Community College Journal of Research and Practice

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ucjc20

Computer Science Intensive Intervention to Prepare and Engage Underrepresented Novice Students at Community College

Louise Ann Lyon, Colin Schatz, Yukie Toyama & David Torres

COMMUNITY COLLEGE JOURNAL OF RESEARCH AND PRACTICE
https://doi.org/10.1080/10668926.2021.1894508

Computer Science Intensive Intervention to Prepare and Engage Underrepresented Novice Students at Community College

Louise Ann Lyon[a], Colin Schatz[b], Yukie Toyama[c], and David Torres[a]

[a]ETR, Scotts Valley, CA, USA; [b]Computer Science, Mills College, Oakland, CA, USA; [c]University of California, Berkeley, CA, USA

ABSTRACT

As open-access institutions serving diverse student populations, community colleges are perfect settings for broadening participation in computing efforts in higher education. The very nature of open access, however, places students with a wide variety of previous experience in the same introductory computer science classroom, intimidating the students with little programming exposure, many of whom are traditionally underrepresented in computer science. This paper reports on a pre-semester, intensive program designed to increase the computer science confidence and motivation of students with no previous programming exposure implemented at a community college in California. Framed using social cognitive career theory, results from the accompanying research project indicate preliminary success; we found the program to be well received by the majority of students, who had increased self-efficacy and interest in computer science. Implications for practice and research are discussed.

The pervasiveness of technology in the workplace and an ongoing shortage of educated workers to fill jobs requiring programming skills have elevated the importance of increasing the numbers of college graduates with computer science degrees. Compounding the problem of technology worker scarcity is the chronic underrepresentation of certain groups such as women and minorities in computing fields. This paper reports on an intervention implemented at a community college in California targeted at broadening participation in computer science through addressing the lack of computer science experience and engagement.

While many interventions and research projects have focused on the recruitment and retention of students from underrepresented groups in computing at bachelor's degree-granting institutions, far less work has been done in the context of community colleges, despite the fact that individuals from these groups attend community colleges in disproportionate numbers. Community colleges are not only widely diverse in demographic categories such in race/ethnicity and socioeconomic status but also include large categories of students less frequently represented in four-year institutions, including older students, veterans, first-generation college attenders, and parents of small children (American Association of Community Colleges, 2018). Community colleges are thus poised to be key institutions in efforts to broaden participation in computing and are deserving of more attention.

Introductory computer science (CS1), the primary entry course for students pursuing a computing pathway, enrolls students with an extremely broad range of computer science experience, in part due to the open-access nature of community colleges. This variation in student preparation, in our observations, appears to depress the confidence and self-efficacy of less-prepared students, who may compare themselves negatively to classmates with significant computing experience. The social and psychological struggles of CS1 students that have less exposure to computing led us to create and test

CONTACT Louise Ann Lyon louann.lyon@etr.org ETR, 5619 Scotts Valley Drive, Suite 140, Scotts Valley, CA 95066

a targeted intervention – named *Code Jam* – designed to boost the preparation and engagement of novice students, many of whom are from underrepresented groups. The first Code Jam was conducted at Las Positas College (LPC) in California in early 2020, and this paper gives a description of the program and reports on preliminary findings from the research and evaluation done in conjunction with Code Jam.

The Code Jam approach was grounded in theories of postsecondary and career interest development. Social cognitive career theory (SCCT; Lent et al., 1994) posits learning experiences and the resulting student self-efficacy and self-concept build messages about possible and appropriate fields of study and work. The conceptualization of this study draws from SCCT as adapted by Sax et al. (2017) to focus on computing, using the framework as a guidepost (Kaplan & Maxwell, 2005). In conjunction with our first implementation of the Code Jam intervention, we conducted pilot research and evaluation to investigate the following questions:

- RQ1: *Did students' interest and confidence in computer science increase after participating in the Code Jam program?*
- RQ2: *How well was the program received by the students?*

Code jam program description

While many colleges offer loosely categorized *CS0* semester-long courses that precede the formal first course in a computer science program's core major sequence, we intentionally designed a week-long, intensive, pre-CS1 intervention to expedite student time to graduation. Code Jam consisted of material that prepared students for the skills and topics included in CS1 and provided an expansive vision of the field of computer science, all with an aim to increase students' interest and confidence. We modeled our Code Jam program on two similar and successful programs: a Las Positas College pre-semester intensive math program (*Math Jam*) and The Siena Plan for Attracting and Retaining Computer Scientists (*SPARCS*) at Siena College (Vandenberg et al., 2018). Code Jam was designed as a non-credit set of workshops and related experiences that were voluntary and free to students, like Math Jam, and was a brief and intense computer science program, similar to SPARCS. We chose this format rather than embedding the Code Jam experience in an existing course because existing courses (e.g., CS1) typically contain a non-trivial proportion of experienced students and have substantial practical constraints dictated by formal curricular requirements and scheduling.

