working," "intelligent", and "perseverant," all mentioned by more than 10% of students. Students *with* mentors used largely the same language, but unlike students without mentors, also used the adjectives "logical", "collaborative," "enthusiastic," and "precise."

We also asked students *"What do you think software developers do at work?"* Most students, regardless of whether they had a computing mentor, believed that developers "code all day," "go to a few meetings," and "think of product ideas." Most students did not describe the collaborative aspects of the job, portraying a typical day as a solitary one. We also asked students *"What must someone do to become a software developer?"* Responses universally mentioned taking courses and practicing extensively, as in this representative response: *"They need to learn about coding and practice it until it becomes as easy as breathing"* (UB, Vietnamese male, 15).

3.2.3 Engagement in Computing Education. Next we consider students' engagement in computing education and its relationship to students' access to computing mentors. We asked students to list all of the technology courses that their school offers, what kinds of students take those technology courses, and whether they had taken those courses. We also asked if they had used other online learning technologies to learn to code.

Of the 57 students, 50 (88%) could name at least one computing course in their school. There were no significant relationships between the number of courses known and SES (r(55)=.07, p=.60), gender (F(1,55)=2.5, p=.12), UB/SY class (F(1,55)=.84, p=.36), or whether a student had a computing mentor (F(1,55)=0.24, p=.63). We confirmed that the schools of the 7 students who could not name a computing class *did* offer one in the year prior; they were just not aware of it because they were incoming freshmen or had recently moved. Of these 7 students, 6 did not report having a computing mentor.

We asked students what CS courses or learning experiences they had engaged in. Twenty-seven (47%) reported having already engaged in some kind of computing education. Of these students, 89% (24 students) had taken an elective class at school. Who had taken a course did not differ by SES (F(1,55)=1.1, p=.29), gender (*Fisher's*, p=.10), or UB/SY class (*Fisher's*, p=1.0). Courses taken *did* differ significantly by whether a student had a computing mentor (F(1,55)=4.44, p=0.04); 16 students (67%) with a mentor had taken at least one class (and only 3 of these reported that their sole mentor was their teacher), compared to 27% of students without a mentor.

Only 4 students mentioned engaging in informal learning. One read a Python book over the summer; one started but did not complete a Codecademy Java tutorial; one started but did not complete a Codecademy JavaScript tutorial; and one had completed a series of Udacity online classes and Codecademy tutorials. Three of these four students reported having a computing mentor.

Despite only half of the students mentioning that they had engaged in computing education, when we asked students to list the programming languages that they had encountered, 42 (74%) mentioned having encountered at least one. Students mentioned familiarity with Scratch (28), HTML (19), Java (18), JavaScript (12), Python (12), CSS (8), Excel (8), C/C++ (6), Minecraft mods (7), Kodu (4), C# (2), Processing, Alice, Ruby, Go, Swift, Lua, and PHP. There were no trends in who had encountered a language by SES (F(1,55)=1.2, p=.28), gender (*Fisher's*, p=.22), or UB/SY class (*Fisher's*, p=.29). However, students with computing mentors reported encountering significantly more programming languages (F(1,55)=11.4, p=0.001), with 92% of students with a mentor encountering at least one language, compared to only 40% of students without a mentor.

3.2.4 Interest. We now turn to students' interest in computing. According to our adopted theoretical framework [26], students' engagement in computing education and access to mentors would have triggered and maintained interest in computing. We would therefore expect that there would be a strong relationship between mentorship, engagement in computing education, and interest. Prior work would also suggest a relationship between gender and cultural factors, mediated by beliefs and access [36, 37].

To investigate interest, we first analyzed the *possible selves* [41] that students described, inspecting them for interests. Although 13 students wrote that they were unsure about their careers and 15 mentioned computing careers (software developer (8), game developer (5), and web developer (2)), the other 30 described diverse interests: including doctor (6), writer (2), mechanical engineer (2), aeronautical engineer, autism activist, bioengineer, businessman, computer engineer, drug researcher, Ethiopian politician, illustrator, and several other distinct professions. This variety suggests that most of the students did not enroll in our courses because they were explicitly interested in computing as a career.

