The Persistent Effect of Pre-College Computing Experience on College CS Course Grades

Christine Alvarado, Gustavo Umbelino and Mia Minnes
University of California, San Diego

From Proceedings of the 2018 ACM SIGCSE Conference. Reprinted with permission.

Many college computer science majors have little or no pre-college computing experience. Previous work has shown that inexperienced students underperform their experienced peers when placed in the same introductory courses and are more likely to drop out of the CS program. However, not much is known about what, if any, differences may persist beyond the introductory sequence for students who remain in the program. We conducted a study across all levels of a CS program at a large public university in the United States to determine whether grade differences exist between students with and without pre-college experience, and if so, for what types of experiences. We find significant grade differences in courses at all levels of the program. We further find that students who took AP Computer Science receive significantly higher average grades—by up to a half grade—in nearly all courses we studied. Pre-

college experience appears to have a weaker relationship with retention and with low-stakes assessment grades. We discuss the limitations of these findings and implications for high school and college level CS courses and programs.

INTRODUCTION

Computer science as a discipline in higher education presents many challenges for educators and students. Because of the lack of standard CS expectations for K-12 students and widely varied access to computing, students arrive to college with a vast range of backgrounds and interests in the discipline. In addition, these experiences are not distributed randomly among all students: Women, members of some racial and ethnic groups, and those from low socioeconomic backgrounds tend to have less experience than White and Asian middle and upper class

men [13,14]. Students in these groups are also less likely to take Advanced Placement (AP) CS [6], in no small part because of lack of access [13,16]. Despite the many current efforts to level the playing field in K-12 CS education, different pre-college experience levels, whether due to access or to student choice, are likely to persist into college for some time.

At the college level, educators have developed a number of curricular approaches to level the playing field for students with less pre-college experience. Several universities have introduced a separate introductory course or track specifically to allow students with less pre-college experience to catch up [4,11,15]. A complementary approach uses curriculum that will not disadvantage students with no experience in introductory courses. For example, some redesigned introductory courses begin with content that students are unlikely to have had exposure to in high school (e.g. [5,25]). Others use a contextualized computing approach such as media computation, which has been shown to increase retention and decrease fail rates for non-majors and other students with no pre-college CS experience [7,22,24]. Other programs have courses that let students choose their own focus area based on their interest [9].

Little work has studied the outcome in later courses for students with no pre-college experience, and there is evidence that these students under-perform their experienced peers even in courses that employ a host of best practices [2]. We wished to explore whether students with less pre-college experience are less likely to do well not just in introductory courses, but also in courses later in the major? If performance differences disappear, at what point do they do so?

We examine the relationship between different pre-college CS experiences and students' grades and persistence in undergraduate CS classes at a large US public research university. We find that both students' self-assessed level of pre-college experience and their participation in specific pre-college experiences are associated with a difference in course performance at levels. We notice a particularly strong effect with students who take AP CS in high school. However, we find no significant difference in performance on low-stakes assessments of CS knowledge between students with pre-college experience and those with none. These findings have implications for ensuring all students get a high-quality computer science background in high school and on the design of our college-level courses and extracurricular opportunities.

RELATED WORK

Prior experience in CS (and related subjects) is widely shown to be an important factor of success and persistence in introductory CS at the college level [8,10]. One explanation for why prior experience is helpful is that students with prior experiences often have different motivations, confidence levels and goals. Recent work by Kori et al. [12] compared the attitudes, persistence and success of students with and without pre-college programming experience throughout

several introductory IT course sequences in Estonian universities. They found that students with pre-college experience received higher grades in their first semester than those without experience, and that they also had higher external motivation for their studies.

Prior experience can also affect students' sense of belonging and stress level. In a qualitative study, Tafliovich et al. found that students believe prior experience boosts confidence and that discussion forums can be stressful for those with no experience [23]. Indeed, Rogerson and Scott report that stress and fear are regularly associated with learning to program and can interfere with a student's ability to succeed [21].