The goal of Code Jam was to help students visualize belonging in computing through increased self-efficacy and outcome expectations, as well as to arm novice students with some of the core knowledge and conceptual understanding their more-experienced peers have at the beginning of a CS1 course. The week-long program was built for students with no prior experience with computer programming, helping students explore computer science and experience introductory activities in a *low stakes* (Shah et al., 2013) setting with no grades or formal assessments. The design of this program was intended not only to better prepare students academically for CS1 but also to implement best practices that foster interest, motivation, and identification, particularly for underrepresented students. A primary goal was to expose the students, through a wide array of activities (Denner & Campe, 2018), to the thinking and problem-solving that goes into computer science (See Table 1). In addition, Code Jam was deliberately designed as neither a programming boot camp nor an accelerated version of CS1. Instead, it was a combination of active computer-based and unplugged activities (Freeman et al., 2014) that, in combination, was intended to give students both an overall sense of what computer science *is* – including job and career possibilities as well as computing concepts (Shah et al., 2013) – and an early set of engaging, successful experiences for students intending to go on to a formal computer science course but with potential concerns about their abilities or uncertainty about what that experience would be like (Shah et al., 2013).

Table 1. Code Jam Schedule.

	Morning 9:00–11:45	Lunch/Smart Shop 11:45–12:45	Afternoon 12:45–3:00
Mon	Intro/Community Building PeopleBots	What We Carry: How Our Stories Can Help or Hinder Academic Progress	Block Robots (Parallax + BlocklyProp)
Tue	Numeric Aerobics	Panel: WorkDay employees	On the Move 1 (Thunkable)
Wed	On the Move 2 (Thunkable)	Using BRAIN RESEARCH to change how you LEARN	Sharing Your Story (Twine)
Thu	Cooking Up Eye Candy (Processing)	Panel: Current LPC CS students	Crank It Up (Python)
Fri	Vehicles of the Future (Parallax + C++)	Successfully Marketing YOU – Resume and Interview Strategies	Focus Groups

Recruitment and student enrollment procedures

To recruit participants, we pursued marketing and outreach to students through numerous avenues beginning several months before the program. In-person visits were made by Code Jam instructors to meetings of specific student cohort programs focused on underrepresented student groups, including the college's Puente (Latinx students), Umoja (African diaspora students), and HSI Gateway to STEM Success Programs (Latinx students in STEM majors). At the time, the college did not have any student groups focused on women in STEM or specific STEM disciplines, but we sought women students' feedback on marketing images and language and worked informally with faculty to encourage individual women students to consider the program. Marketing materials and reminders were subsequently distributed to students by the faculty and counseling staff affiliated with those programs. In addition, current instructors who were scheduled to teach CS1 in the semester following Code Jam were given marketing materials and asked to reach out to their already-enrolled students and to encourage novice students to enroll in Code Jam and to "spread the word" to friends who had not yet taken any CS courses. We also solicited the assistance of the college's director of the tutorial center, who collaborated with us by informing current peer tutors of the program. Tutors were encouraged to share the information about Code Jam with their friends and their tutees. These more traditional, general forms of outreach, alongside the targeted recruitment mentioned above, helped awareness of the program reach general student populations that included White and Asian students and women.

Due to institutional arrangements for activities that are not traditional courses, Code Jam registration was tracked through the school's Community Education office. Interested students were asked to fill out a survey before being registered in the class to allow the organizers to filter students based on past computing experience. Initially, students were informed that they would only be allowed to enroll if they had no past programming experience and if they were enrolled in CS1 for the semester immediately following Code Jam. When the class did not fill by 3 weeks prior to the start date, we opened up enrollment to students who insisted that they were interested even though the class was designed for true novices and to students who were not enrolled in CS1 the subsequent semester. This resulted in an enrollment of three students out of 18 (17.6%; two male and one female) with some programming experience. The first day of Code Jam, instructors explicitly noted the presence in class of students with varying experience, and presented a list of classroom agreements, including one directed explicitly at non-novice students ("If you have prior experience or a lot of relevant knowledge already, be especially mindful not to deprive others of learning opportunities"). This agreement was highlighted by instructors verbally on the first day of Code Jam with the suggestion that more advanced questions be held to discuss with an instructor during the break.

Conceptual framework

In line with previous work focused on better understanding the choices of underrepresented groups to participate in computer science (Lent et al., 2008, 2011; Sax et al., 2017), we used a conceptual framework of Social Cognitive Career Theory (SCCT) to guide our study (Lent et al., 1994). SCCT, derived from Bandura's (1986) social cognitive theory, describes the process – both personal agency and extra-personal factors – by which individuals develop career interests over time. In this work, we examine the early mechanisms in the SCCT model surrounding the Code Jam learning experience to investigate the attitudes of students at the time of the program and student evaluations of the program. Specifically, we report here on the *personal characteristics* and *contexts* of the students who enrolled in Code Jam, student's evaluation of the *learning experience*, and measures of student s*elf-efficacy* (which we measured as confidence in CS), along with *outcome expectations* and *interest* (which we measured as interest/usefulness of CS). The SCCT framework suggests that these psychosocial factors can help explain the ongoing choices students make to persist in or drop out of a field of study and work, including computing fields. We used the conceptual framework as a guide (Kaplan & Maxwell, 2005) to highlight psychosocial constructs that can lead to a choice to pursue computer science rather than as a set of hypotheses. This exploratory project expands on previous work by focusing on an understudied population of students at a community college and by describing and examining a short-term, intensive program to investigate if there are any indications that it could be successful in helping to broaden participation in computing.