To measure students' interest in computing, we adapted the scale used by Oh et al. [40], presenting the following five items on a 7-point Likert scale: 1) *I'm interested in taking courses that help me learn to code*, 2) *I am interested in careers that allow me to use coding skills*, 3) *I would like to learn to code because it will help me prepare for college*, 4) *I would like to learn to code because it will help me get a good job*, and 5) *I would like to learn to code because it will help me create new technologies*. We mapped each response to a -3 to 3 scale and computed the mean of the five items.

Interest in computing skewed positive. The mean response was 1.1 and ranged from -2 to 3, suggesting most teens viewed computing as an interesting, valuable subject to learn. Only 10 students (18%) had attitudes below 0 on the scale, including 5 of 11 (45%) of the UB students and 5 of 46 (12%) of the SY students. This was consistent with the reasons that students listed for enrolling, which included: "I'm interested" (28), "parents wanted me to" (9), "seemed useful" (7), "curious about the topic" (3), "friends encouraged me" (2), "bored this summer" (2), "placed by counselor" (2), "retake" (2), "avoiding physical activity," and "for credit."

To test the relationship between mentorship and interest in computing, we built a linear regression. For predictors, we included gender, since prior work has observed significant disinterest from girls because of the culture of computing education (e.g., [37]). We included number of programming languages encountered as an indicator of prior learning about computing. We included SES, as prior work has shown that higher SES is related to higher academic ambitions [46]. And finally, we included whether the student reported a mentoring relationship, given its clear relationship to interest reported in prior work [4, 26] and many students had described it as a significant factor in their interest. The interest scale and the predictor variables satisfied the assumptions for a linear regression.

Table 1 shows our resulting model, which explained a significant proportion of the variance in interest (R^2=.223, F(4,52)=3.72, p=.01).

Table 1: Linear regression predicting interest. *=p<.05.

	B	SE B	β
Gender	.234	.316	.098
SES	.097	.070	.177
# PL encountered	.131	.071	.254
Had mentor	.623	.319	.265*

Gender, SES, and programming languages encountered did not explain a significant proportion of variance in interest, but having a computing mentor did, and was responsible for a 0.623 increase on the -3 to 3 computing interest scale.

4 STUDY 2: FORMING COMPUTING MENTORING RELATIONSHIPS

The results of Study 1 suggested that the most significant factor related to interest in computing was mentoring. However, this study did not allow us to observe mentoring relationships being formed, or the possible effects of acquiring a mentor on change in interest. Therefore, in Study 2 we taught the 6-week UB class with the explicit goal of the instructor (the 1st author) to develop informal computing mentoring relationships with each of the 11 students. To do this, the instructor aimed to give the students an accurate portrayal of the web development communities of practice [34], while developing a sincere interest in each individual student's learning, leveraging his decade of experience as a teacher and 5 years of experience as a professional web developer.

4.1 Course Design

Throughout the course, we attempted to follow best practices from education and computing education. The class was only 11 students, reducing the negative effects of large class sizes (e.g., [2]). The instructor followed the best practices of classroom management [18]. For example, because the class was in a computer lab, he managed student attention by creating separate spaces for lecture and computer-based learning to prevent students from being distracted. He established and enforced clear rules of conduct, preventing disruptions. He also followed the evidence-based practice of learner-centered teacher-student relationships [9], attempting to create authentic relationships in which students were trusted, given responsibility, spoken to honestly and warmly and treated with dignity. This approach was consistent with the tone of instruction throughout the rest of the UB program. In addition, the class followed NCWIT's inclusive learning practices (*ncwit.org/resources/type/promising-practices*), explicitly discussing stereotype threat, imposter syndrome, and theories of intelligence in the context of the software industry, computing education, and in the classroom. The instructor also followed NCWIT's practice of intentional role modeling (based on studies such as [48]), 1) explaining what made his role as a practitioner relevant to their learning, 2) describing his personal history and how it related to the students' experiences, 3) speaking about his strengths and weaknesses and how they related to his expertise, and 4) showing them how he attained his position.

Our instruction was also informed by communities of practice theory [50], as prior work has explored in computing education [13, 22]. In this theory, a newcomer's purpose is to learn to talk and do as the community does, acquiring the norms, practices, skills, and tools that the community uses to do its work. In our class, we followed the practices described in prior work (e.g., [7, 22]), having students use authentic web development tools of GitHub, JSFiddle, and Bootstrap, but scaffolding them to facilitate learning. We introduced the students to three experienced web developers who had worked at local companies. We showed the students the diversity of software companies in the Puget Sound region, teaching them about the dozens of large companies and their different corporate cultures and values, but also the hundreds of software startups. Across four reading assignments, students reflected on equity in computing education, reading the White House press release on the CS for All initiative, Chapter 4 of Stuck in the Shallow End [36] (which covers preparatory privilege), a viral blog post by a female software engineer on imposter syndrome in the software industry, and a case study written by the 1st author about his experiences at a software startup.