The idea of Learning Edge Momentum as applied to CS1 [1,20] provides another explanation for performance differences due to prior experience. This theory suggests that rather than disappear, performance differences are likely to persist into later courses because once students learn one concept, it becomes easier for them to learn subsequent concepts. However, most prior work on the effect of pre-college experience on college CS success stops at CS1. Our study complements this prior work by looking at courses much further into the CS major, and by examining the role of several specific pre-college experiences students might have.

Recent work has also studied how pre-college experiences (among other factors) affect students' choice to major in computer science. McGill et al. find that pre-college outreach activities have some effect on students' choice of major in college, but that this effect appears to be stronger for men than for women. [17]. In a 2015 survey, Wang et al. found that pre-college academic opportunities, both formal and informal, were an important factor in girls' choice to major in computing in college, and that taking a pre-college CS course was more significant for girls than for boys in their decision to major in computing in college [26]. A natural follow-up study to the work we present here would examine whether the results we report are consistent for both men and women.

APPROACH

Based on previous work and our anecdotal interactions with students in the classroom, we postulated the following hypotheses:

- **H1:** Students with more pre-college CS experience will do better initially than their less experienced peers, but these differences will disappear as course level increases.
- **H2:** Many types of pre-college experience will provide an advantage over having no pre-college experience, but some experiences will stand out as particularly beneficial on their own regardless of other experience.

We collected data over four quarters (fall 2014 to fall 2015) from courses at all levels in our undergraduate curriculum. We included our two introductory programming tracks: a two-course sequence designed for students with no prior CS experience (CS1:n=1258, CS1.5:n=543), and a one quarter course

for those with prior experience ($n=411$). We note that the two-course introductory sequence employs many educational best-practices and has been shown in prior work to improve student outcomes and retention [18,19]. Also in the lower-division, we studied the basic data structures (CS2: $n=1117$) and computer systems/low-level programming ($n=913$) courses. The two required upper division courses we included were advanced data structures ($n=915$) and programming languages ($n=381$). The last course in the study was computer networks, an advanced upper-division elective ($n=132$).

To measure pre-college experience, we administered a beginning of term survey in each course. The survey was comprehensive, but for the purpose of this study we focus on the two questions pertaining to students' prior CS experience:

- **Q1:** How much experience had you had with programming/computer science before taking your first CS course at [this university]?

A1: Select one: *Absolutely none* / *A little* / *A fair amount*

- **Q2:** Please rate your level of experience with CS/programming before taking your first CS course at [this university].

A2: Select all that apply: *Took AP CS* / *Took another (non-AP) CS course in high school or middle school* / *Took one or more CS courses at a college or university* / *Self-taught one or more programming languages (not including HTML)* / *Participated in a CS club or extracurricular activity* / *Learned HTML (either in a class or on your own)* / *Participated in a summer CS experience (e.g. summer academic program or camp)* / *No experience* / *Other (please specify)*

"AP CS" refers only to the AP CS A course, which is similar to our introductory CS course(s), as the AP CS Principles course was launched two years after our data collection.

The survey also asked students whether or not they were currently a declared CS major, and whether they entered our university directly from high school or as a transfer student. In our current study we included only students who entered our university directly from high school because community college preparation varies so widely. Examining the performance of transfer students is an important topic for future work.

To quantify student performance, we used final course grades, persistence in the CS program (measured by whether participants in the lower division courses went on to complete Advanced Data Structures, the first required upper-division CS course), and pre-test scores from two courses. The pre-tests, given online as part of the first homework assignment in two terms each of CS2 and Advanced Data Structures, were designed to test the prerequisite knowledge for each course. The instructors of these courses modeled the pretest questions on quiz questions that would have been given toward the end of the preceding courses. Students were given full credit for attempting the quiz, regardless of their score.

Students' final course grades, pre-test scores, and whether or not they took Advanced Data Structures by spring 2017 were associated with their survey results using de-identified coding.

RESULTS

SELF-ASSESSED EXPERIENCE AND GRADES

To give context for effects of pre-college CS experience on students' grades in their courses, we first present the proportion of students reporting different levels of experience via survey question Q1. Figure 1 shows the breakdown of these experience levels by course. Overall, 18% of study participants report having a fair amount of pre-college CS experience, 43% report having a little experience, and 39% report having absolutely no experience.