Methods

Setting

Las Positas College (LPC) is a two-year public Hispanic-serving (HSI) community college located 40 miles southeast of San Francisco that enrolls approximately 18,000 students annually. Both the college's enrollment as a whole and the diversity of its student population have increased dramatically recently, including a rapidly increasing number of Hispanic students. While the proportion of White students at LPC has decreased from 74% to 45.8% over the past decade, the percentage of Hispanic students has nearly tripled – up from 10% in 1997 to 28% in 2014, per the Chabot-Las Positas Institutional Research Dataset. Asian students are the next largest ethnic group, at 18% of the student population in the fall of 2019. A consistently, slightly higher, proportion of women than men enroll at the college, with females making up 52% of the student population in 2019, down from 54% in 2010. The proportion of non-traditionally aged students (25 and older) has remained fairly constant over the past decade, comprising 36% of the student population in 2010 and dropping to 31% by 2019. First-generation college students were 19% of the college population in 2019, having varied between a high of 20% and a low of 17% since 2010. LPC has a tiny international student population (e.g., 109 students in the fall of 2019), with the largest group from China and a larger proportion of women than men.

At Las Positas College, CS1 (*Computing Fundamentals I*) is the first course required for the A. S. degree in Computer Science, as well as meeting requirements for several other degrees and certificates and college-wide General Education requirements for any A.S. or A.A. degree. CS1 is taught in C++ and is formally articulated to many four-year college and university computer science department courses. CS1 has no prerequisites and is considered appropriate for students with no prior experience.

Within CS1, student enrollment and success data mirror larger trends in demographic discrepancies in the field. For instance, 20% of students enrolled in CS1 were Hispanic, compared to 29% of total college enrollments during the period of Fall 2014 through Spring 2017, and the proportion of students completing the course with a passing grade was 55% for Hispanic students, compared to 63% overall. Similarly, when looking at gender, within the same period, only 20% of students enrolled in CS1 identified as female. Overall success rates by gender within that set of students were nearly identical – 63.9% for women versus 63.3% for men – but discrepancies emerge when looking at intersectional data, with success rates for female students slightly higher (72%) than for male students (68%) overall, but among Hispanic/Latino students the success rate for male students was higher (56%) than for female students (50%). There are a few likely root causes for these discrepancies, including structural inequities in K-12 schooling that are correlated to geography and ethnicity. As data like these large grain-size success-rate statistics make clear, when examining equity issues in CS and technology, we need to not only examine individual demographic variables, such as gender or ethnicity, but look at interactions between them.

Participants

A total of 18 students at LPC participated in the first Code Jam program. Of these, one student was dropped from all data analysis since he was a minor at the time of the program and the additional IRB requirements for underage populations was not met. All further reporting uses a total participation number of 17 students. In the SCCT model, individuals bring both personal characteristics (e.g., gender, race/ethnicity) and background (e.g., mother's and father's education) into a learning experience. In this study, of the 17 participants, 10 were female (58.8%). The participants' age ranged from 18 to 44 years (Mean = 24.88 years). The most frequently reported race/ethnicity was Asian (58.8%), followed by Hispanic/Latinx (17.6%) and White (11.8%). Eight of the participants' (47%) fathers and 10 of the participants' mothers (58%) had less than a university education. Fourteen of the students (82.4%) were true novices with no prior programming experience. About 41% of the students had a job(s), 30% were married, and 23.5% had a dependent who lived with them. A great majority (88%) had a plan to get a bachelor's degree or higher.

Procedure

The Code Jam program was offered during the first week of January 2020 at LPC. The 17 participants took the pre-treatment survey on the first day of the program while they took the post-survey on the last day (the fifth day) of the program; our primary interest was in assuring that the program did not have any negative effects on participating students. We used a one-group, pre-post design, to examine whether participants' confidence and interest in computer science increased after having participated in the program. We are aware this research design does not allow causal inference of the program's effectiveness without a random assignment of a treatment and control group, but our focus was on describing key program elements and documenting preliminary feedback from the program participants to guide program modifications and implementation at scale in the future.

In addition to the pre-post survey administration within the Code Jam program, the same pretest survey was administered to 128 students who enrolled in CS1, which started the second week of January 2020. The data from this larger CS1 sample, which included 13 of the 17 Code Jam participants, were used to validate the survey instrument used in this study by examining its factor structure and reliability. All statistical analyses were conducted using R version 3.6.2.

Measures

Survey scales

For the constructs of interest – namely self-efficacy, interest, and outcome expectations – we used the following three scales from the Student CS Attitude Survey (Haynie, 2017): (a) confidence (students' confidence in their ability to learn CS skills and solve CS problems), (b) interest (students' interest in learning CS), and (c) usefulness (students' beliefs in the usefulness of learning CS). This survey was originally designed for students in eighth grade and above, and thus was more general and relevant for our study sample than the survey instruments used by SCCT scholars (Lent et al., 2008, 2011), which were focused on CS and STEM field majors at 4-year colleges. Yet, the three scales we chose were specific enough to CS and a detailed examination of items constituting the scales ensured the close match between the scales and the SCCT.

Each scale had five items with a 4-point Likert scale ranging from "strongly disagree" to "strongly agree". All items were positively worded, such as "I am sure I could do advanced work in computer science" (confidence), "I would like to use computer science to solve problems" (interest). We made minor modifications to the wording of a few statements to increase clarity.[1] A full set of items used in the current study can be found in the appendix.