We designed a 3-week project in which the students designed and developed a website individually or in teams, creating something personally meaningful. The class involved 6 projects across the 11 students. They chose diverse topics, including an informational site about Ethiopian culture, practical applications of the philosophy of ethics in everyday high school life, a youth book recommendation site, a site to help high school students reduce stress and avoid procrastination, a site for sharing trends in athletic shoes, a site for sharing life hacks that make high school easier, and a site for surviving junior-year humanities courses. Throughout 3 weeks of web development, the instructor provided individual help, offering constructive, guiding feedback about each student's efforts. He also required students to provide weekly peer evaluation about their teammates' collaboration skills.

The instructor taught students about pathways to joining web development communities of practice, giving detailed information about what kind of education was required, what types of jobs and companies hire web developers, and what kinds of people currently pursue these pathways, including his own path. The web developers who visited the class also talked about the pathways that they took, what they regretted about their path, and what surprised them as they followed their path.

In the context of the pedagogical strategies and content described above, the instructors' mentorship formation strategies included the following: 1) talking each day to each student about their progress on learning, 2) in these conversations, explicitly linking their progress to the goals they reported in their possible selves, and 3) at the end of the course, having an explicit mentoring conversation with each student individually, offering to help them with their college applications, connect them with further computing education experiences, and answer their questions over email as they approached college.

Finally, at the end of the course, the instructor gave students the survey from Study 1, excluding redundant demographic questions.

Table 2: An ordinal regression predicting change in interest in computing. *=p<.05.

	B	SE B	Wald	β
Interest before class	-0.46	0.25	3.49	.06
Male	4.05	2.27	3.18	.07
SES	-0.21	0.58	0.13	.71
# PL encountered	-0.67	0.67	0.99	.32
No mentor	5.79	2.82	4.21	.04*
Midterm	-0.43	0.19	4.97	.16

4.2 Results

UB student interest in computing changed significantly, moving from a mean of 0.5 to 1.35 (t(10)=2.9, p=.016). None of the students began at the top of the interest scale and all but one had an increase toward neutral or positive interest.

Students' descriptions of their possible selves at the end of the class revealed some reasons for these changes. Six of the 11 students were still sure they wanted to be doctors, lawyers, software developers, and pharmacists. However, 5 of the 11 students described substantially different possible selves, incorporating computing into their identities. One student who was particularly engaged in class went from being generically interested in medicine to wanting to strengthen her coding and drawing skills, both of which she explored in the design and implementation of her team's website. One went from wanting to be a doctor to wanting to first study computer science to make money, then use the money to go to medical school to study ophthalmology. Another student who was particularly adept at learning JavaScript went from describing his future as *"a blank canvas"* to wanting to study computer science and aviation, because *"perhaps those fields need someone to write programs to analyze data and make better equipment"* (UB, Asian male, age 15). Yet another student came into the class with strong web development skills from a prior course but low self-efficacy due to a bad teamwork experience. After working independently on his project, he left the class confident in his skills and said he was now considering computer science as a major.

To investigate what was related to these changes, we built a regression similar to that in Study 1, but this time predicting *change* in interest from several factors. We included the same factors from before: number of programming languages encountered before the class as an indicator of prior knowledge, gender, SES, and whether a student had a mentor. However, we also included interest in computing prior to the class (as this would likely contribute to interest after the class) and the students' scores on their final exam (as success in CS classes can greatly influence interest [33]). Change in interest violated normality assumptions for a linear regression, and so we built an ordinal regression model instead, computing the difference in the pre and post sum of Likert items.

Table 2 shows the resulting model. The model explained a significant amount of the variance in change in post-class computing interest (χ^2=21.8, p=.001, Cox and Snell R^2=.86). In this model, not having a mentor before the class was significantly related to positive changes in interest in computing.