We studied the relationship between self-assessed pre-college experience level and final course grades, using a 4.33 point grading scale ($A+=4.33$, $A=4$, $A-=3.67$, etc)¹ in each of the courses in our study. The results are shown in Figure 2. We analyzed the differences between the mean grades for students with the different experience levels using an ANOVA.

Figure 1: The distribution of self-reported pre-college experience levels (Q1) by course in our study. Columns are normalized to 100%. Numbers at the top give total n for each course across all terms in our study.

Figure 2: Mean grades for each course by pre-college experience level. Error bars show standard error. Asterisks by the course name indicate significant difference between experience levels' means: * = $p < 0.05$; ** = $p < 0.01$.

Figure 2 shows clear differences in average grades between the students in the three experience level groups for several courses, with more pre-college experience generally leading to a higher average grade. Differences are more pronounced in the

¹ Although our institution awards 4 grade points for both an A and an A+, we wished to distinguish the higher level of performance afforded by an A+ in our study.

lower division and mid-level courses, but the Advanced Data Structures course clearly shows that more experience is correlated with higher performance, while in the Programming Languages course reporting at least some experience a higher mean grade than reporting no experience at all. In courses that are less aligned with the type of experience students typically acquire pre-college (i.e. basic programming experience), the relationship between reported pre-college experience level and grade is not as straightforward. For example, in the low-level programming course the same grade trend exists, though the difference between the groups is not significant. In the upper division programming languages course, having a lot of pre-college experience does not seem to be much of an advantage, while having none is still a disadvantage. A similar trend appears in the (very challenging) computer networks course, though these differences are not statistically significant.

The results in this section partially confirm and partially reject hypothesis H1 when measuring student success by course grades: In many courses students with more pre-college experience do better than their less experienced peers, and (contrary to H1) these differences appear to persist into some advanced CS classes.

SPECIFIC EXPERIENCES AND GRADES

Survey question Q2 asked students to report on specific CS-related activities they engaged in before starting at our university. Based on our judgment of the different activities, we grouped them into two categories: formal experiences (AP CS, other high school classes, and college classes) and informal experiences (CS clubs, summer programs, self-taught programming, and learning HTML). We did not code the “Other experience” responses, and omitted students who selected only this option, as well as students who did not select any options, from our analysis (4% of respondents).

We then compared grades in each course between the following pairs of student groups using t-tests: (1) students with each of the specific types of pre-college experience vs. students who did not have that specific type of experience (but may have had any of the other pre-college experiences or no experience), (2) students with only informal pre-college experience (i.e. no formal or “other” experience) vs. students with no pre-college experience (i.e. those who selected “No experience” in response to Q2), (3) students with any kind of formal experience (possibly in addition to informal experience) vs. students with no pre-college experience. Of the many combinations in (1), we include only AP CS vs. no AP CS in our presentation of the results for three reasons: it was a fairly common pre-college experience with 18% of our respondents reporting this experience (and around 24% excluding the introductory courses), it plays a central role in the education community’s conversations about K-12 CS preparation and CS access, and, as shown below, it was robustly associated with higher average grades.

Table 1 shows the percentage of students in the courses in our study that had each of the different pre-college experience categories. For clarity and brevity, this table aggregates courses

with similar experience distributions as follows: Mid-level courses are CS2, low-level programming, and Advanced Data Structures; Upper-div courses are Programming Languages and Computer Networks. We observe that the proportions of experiences in Table 1 bear similar patterns to those shown in Figure 1. Students appear to be good at determining their own level of prior experience, especially classifying no experience compared to some/a lot of experience.

Course group	APCS	Any Formal	Only Informal	None
CS1+CS1.5 (no exp)	8%	26%	23%	47%
CS1 (exp)	43%	62%	17%	18%
Mid-level	23%	41%	21%	36%
Upper-div	24%	36%	16%	46%

Table 1: The percentage of students in each course set (see text) who reported having each pre-college experience type.