Our initial confirmatory factor analyses with the larger CS1 sample indicated that the interest and usefulness scales were highly correlated (r = .99), therefore these two scales were combined as the interest/usefulness scale in the subsequent analyses (details of the factor analyses can be found in the appendix). Both the confidence (α = .78) and interest/usefulness (α = .89) scales proved quite reliable, and they were moderately correlated with each other (r = .73; for a visual representation of this relationship see Figure A2 in Appendix A).

Program evaluation questions

In addition to the survey scale measures, the post-survey asked students evaluation questions about their experience in the Code Jam program using a modified version of the LPC semester-end evaluation questionnaire. These questions were designed to understand students' perceptions of the program and to plan modifications for the next iteration of Code Jam. Thirteen evaluative questions were given in selected response format with the same 4-point options as the attitudinal scales (e.g., "The Code Jam material was interesting") while four questions were given as constructed response format (e.g., "What did you especially like about the Code Jam experience?").

Data analysis

To answer the first research question (RQ1: "Did students' interest and confidence in computer science increase after participating in the Code Jam program?"), paired sample t-tests were performed using total sum scores at pre- and posttests for the two scales. One student skipped one of the confidence scale questions at the pretest, thus the most likely value was imputed based on the examination of response patterns in the rest of the items within the scale. To better handle the missing value, the Rasch-model-based analyses were conducted to estimate students' attitudinal attributes before and after the Code Jam program, and paired sample t-tests were performed using the Rasch-based scores. Since both sets of analyses yielded similar results, we report the t-test results of the total sum scores in the remainder of this article.

To answer the second research question (RQ2: "How well was the program received by the students?"), students' responses to evaluative Likert scale items as well as constructed-response items were examined.

Results

Self-efficacy

Paired sample t-tests from 17 Code Jam students were used to answer RQ1. Figure 1 shows the distribution of confidence scale scores at the pretest as well as the posttest. A gray line pairs the pre- and posttest scores for each student. While on average participants' confidence scores increased from the pretest from the pretest (M = 14.47, SD = 3.26, max possible = 20) to the posttest (M = 15.24, SD = 3.27), this increase was not statistically significant, $t(16) = 1.01$, $p = .329$. A closer examination of Figure 1, however, shows that the confidence scores increased for the majority of students (11 out of 17, or 65%) although five students' scores went down as much as seven points for student 18.

Among the five statements related to students' confidence in CS, "I can learn CS without a teacher" saw the largest increase in the percentage of students agreeing or strongly agreeing (from only 29% at the pretest to 47% at the posttest). Conversely, no pre-post change was observed with a statement "I think I will do well in CS": 88% of the students agreed or strongly agreed with the statement at both time points. Two additional statements out of the five confidence statements also showed similar ceiling effects, with over 75% of students agreeing or strongly agreeing even at the pretest.