When we asked the students to explain how the class influenced their interest in computing, if at all, their explanations included many references to the instructor, and particularly to the inclusive learning environment:

> *"This class helped me understand what jobs involved with tech would look like and how it's not just a lonely person in the basement." (UB, Asian male, 15)*

> *"The readings from this summer encouraged me to acquire technology skills since President Obama plans to enforce Computer Science classes and it seemed to be possible for anyone to learn once they get past imposter syndrome like that one woman in the reading." (UB, Asian male, 16)*

> *"This class has greatly influenced my attitude towards coding. It allowed me to look past the stereotype that I never realized." (UB, Asian male, 16)*

> *"You definitely changed what I thought about computer science and web design. I love designing things, but I never thought one day I could design my own website in 6 weeks. The environment and class made class more enjoyable for me. This environment made me want to be part of it. Unlike classes where you feel like you aren't needed. Lastly, you taught me the thought and idea of what technology skills can do for you in the future even if it's in the medical field. That makes me want to continue learning." (UB, Asian female, 16)*

Students' reflections on the reading assignments revealed similar shifts in interest. For example, one of the readings was a blog post by a woman who started as a web developer and faced imposter syndrome, but eventually realized that failure was part of every developer's job. Two girls were surprised:

> *"Learning to code is intimidating because it's like learning a foreign language...When I feel frustrated, I start to feel like I don't belong in the class. I start to question whether I should pursue a career where I would never be able to master a subject. Technology is always changing, and I have to ask myself if I would feel comfortable learning something new every day for the rest of my life. I now know it's normal to feel uncomfortable when you are learning something new and that is why I believe I will have the perseverance to learn as much as I can about web design." (UB, Asian female, 15)*

> *"When I first took web design, I thought the guys would be experts in this. Because I know guys who knew a lot about computer [sic] and how it works. So in the beginning I was a bit intimidated until I learned that all of us know nothing about web design. It made me less scared when I want to ask a question about web design." (UB, Asian female, 16)*

Three of the students that did not have mentors prior continued to engage the instructor as a mentor after the class ended. One sought advice on computing-related summer jobs and at the time of this writing is participating in a Girls Who Code camp as an instructor upon the instructor's recommendation. Another sought advice on how to engage in computing research as an incoming university freshman and is now participating in a first-generation college student research mentorship program. A third sought advice on what to study to combine computing and medicine.

5 DISCUSSION

Our results contribute the following discoveries about Puget Sound teens with interests in computing:

- Teens can have mentoring relationships with a range of people, including friends, parents, siblings, cousins, teachers, and even neighbors.
- Teens' beliefs about peers who engage in computing education were mostly positive, describing high-achieving youth of any gender, race, or ethnicity.
- Teens' beliefs about software developers described developers as creative, patient, intelligent, hard-working, and perseverant people.
- Beliefs held by those with mentors were more diverse and had greater depth than those without mentors.
- While most teens knew about computing education opportunities regardless of whether they had a mentor, having a mentor was related to engaging in more of them and encountering more programming languages.
- While most teens expressed interest in computing as a skill, subject, and career, teens with mentors reported stronger interest regardless of gender or socioeconomic status.
- Teens that reported the first author as a teacher-mentor reported significantly higher changes in interest in computing than students that already had mentors, regardless of gender, socioeconomic status, performance in class, prior knowledge, or prior interest.

There are many ways to interpret these results. First and foremost, none of these results are causal. It is possible, for example, that students who developed interests in computing did so without mentors (for example, through Code.org's Hour of Code or some other compulsory learning setting) and then sought mentors in their social networks. It is also possible that interest was entirely mentorship driven, with mentors proactively triggering the students' interest in computing and then maintaining it through subsequent activities. Students' self-reports about their mentoring relationships suggest the latter was more likely than the former, as most students with mentors described fathers and siblings proactively introducing them to computing.

Similarly, our efforts to *form* a mentoring relationships with the UB students may have not been the *cause* of students' increased interest in computing. Students may have had other experiences in the summer that developed interest in computing, such as the college prep class that encouraged the UB students to develop their career interests. It may have been the *combined* effect of mentorship and career interest reflection that produced the increases in interest. The ability to form a mentoring relationship with the Asian students may have also been mediated by the somewhat Asian appearance of the instructor. That said, prior work on role models suggests that identity matters less in recruiting than in retention [15].