In our analysis of grade differences by experience type, we find differences at all levels for most pairs of experience types. Figure 3, plots the differences between the mean course grades of each pair for CS2 and above. We focus on CS2 and beyond to look for lasting effects of pre-college experience, though as expected we also found grade differences at the introductory level.

Figure 3: Differences between mean course grades for three different pairs of pre-college experiences. Data points indicated by • exhibit no significant difference; data points indicated by an × show significant differences ($p < 0.05$).

In every course except low-level programming (whose content is less likely to be familiar even to students with pre-college programming experience), students with any kind of pre-college experience out-perform students with no pre-college CS experience, though this difference is not always statistically significant (particularly for informal experience). When the experience is a formal high-school or college course, the difference is even larger (and statistically significant). Moreover, the magnitude of the differences between those with experience and those without experience does not shrink as the course level gets higher.

AP CS on its own turns out to be particularly related to higher average grade in nearly all courses. Students who took AP CS in high school out-perform students who did not take AP CS in high

The Persistent Effect of Pre-College Computing Experience on College CS Course Grades

school on average, even when the latter group includes students who have had other kinds of formal and informal experience.

The data confirm hypothesis H2 when measuring student success by course grades: certain types of pre-college CS experience, even in isolation, are linked with significant improved average performance in CS classes at all levels. In particular, students who take AP CS have significantly higher mean course grades than those who don't, even though students who do not take AP CS may have other in-class or informal pre-college CS experiences.

ADVANCING THROUGH THE CS PROGRAM

Beyond course grades, persistence in the CS program is a measure of students' performance. To examine the impact of pre-college CS experience on this persistence, we analyzed whether students in the lower-division classes in our study (all CS1 and CS1.5 courses, CS2, and low-level programming) eventually went on to take Advanced Data Structures—the gateway course to our upper division—by the end of spring 2017, two years after we collected our survey data.

For this section of our study, we restricted our attention to only students who were declared CS majors. We wanted to remove students who were never intending to continue with computer science and for whom continuing to the advanced classes would not be a reliable measure of success. A limitation, however, is that this data misses students who intend to declare a CS major but have not yet been able to (because our CS major is closed, student interested in switching into the major from another major on campus have to apply to do so after completing most lower-division classes).

We examined retention rates for the following groupings of students: (1) those who took AP CS vs. those who did not take AP CS, (2) those with formal (in-class) pre-college CS experience vs. those with only informal pre-college CS experiences vs. those who reported no pre-college experience on survey question Q2, and (3) those reporting “a fair amount” vs. “a little” vs. “absolutely no” precollege experience on survey question Q1. We used Chi Squared to test for significant differences between groups' continuation rates.

Results are shown in Table 2. When we consider students in all of the lower division courses in our study, most of the time students with experience continue on to the upper division in

higher rates than students with no experience, though these differences are significant only in the case of AP CS vs. no AP CS. When we limit our attention to students who have made it past the introductory course(s), we see that these differences mostly disappear. Controlling for grades in these courses did not change these observations.

These data mostly do not support either hypothesis H1 or H2 when measuring student success by retention in the program: Pre-college CS experience mostly does not lead to statistically significant differences in continuation rates. However, when looking just at continuation of students after the introductory programming sequence, the data do support hypothesis H2: CS majors who took AP CS are more likely to persist after their lower division courses to the Advanced Data Structures course than those who did not.

PRE-TEST SCORES

Significant differences in the course grades of students with different levels of experience. We now examine whether differences are also observable in low-stakes pretest grades. Table 3 shows students' average pre-test scores for each of the groupings we used in the previous section. In general, we see that there is no significant difference in pre-test scores between groups with different pre-college experience types and levels. The significant difference between experience types for CS2 is small (0.4 points out of 10) and favors students with informal experience only. Thus, the pre-test data partially reject and partially confirm hypothesis H1: students with more pre-college experience do no better than their less experienced peers on pre-tests in either the lower-division or the upper-division class.