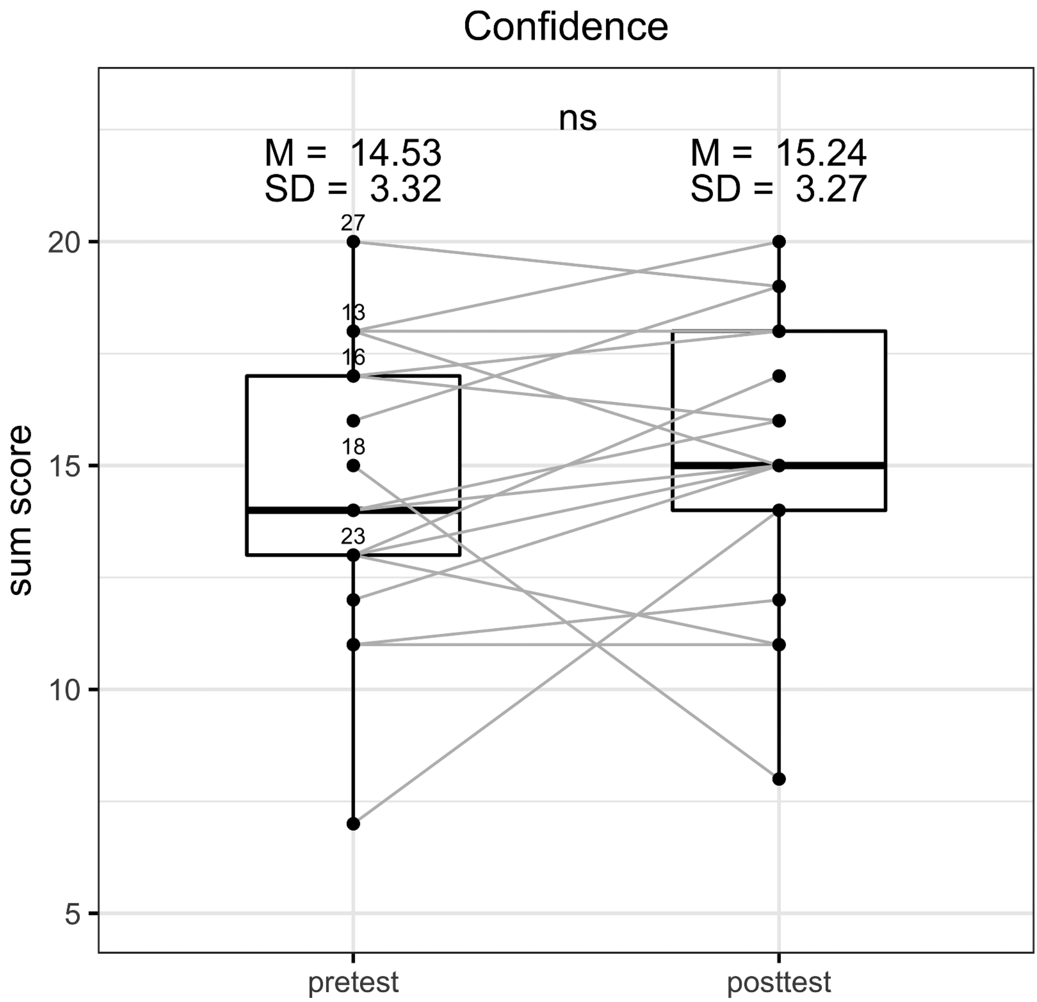

Figure 1. Confidence Scale Scores at Pretest and Posttest. *Note*. ns = not statistically significant. Seventeen students took a total of five items. Max possible = 20 points. ID numbers shown for students whose score decreased from pretest to posttest.

Interest/outcome expectations

Figure 2 shows the distribution of interest/usefulness scale scores at the pre- and posttest. Consistent with the confidence score results, on average, participants' interest/usefulness scores increased from the pretest (M = 32.94, SD = 3.77, max possible = 40) to the posttest (M = 33.94, SD = 4.35). However, this increase was not statistically significant, $t(16) = 1.446$, $p = .167$. A more detailed examination of Figure 2, however, reveals that the interest and usefulness scores increased for the majority of students (11 out of 17, or 65%) although five students' scores went down – as much as five points for student 18.

Among the 10 statements related to interest and usefulness of CS, "I hope my career will require the use of CS" saw the largest gain in the percentage of students agreeing or strongly agreeing (from 76% at the pretest to 94% at the posttest). In contrast, all students (100%) agreed or strongly agreed with the following two statements at both time points: "I would like to use CS to solve problems" and "Knowledge of CS will help me earn a living." In fact, all 10 items in the interest/usefulness scale showed ceiling effects, with 76–100% of students agreeing or strongly agreeing with the statements at the pretest. Taken together, the results from the two survey scales show that students were very enthusiastic about CS at the outset of the Code Jam program.

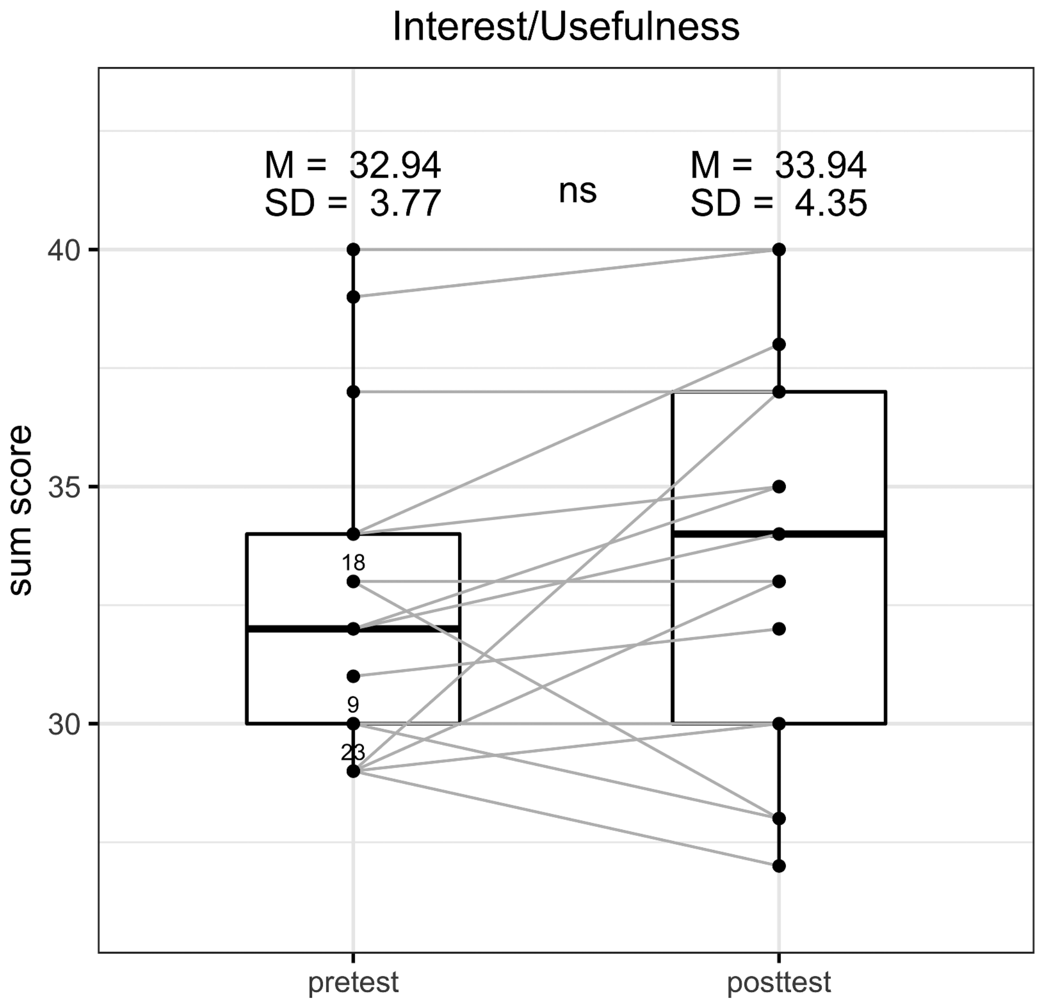

Figure 2. Interest/Usefulness Scale Scores at Pretest and Posttest. *Note*. ns = not statistically significant. Seventeen students took a total of five items. Max possible = 20 points. ID numbers shown for students whose score decreased from pretest to posttest.