Our study has limited generalizability, first and foremost because of the sampling biases in our data. At the time of our study, there were tens of thousands of high school students in the Puget Sound metropolitan area and we only learned about 57 of them. Our claims are limited to the types of students who enroll in UB (low income, first-generation college students), and the types of students who enroll in the SY camps that we offered (students with highly educated parents interested in their teens learning to code). We did not study any low-income students who were not college bound, nor did we study middle-class students who did not have sufficient prior interests (or parents with sufficient interest) to enroll in our web design courses. There were also likely structural inequities that prevented the full population of students of being aware of our classes, limiting participation from teens in lower SES, lower educational attainment families.

There is also little evidence to claim generality of our results to other regions in the world. For example, even regions that share some features to the Puget Sound such as Silicon Valley have other features that could have changed our findings: it is more diverse racially and socioeconomically than Seattle, not to mention an order of magnitude larger in population. Moreover, it is unclear how our results would generalize to regions that do not have vibrant software industries or universal access to computing education.

Our measurements relied on self-report surveys and scales that can be sensitive to the timing of self-report, introducing some threats to construct validity. Moreover, while most of the measurements in the surveys were taken from validated scales, not all of them were, and so some of the data may not accurately reflect the phenomena we intended to study. Moreover, we used regressions to investigate predictors of interest, which only suggest correlation.

In light of these multiple interpretations and limitations, the need for future work is extensive. Studies should investigate the causal effects of mentorship, for example, through formal mentorship programs, exploring the potential for virtuous cycles of mentorship-triggered interest and learner-driven interest maintenance. Future studies should investigate the granular effects of specific forms of mentorship, such as the explicit instruction by parents and siblings that students reported. Studies should also investigate the specific effects of mentorship on other constructs such as changes in identity and shifting of beliefs, especially in settings with the more negative beliefs we did not observe.

In addition to developing a better theoretical account of computing mentorship, there are also numerous design questions about how to structure mentorship in computing. For instance, are adults, siblings, friends, or teachers better or worse positioned to mentor adolescents about computing? Are teacher-mentors scalable, given typical class sizes in public schools? What kinds of skills do mentors need to be credible and relevant to teens? Would recent high school graduates pursuing computing be effective mentors? Is remote mentorship feasible, especially in light of the majority of regions in the world having substantially fewer mentors than regions like the Puget Sound? These alternatives suggest that increasing interest in computing, even in the presence of universal access to learning opportunities, may require significant investments from the computing community to increase interest and engagement in computing education. Recent efforts like *NextBillion.org*, which helps connects people with disabilities to industry professionals is one example of what such efforts might entail.

ACKNOWLEDGMENTS

This material is based upon work supported by Microsoft, Google, Adobe, and the National Science Foundation under Grant No. 1539179, 1314399, 1240786, and 1153625.

REFERENCES

[1] Tammy D. Allen and Lillian T. Eby. 2011. *The Blackwell handbook of mentoring: A multiple perspectives approach.* John Wiley & Sons, USA.

[2] Joshua D Angrist and Victor Lavy. 1999. Using Maimonides' rule to estimate the effect of class size on scholastic achievement. *The Quarterly Journal of Economics* 114, 2 (1999), 533–575.

[3] Louise Archer and Hiromi Yamashita. 2003. 'Knowing their limits'? Identities, inequalities and inner city school leavers' post-16 aspirations. *Journal of Education Policy* 18, 1 (2003), 53–69.

[4] Brigid Barron, Caitlin Kennedy Martin, Lori Takeuchi, and Rachel Fithian. 2009. Parents as learning partners in the development of technological fluency. *International Journal of Learning and Media* 1, 2 (2009), 55–77.

[5] Margaret R Beam, Chuansheng Chen, and Ellen Greenberger. 2002. The nature of adolescents' relationships with their "very important" non-parental adults. *American Journal of Community Psychology* 30, 2 (2002), 305–325.

[6] Lori Carter. 2006. Why students with an apparent aptitude for computer science don't choose to major in computer science. *SIGCSE Bulletin* 38, 1 (2006), 27–31.

[7] Gwo-Dong Chen, Liang-Yi Li, and Chin-Yea Wang. 2012. A Community of Practice Approach to Learning Programming. *Turkish Online Journal of Educational Technology-TOJET* 11, 2 (2012), 15–26.

[8] Sapna Cheryan, Benjamin J Drury, and Marissa Vichayapai. 2013. Enduring influence of stereotypical computer science role models on womenfis academic aspirations. *Psychology of Women Quarterly* 37, 1 (2013), 72–79.