DISCUSSION

THE IMPORTANCE OF AP CS

Our results highlight the critical importance of AP CS at the high school level. At 18% overall, AP CS was the most commonly reported type of formal prior experience, and second-most common prior experience (after “self-taught one or more languages,” at 21%). Moreover, students who took AP CS had statistically significantly higher average course grades in nearly all

	Took AP CS			Experience Type				Experience Level			
	Yes	No	<i>p</i>	Formal	Inf. only	None	<i>p</i>	A lot	A little	None	<i>p</i>
All LD	92%	85%	< 0.05	88%	89%	83%	0.12	90%	87%	85%	0.30
No Intro	95%	93%	0.62	92%	95%	94%	0.56	92%	94%	93%	0.76

Table 2: Percent of students who go on to take Advanced Data Structures from different groups with different pre-college experience. All LD=students in all lower division courses; No Intro = students in CS2 and low level programming only.

	Took APCS			Experience Type				Self-reported Experience Level			
	Yes	No	<i>p</i>	Formal	Inf. only	None	<i>p</i>	A lot	A little	None	<i>p</i>
CS2	8.7	8.5	0.24	8.4	8.8	8.5	< 0.05	8.6	8.4	8.5	0.50
Adv. Data Structures	7.3	7.2	0.56	7.3	7.2	7.1	0.51	7.3	7.3	7.0	0.41

Table 3: Average pre-test scores (out of 10) for CS2 and Advanced Data Structures by different experience groups.

the mid- and upper-level courses studied than those who did not take AP CS. Of course, some of this difference is likely due to correlation: Students with access to AP CS might have access to higher quality education overall or have higher motivation to learn CS. Regardless, taking AP CS may help boost a student's feeling of "leading the pack" when they reach college and mix with peers who did not take AP CS. This impression may be reinforced in early programming courses, where (at our university) the curriculum is very similar to that of AP CS.

Our results apply only to AP CS A. Future studies must consider whether similar gains exist for students who take AP CS Principles.

LOW-STAKES VS. HIGH-STAKES ASSESSMENT

The effects of confidence may also be relevant in the differences we saw in results on course grades compared to pre-test scores. In high-stakes assessments, such as in-class exams, students with less pre-college experience might suffer from a lack of confidence and thus under-perform relative to their true understanding and ability. Further work with more thoroughly validated instruments given in a low-stakes context would be needed to conclude that performance levels are indeed the same on low-stakes assessments. Yet our results suggest that grade differences might be due to more than just gaps in understanding.

PRIOR EXPERIENCE AND RETENTION

We saw only a slight difference in retention rates, mostly after the introductory sequence, between students with and without pre-college experience. Indeed, we saw that a sizable fraction of students in classes at all levels report having no precollege CS experience. It is concerning that these students are underperforming their experienced peers into the upper division.

The performance gap at the upper division could be exacerbated by students going through the wrong introductory track. While most students follow placement guidelines, there are some students who seem misplaced: either students with a lot of experience who are in the two-course sequence for those with no experience, or those with no experience in the one-course introduction (see Figure 1). For some, this may be due to scheduling or preference for a particular professor. In addition, our university puts pressure on students to graduate quickly, and we hear that some students with no experience elect to take the single-quarter introductory sequence in hopes of reducing their time to graduation. Our results underscore the importance of working with students, particularly those with no experience, to ensure that they take the time they need to get caught up.

Interestingly, both Figure 1 and Table 1 show that the percentage of students in the advanced upper division courses reporting no pre-college experience is actually higher than that percentage of students reporting no pre-college experience in the mid-level courses (CS2 through Advanced Data Structures). Because our data was collected over only 4 quarters (just over a year), the students in the more advanced classes are mostly

students who entered our university earlier than the students in the mid-level courses. Students' pre-college experience is likely changing (and increasing) over time, though a confounding effect may be that those further into college may not remember some of their informal experiences pre-college.

THREATS TO VALIDITY AND LIMITATIONS

The first threat to the validity of this work is common to many large-scale data analysis studies: The results presented here are all correlations and do not necessarily imply causation. It is likely that many students with pre-college experience also have access to higher-quality secondary education, and more computing resources outside of school. They may even have a greater interest and motivation for learning CS. So we cannot conclude, for example, that simply getting more students into AP CS will fix the performance gap in higher education (though we believe it will certainly help).