To learn more about student 18, whose confidence and interest/usefulness levels decreased dramatically from the pretest to the posttest, we closely examined the demographic data and qualitative data from surveys. We found that the student was an international, and English as Second Language (ESL) student who had to "always focus to catch up with the teacher" and who found Code Jam "too intense," according to her class evaluation data (see next section). From researcher casual conversations with such students during Code Jam, the lack of a textbook and the resulting inability to do prereading in preparation for each day of class made the style of Code Jam incompatible with the usual practices of students who may struggle to understand English as well as to understand class content. This feedback perhaps would explain the drop in confidence and interest levels for this student.

Students' program evaluation

Overall, student evaluation responses to the Code Jam program were extremely positive. All students agreed or strongly agreed with the statements "the Code Jam material was interesting," "the activities we did in Code Jam increased my interest in computer science," and "I made friends while participating in Code Jam." All but one student agreed with the statements "the activities we did in Code Jam increased my confidence that I can successfully complete an introductory computer science class such as CS1" and "I felt encouraged to ask questions during Code Jam." Least positive were responses to the statement "I felt like I belonged in Code Jam," with three students (or 17.6%) disagreeing. Overall, students found the pacing of the class about right, according to answers to several questions.

Write-in responses for suggested improvements to Code Jam focused primarily on logistical ideas, such as building in more breaks and allowing more time to speak with professional panelists after their presentation. Positive comments about Code Jam fell into three broad themes: the variety of interesting activities/coding platforms (e.g., "liked being introduced to many different types of coding languages and platforms to which coding was necessary from interactive fiction to the robots"), the chance to learn about topics before the first day of CS1 (e.g., "I like the opportunity to learn something new in advance of my class so that I will have some confidence when class begins"), and the positive interactions with the instructors, TAs, and other students (e.g., "I loved talking to the friendly instructors"). When prompted to describe the Code Jam experience as if summarizing it for friends, one student wrote:

> To take Code Jam would be similar as to putting sunscreen on you [sic] skin before you get out on the beach when it is sunny and you could get a sunburn. I feel as if I am not going to burn (get left behind) in class now because I have applied the sunscreen (Code Jam). I understand concepts and some syntax that will prepared [sic] me to become mentally ready for what this class has in store for the semester.:)

Discussion and implications

The Code Jam program described here helps us better understand how we might recruit a broad diversity of students in computer science and prepare them for CS1 at community colleges, with data from the pilot instance of the program serving to inform iteration of program design and pointing to important aspects of evaluation and investigation going forward. Results from this exploratory study demonstrate that the program successfully attracted historically underserved female students: about 60% of the program participants were female compared to 20% in a typical CS1 course at LPC. Additionally, the program was well received by the majority of students, and their interest and confidence in computer science increased after participating in the Code Jam program. The increases averaged across the participants were not statistically significant, due in part to a small sample size as well as to the ceiling effects that reflected participants' enthusiasm toward CS even at the outset of the program. However, we find it encouraging that the program did not lead to, on average, any *decrease* in interest and confidence.

To apply lessons learned from these results and to examine further the efficacy and effects of an intervention like Code Jam, one future avenue would be a random assignment of students to treatment and comparison conditions; this type of randomized control trial was not appropriate in the investigation reported here, occurring during the formative phase of an intervention in which promising program features, necessary supports, and possible barriers are being examined descriptively (Bakker, 2018).

Another lesson from this study would be that a future study should add survey scales that are sensitive to capture the pre-post change in a relatively short intervention like Code Jam (We note that the survey scales used in this study were originally designed for students taking a yearlong CS course; Haynie, 2017). But we also argue that some scales should ideally capture changes in students' self-efficacy, interest, and outcome expectations in a longer time span as they navigate a pathway to and through classes toward a college major. Theorists argue that students' interest in and attitudes toward a major and career get revised through the influence of ongoing interactions between a person and their environment over time (Bandura, 1986).

Despite the statistically insignificant results with the survey scales, our descriptive findings still illustrate that a short-term, intensive program addressing challenges of true novices is a promising avenue to broaden participation in computing at community colleges. The findings also imply some general prescriptions for elements to be included in any similar effort at other community colleges. In particular, a low/no stakes environment combined with a relaxed, comfortable, cooperative atmosphere will likely serve to engage students, boosting self-efficacy and helping motivate them to persist in computer science. A diversity of instructors and topics is similarly important, giving participants exposure to a variety of different experiences and platforms. This not only provides an alternative to the conventional structure of a CS1 course but builds wider awareness among novice students of “what can I do with this?” and cultivates their sense that they both know what to expect from the formal course and will be able to handle it. As student 18 in our sample indicated, implementation of such a program should provide text-based materials as well as other supports for English as a Second Language and English Language Learners. With such supports in place, recruitment from international student groups could be a fruitful avenue of participant enlistment.