[9] Jeffrey Cornelius-White. 2007. Learner-centered teacher-student relationships are effective: A meta-analysis. *Review of educational research* 77, 1 (2007), 113–143.

[10] A Craig. 1998. Peer mentoring female computing students–does it make a difference?. In *Proceedings of the 3rd Australasian conference on Computer science education.* ACM, USA, 41–47.

[11] Gloria Crisp and Irene Cruz. 2009. Mentoring college students: A critical review of the literature between 1990 and 2007. *Research in higher education* 50, 6 (2009), 525–545.

[12] Phillip Dawson. 2014. Beyond a definition: Toward a framework for designing and specifying mentoring models. *Educational Researcher* 43, 3 (2014), 137–145.

[13] Betsy DiSalvo, Mark Guzdial, Amy Bruckman, and Tom McKlin. 2014. Saving face while geeking out: Video game testing as a justification for learning computer science. *Journal of the Learning Sciences* 23, 3 (2014), 272–315.

[14] Peggy Doerschuk. 2004. A research and mentoring program for undergraduate women in computer science. In *Frontiers in Education.* IEEE, USA, S2H–7–12.

[15] Benjamin J Drury, John Oliver Siy, and Sapna Cheryan. 2011. When do female role models benefit women? The importance of differentiating recruitment from retention in STEM. *Psychological Inquiry* 22, 4 (2011), 265–269.

[16] Daryl D'Souza, Margaret Hamilton, James Harland, Peter Muir, Charles Thevathayan, and Cecily Walker. 2008. Transforming learning of programming: a mentoring project. In *Conference on Australasian Computing Education.* Australian Computer Society, Inc., Australia, 75–84.

[17] Eric F Dubow, Paul Boxer, and L Rowell Huesmann. 2009. Long-term effects of parentsfi education on childrenfis educational and occupational success: Mediation by family interactions, child aggression, and teenage aspirations. *Merrill-Palmer quarterly (Wayne State University. Press)* 55, 3 (2009), 224.

[18] Carolyn M Evertson and Carol S Weinstein. 2013. *Handbook of classroom management: Research, practice, and contemporary issues.* Routledge, England.

[19] Arline T Geronimus, John Bound, and Lisa J Neidert. 1996. On the validity of using census geocode characteristics to proxy individual socioeconomic characteristics. *J. Amer. Statist. Assoc.* 91, 434 (1996), 529–537.

[20] Tony Greening. 1998. Computer science: through the eyes of potential students. In *Australasian conference on Computer science education.* ACM, New York, 145–154.

[21] Denise Gürer and Tracy Camp. 2002. An ACM-W literature review on women in computing. *ACM SIGCSE Bulletin* 34, 2 (2002), 121–127.

[22] Mark Guzdial and Allison Elliott Tew. 2006. Imagineering inauthentic legitimate peripheral participation: an instructional design approach for motivating computing education. In *ACM International Workshop on Computing Education Research.* ACM, New York, 51–58.

[23] John C. Hall. 2003. *Mentoring and Young People: A Literature Review.* ERIC, USA.

[24] Michael Hewner. 2013. Undergraduate conceptions of the field of computer science. In *ACM International Computing Education Research.* ACM, New York, 107–114.

[25] Michael Hewner and Mark Guzdial. 2008. Attitudes about computing in post-secondary graduates. In *ACM International Workshop on Computing Education Research.* ACM, New York, 71–78.

[26] Suzanne Hidi and K Ann Renninger. 2006. The four-phase model of interest development. *Educational psychologist* 41, 2 (2006), 111–127.

[27] Mark A Holliday and David R Luginbuhl. 2004. Peer-centered service learning. In *Frontiers in Education.* IEEE, USA, F3E–1–6.

[28] Beth K Humberd and Elizabeth D Rouse. 2016. Seeing you in me and me in you: Personal identification in the phases of mentoring relationships. *Academy of Management Review* 41, 3 (2016), 435–455.

[29] David Marshall Hunt and Carol Michael. 1983. Mentorship: A career training and development tool. *Academy of management Review* 8, 3 (1983), 475–485.

[30] Thomas E. Keller. 2005. The stages and development of mentoring relationships. *Handbook of youth mentoring* 1 (2005), 82–99.

[31] Harold R Kerbo. 1996. *Social stratification and inequality: Class conflict in historical and comparative perspective.* McGraw-Hill College, New York.