Second, we did not ask students whether they had access to the different pre-college experiences, so we cannot distinguish between students who were interested but were unable to participate and those who chose not to participate. It is possible that these two groups have different outcomes in their college classes, but we do not have the data to treat them separately.

Third, we present results from courses at many levels through the CS program, but our data was gathered from a single snapshot in time. Thus, we are not tracking individual students' trajectories, rather reasoning about mostly disjoint groups at different levels. It is possible that there are differences in the student groups over time that could account for some of our results.

Finally, these results come from a single university at a single point in time. Although our large public research university similar to many others in the United States, its student body and courses are not exactly like those at other institutions, and certainly very different from universities in other countries, with different target student populations (e.g. HSIs or HBCUs) or of different styles (e.g. liberal arts colleges). More work is needed in different contexts before we can really generalize these results.

CONCLUSION: ACTION REQUIRED TO LEVEL THE PLAYING FIELD

Our results provide evidence that we need to do more both in high school and in higher education to level the playing field for computer science majors. Our work provides strong support for the ongoing efforts to provide high-quality computer science education to all high school students.

At the college level, longer introductory sequences for students with no experience likely help. Yet, programs must be designed to ensure that students are given the time to complete these sequences without unreasonably extending their time to degree, and students must be properly placed into these sequences. Beyond longer course sequences, students might benefit from supplementary enrichment activities offered in paral-

The Persistent Effect of Pre-College Computing Experience on College CS Course Grades

lel with their courses. For example, if students' upper-division grades are suffering because they did not learn the lower-division content as deeply, we can provide supplemental "refreshers" courses that help students re-learn basic material without forcing them to retake a whole course.

We must also do more to close any potential confidence or belonging gap between students with and without pre-college computing experience. Affinity groups for students who feel unsure of their preparation, or explicit growth mindset education could help these students meet their full potential. Instructors must be taught techniques to frame classroom climate so that it does not disadvantage students with less experience [3]. Universities might even consider offering sections of courses beyond the introductory level restricted to students with limited pre-college computing experience.

The result that more experience is linked with better outcomes is not surprising, yet we as university educators are often blinded by the star student who had no pre-college experience who becomes the best student in the program. This work provides a push for us to acknowledge that this student is the exception, and to take steps to ensure that CS education is truly open for everyone to succeed. ♦

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1339335.

Advertise with ACM!

Reach the innovators and thought leaders working at the cutting edge of computing and information technology through ACM's magazines, websites and newsletters.

◆◆◆◆

Request a media kit with specifications and pricing:

Illia Rodriguez
+1 212-626-0686
acmm mediasales@acm.org

References

1. Ahadi, A. and Lister, R. Geek Genes, Prior Knowledge, Stumbling Points and Learning Edge Momentum: Parts of the One Elephant? *ICER* (2013), 123–128.
2. Alvarado, C., Lee, C.B., and Gillespie, G. New CS1 Pedagogies and Curriculum, the Same Success Factors? *SIGCSE* (2014), 379–384.
3. Barker, L.J., O'Neill, M., and Kazim, N. Framing Classroom Climate for Student Learning and Retention in Computer Science. *SIGCSE* (2014), 319–324.
4. desJardins M. and Littman, M. Broadening Student Enthusiasm for Computer Science with a Great Insights Course. *SIGCSE* (2010), 157–161.
5. Dodds, Z., Libeskind-Hadas, R., Alvarado, C., and Kuenning, G. Evaluating a Breadth-first CS 1 for Scientists. *SIGCSE* (2008), 266–270.
6. Ericson, B. AP Data for the United States 1998–2016; <http://home.cc.gatech.edu/ice-gt/595>. Accessed 2017 August 20.
7. Guzdial, M. Exploring Hypotheses About Media Computation. *ICER* (2013), 19–26.
8. Hagan, D. and Markham, S. Does It Help to Have Some Programming Experience Before Beginning a Computing Degree Program? *ITICSE* (2000), 25–28.
9. Haungs, M., Clark, C., Clements, J., and Janzen, D. Improving First-year Success and Retention Through Interest-based CS0 Courses. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education* (New York, NY, USA, 589–594: ACM, 2012); doi: <http://dx.doi.org/10.1145/2157136.2157307>
10. Holden, E. and Weeden, E. 2003. The Impact of Prior Experience in an Information Technology Programming Course Sequence. *CITC4* (2003), 41–46.
11. Kirkpatrick, M. S., and Mayfield, C. Evaluating an Alternative CS1 for Students with Prior Programming Experience. *ACM SIGCSE Technical Symposium on Computer Science Education* (New York, NY, 333–338: ACM, 2017); doi: <http://dx.doi.org/10.1145/3017680.3017759>
12. Kori K., Pedaste M., Leijen, A., and Tõnisson, E. The Role of Programming Experience in ICT Students' Learning Motivation and Academic Achievement. *International Journal of Information and Education Technology* 6, 5 (2016), 331–337.
13. Margolis, J. Stuck in the Shallow End: Education, Race, and Computing. *MIT Press* (2008).
14. Margolis, J. and Fisher, A. Unlocking the Clubhouse: Women in Computing. *MIT Press* (2003).
15. Marling, C. and Juedes, D. CS0 for Computer Science Majors at Ohio University. *SIGCSE* (2016), 138–143.
16. Martin, A., McAlear, F., and Scott, A. Path Not Found: Disparities in Computer Science Course Access in California High Schools. Technical Report. *Level Playing Field Institute* (2015).
17. McGill, M.M., Decker, A., and Settle, A. Undergraduate Students' Perceptions of the Impact of Pre-College Computing Activities on Choices of Major. *Trans. Comput. Educ.* 16, 4, Article 15 (June 2016), 33 pages.
18. Porter, L., Lee, C.B., and Simon, B. Halving Fail Rates Using Peer Instruction: A Study of Four Computer Science Courses. *SIGCSE* (2013).
19. Porter, L., and Simon, B. Retaining Nearly One-Third more Majors with a Trio of Instructional Best Practices in CS1. *SIGCSE* (2013), 165–170.
20. Robins, A. Learning edge momentum: a new account of outcomes in CS1. *Computer Science Education* 20, 1 (2010), 37–71.
21. Rogerson, C., and Scott, E. The Fear Factor : How It Affects Students Learning to Program in a Tertiary Environment. *Journal of Information Technology Education* 9 (2010), 147–171.
22. Simon, B., Kinnunen, P., Porter, L., and Zazkis, D. Experience Report: CS1 for Majors with Media Computation. *ITICSE* (2010), 214–218.
23. Taffiovi, A., Campbell, J., and Petersen, A. A Student Perspective on Prior Experience in CS1. *SIGCSE* (2013), 239–244.
24. Tew, A.E., Fowler, C., and Guzdial, M. Tracking an Innovation in Introductory CS Education from a Research University to a Two-year College. *SIGCSE* (2005), 416–420.
25. erner, E.H., Albert, E., Turner, R.M., and Latour, L. Retaining Majors Through the Introductory Sequence. *SIGCSE* (2007), 24–28.
26. Wang, J., Hong, H., Ravitz, J., and Ivory, M. Gender Differences in Factors Influencing Pursuit of Computer Science and Related Fields. *ITICSE* (2015).

Christine Alvarado

Department of Computer Science and Engineering
University of California, San Diego
9500 Gilman Drive, Mail Code 0404, La Jolla, CA 92093
cjalvarado@eng.ucsd.edu

Gustavo Umbelino

Department of Computer Science and Engineering
University of California, San Diego
9500 Gilman Drive, Mail Code 0404, La Jolla, CA 92093
gumbelin@ucsd.edu

Mia Minnes

Department of Computer Science and Engineering
University of California, San Diego
9500 Gilman Drive, Mail Code 0404, La Jolla, CA 92093
minnes@eng.ucsd.edu

Copyright held by the owner/author(s). Publication rights licensed to ACM. \$15.00
DOI: <http://dx.doi.org/10.1145/3159450.3159508>