The demographics and background of Code Jam participants represent a crucial population to engage and support both before and during formal computer science courses, and our results and feedback from Code Jam suggest key ingredients that could be fruitfully adapted in a semester-long intro course. These include targeted marketing and recruitment of true novice students and students from marginalized groups, explicitly emphasizing that no prior experience is needed or expected. We note that our recruitment efforts for the next round of Code Jam will focus on attracting Hispanic/Latinx students as well as female students. In this pilot instance of the program, the 17.6% Hispanic/Latinx student rate was lower than the LPC 20% CS1 participation rate and 29% college-wide rate of Hispanic/Latinx participation; we would like to increase the proportion in future program occurrences. Similarly, formal CS1 courses could include numerous activities that provide for formative assessment in the form of *no stakes* practice in addition to those that must be used for formal student evaluation, as suggested also by other researchers (Flynn & Haberman, 2018; Nicol & Macfarlane-Dick, 2006).

It may also be worthwhile to reflect on ways to incorporate some of Code Jam’s key ingredients into a formal course even if they may be difficult to replicate fully in a typical formal course within an institution’s local constraints. For instance, CS1 courses often focus on one language and platform (e.g., Java used to build desktop applications), but instructors could consider taking some time – even part of one class meeting a few different times per semester – to introduce other languages or platforms. Similarly, community college courses rarely include TA roles in the classroom, but efforts to direct students to campus peer tutoring services and create opportunities for them to learn the skills interactively could yield substantial benefit (Freeman et al., 2014; Muller et al., 2018). Finally, a greater sense of community and a personal connection to multiple instructors could be cultivated numerous ways, including shared events outside class time, intentional scheduling of guest presentations from other faculty in the department during introductory courses, and some portion of office hours held in communal and student-centered spaces on campus.

Limitations

The goal of this paper was to describe the pilot instance of a new program designed to increase the self-efficacy and outcome expectations for a diversity of students with no prior experience enrolling in a formal CS course and to report on early findings from the associated research. The pilot nature of the program and the resulting low numbers of students participating in the research limit the generalizability and reliability of the research associated with this project. However, the initial findings reported here show a promising potential of the program and give a direction for designing and implementing future instances of the pre-CS1, intensive intervention targeted at historically underserved students at community colleges.

Note

1. One of the usefulness items was shortened to "skills and knowledge from computer science will help me understand things in everyday life" from its original item "computing skills used to understand computer science material can be helpful to me in understanding things in everyday life."

Funding

This work was supported by the NSF grant [1900153].

References

American Association of Community Colleges. (2018). *Fast Facts*. American Association of Community Colleges. https://www.aacc.nche.edu/research-trends/fast-facts/

Bakker, A. (2018). *Design research in education: A practical guide for early career researchers*. Routledge.

Bandura, A. (1986). *Social foundations of thought and action*. Englewood Cliffs.

Denner, J., & Campe, S. (2018). Equity and inclusion in computer Science education. In S. Sentence, E. Barendsen, & C. Schulte (Eds.), *Computer science education: Perspectives on teaching and learning in school* (pp. 189–206). Bloombury.

Flynn, W., & Haberman, A. (2018). Low stakes, no stakes: Formative classroom assessment techniques. *Teaching, Learning & Assessment*, *27*. https://digscholarship.unco.edu/tla/27

Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, *111*(23), 8410–8415. https://doi.org/10.1073/pnas.1319030111

Haynie, K. C. (2017). *Student computer science attitude survey: CS principles*. http://www.pearweb.org/atis/tools/86

Kaplan, B., & Maxwell, J. A. (2005). Qualitative research methods for evaluating computer information systems. In J. G. Anderson & C. Aydin (Eds.), *Evaluating the organizational impact of healthcare information systems* (pp. 30–55). Springer.

Lent, R. W., Brown, S. D., & Hackett, G. (1994). Toward a unifying social cognitive theory of career and academic interest, choice, and performance. *Journal of Vocational Behavior*, *45*(1), 79–122. https://doi.org/10.1006/jvbe.1994.1027

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

Lent, R. W., Lopez, F. G., Sheu, H.-B., & Lopez, A. M., Jr. (2011). Social cognitive predictors of the interests and choices of computing majors: Applicability to underrepresented students. *Journal of Vocational Behavior*, *78*(2), 184–192. https://doi.org/10.1016/j.jvb.2010.10.006

Muller, O., Shacham, M., & Herscovitz, O. (2018). Peer-led team learning in a college of engineering: First-year students' achievements and peer leaders' gains. *Innovations in Education and Teaching International*, *55*(6), 660–671. https://doi.org/10.1080/14703297.2017.1285714

Nicol, D. J., & Macfarlane-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in Higher Education*, *31*(2), 199–218. https://doi.org/10.1080/03075070600572090

Rosseel, Y. (2012). Lavaan: An R package for structural equation modeling and more. Version 0.5–12 (BETA). *Journal of Statistical Software*, *48*(2), 1–36. https://doi.org/10.18637/jss.v048.i02

Sax, L. J., Lehman, K. J., Jacobs, J. A., Kanny, M. A., Lim, G., Monje-Paulson, L., & Zimmerman, H. B. (2017). Anatomy of an enduring gender gap: The evolution of women's participation in computer science. *The Journal of Higher Education*, *88*(2), 258–293. https://doi.org/10.1080/00221546.2016.1257306

Shah, N., Lewis, C. M., Caires, R., Khan, N., Qureshi, A., Ehsanipour, D., & Gupta, N. (2013, March 6-9). *Building equitable computer science classrooms: Elements of a teaching approach* [Paper presentation]. Proceeding of the 44th ACM technical symposium on Computer science education. Denver, CO, United States.