[32] Rhodes Jean E. Klaw, Elena L. and Louise F. Fitzgerald. 2003. Natural mentors in the lives of African American adolescent mothers: Tracking relationships over time. *Journal of Youth and Adolescence* 32, 3 (2003), 223–232.

[33] Andrew J Ko. 2009. Attitudes and self-efficacy in young adults' computing autobiographies. In *IEEE Symposium on Visual Languages and Human-Centric Computing.* IEEE, New Jersey, 67–74.

[34] Jean Lave and Etienne Wenger. 1991. *Situated learning: Legitimate peripheral participation.* Cambridge university press, Cambridge.

[35] Colleen M Lewis, Ruth E Anderson, and Ken Yasuhara. 2016. I Don't Code All Day: Fitting in Computer Science When the Stereotypes Don't Fit. In *ACM Conference on International Computing Education Research.* ACM, New York, 23–32.

[36] Jane Margolis, Rachel Estrella, Joanna Goode, Jennifer Jellison Holme, and Kim Nao. 2010. *Stuck in the shallow end: Education, race, and computing.* MIT Press, Cambridge, MA.

[37] Jane Margolis and Allan Fisher. 2003. *Unlocking the clubhouse: Women in computing.* MIT press, Cambridge, MA.

[38] J Mitchell Miller, JC Barnes, Holly Ventura Miller, and Layla McKinnon. 2013. Exploring the link between mentoring program structure & success rates: Results from a national survey. *American Journal of Criminal Justice* 38, 3 (2013), 439–456.

[39] Emma Norling. 1995. Encouraging networking through informal mentoring: a look at a newly-established mentor scheme. In *Second Australasian Women in Engineering Forum.* RMIT, Australia.

[40] Youn Joo Oh, Yueming Jia, Mhora Lorentson, and Frank LaBanca. 2013. Development of the educational and career interest scale in science, technology, and mathematics for high school students. *Journal of Science Education and Technology* 22, 5 (2013), 780–790.

[41] Becky Wai-Ling Packard and Paul F Conway. 2006. Methodological choice and its consequences for possible selves research. *Identity* 6, 3 (2006), 251–271.

[42] SS Pisimisi and MG Ioannides. 2005. Developing mentoring relationships to support the careers of women in electrical engineering and computer technologies. An analysis on mentors' competencies. *European journal of engineering education* 30, 4 (2005), 477–486.

[43] K Ann Renninger and Suzanne Hidi. 2011. Revisiting the conceptualization, measurement, and generation of interest. *Educational Psychologist* 46, 3 (2011), 168–184.

[44] Ashley Robinson, Manuel A Pérez-Quiñones, and Glenda Scales. 2015. Understanding the attitudes of African American middle school girls toward computer science. In *Research in Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT), 2015.* IEEE, New York, 1–8.

[45] Amber Settle, Sarah Pieczynski, Liz Friedman, and Nathan Kizior. 2013. Evaluating a Prospective Student Mentoring Program. In *International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS).* IEEE, USA, 1.

[46] Selcuk R Sirin. 2005. Socioeconomic status and academic achievement: A meta-analytic review of research. *Review of educational research* 75, 3 (2005), 417–453.

[47] Claude M Steele and Joshua Aronson. 1995. Stereotype threat and the intellectual test performance of African Americans. *Journal of personality and social psychology* 69, 5 (1995), 797.

[48] Gloria Childress Townsend. 1996. Viewing video-taped role models improves female attitudes toward computer science. *ACM SIGCSE Bulletin* 28, 1 (1996), 42–46.

[49] Gloria Childress Townsend. 2002. People who make a difference: mentors and role models. *ACM SIGCSE Bulletin* 34, 2 (2002), 57–61.

[50] Etienne Wenger, Richard Arnold McDermott, and William Snyder. 2002. *Cultivating communities of practice: A guide to managing knowledge.* Harvard Business Press, Cambridge, MA.

[51] Sarah Wood and Evan Mayo-Wilson. 2012. School-based mentoring for adolescents: A systematic review and meta-analysis. *Research on Social Work Practice* 22, 3 (2012), 257–269.

[52] Sarita Yardi and Amy Bruckman. 2007. What is computing?: bridging the gap between teenagers' perceptions and graduate students' experiences. In *ACM International Workshop on Computing Education Research.* ACM, New York, 39–50.