Vandenberg, S., Small, S. G., Fryling, M., Flatland, R., & Egan, M. (2018, February 21-24). *A summer program to attract potential computer science majors* [Paper presentation]. Proceedings of the 49th ACM technical symposium on computer science education, Baltimore, MD, United States.

Appendix A.

Confirmatory Factor Analysis

We adopted three scales from the Student CS Attitude Survey (Haynie, 2017), namely Confidence, Interest, and Usefulness, with a few minor changes to the wording of a few items. Using the Lavaan package in R (Rosseel, 2012), we conducted confirmatory factor analyses with the larger CS1 sample (n = 128), initially fitting a single-factor model and three-factor model, to examine the structure of the correlations among the 15 items listed in Table A1. The three-factor model showed adequate goodness of fit by most of the measures ($\chi2(87) = 218.4$, $p < .001$; CFI = 0.84; SRMR = 0.077; RMSEA = 0.112). The standardized loadings and the correlations among the three latent variables are shown in Figure A1. Additionally, a $\chi2$ difference test with ANOVA indicated that the three-factor model fits significantly better than the single factor model ($\chi2$ (1) = 56.2, $p < .001$).

However, the results of the three-factor model indicated that the Interest scale and the Usefulness scale were highly correlated (r = 0.99, see Figure A1). Therefore, a two-factor model was fitted, combining the Interest and Usefulness items into a single factor. The two-factor model showed an adequate goodness of fit ($\chi2$ (89) = 222.3, $p < .001$; CFI = 0.84; SRMR = 0.078; RMSEA = 0.111). As can be seen in Figure A2, these two latent variables were moderately correlated (r = 0.73), suggesting they should be treated as separate factors. Thus, the subsequent analyses were conducted using sum scores for the two scales: (a) Confidence and (b) Interest/Usefulness combined.

Table A1. Attitudinal Scales and Items Used.

Item	Item description
Confidence α = .78 (n = 128)	
co_q1	I am sure I could do advanced work in computer science
co_q2	I think I will do well in computer science
co_q3	I can learn computer science without a teacher to explain it
co_q4	I have self-confidence when it comes to computer science
co_q5	I am confident that I can solve problems by using computing
Interest & Usefulness α = .89 (n = 128)	
int_q1	I would take additional computer science courses if I were given the opportunity
int_q2	I would like to write computer programs
int_q3	I hope that my future career will require the use of computer science
int_q4	The challenge of solving problems using computer science appeals to me
int_q5	I would like to use computer science to solve problems
use_q1	Learning to use computing skills will help me achieve my career goals
use_q2	Computer science is a worthwhile and necessary subject
use_q3	Knowledge of computer science will help me earn a living
use_q4	Skills and knowledge from computer science will help me understand things in everyday life
use_q5	I'll need mastery of computer science for my future work

co indicates confidence, int indicates interest, and use indicate usefulness. Originally, these were separate scales

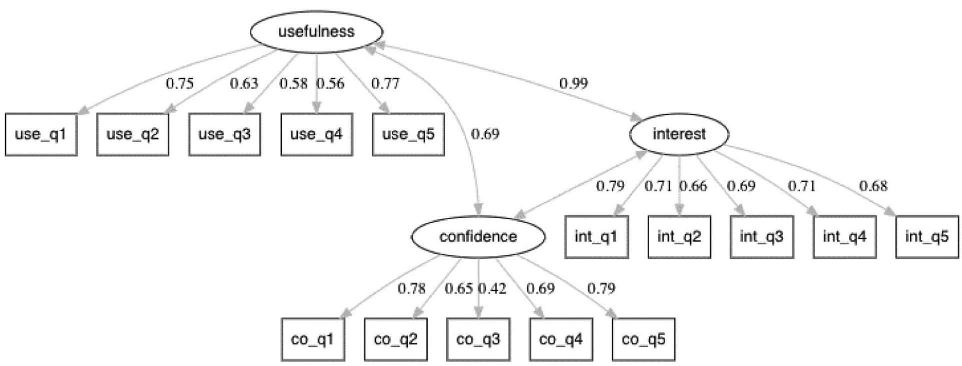

Figure A1. Standardized Factor Loading and Correlations Among Factors From Confirmatory Three-Factor Model.

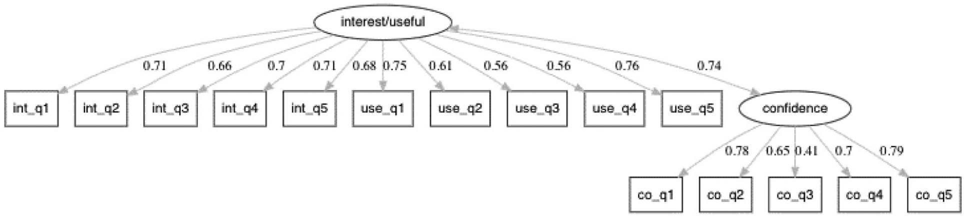

Figure A2. Standardized Factor Loading and Correlations Among Factors From Confirmatory Three-Factor Model.