ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Paper ID #41403

Progress in K-12 Computer Science Education: are Engineering Students Being Left Behind?

Dr. Catherine Molloseau, Grand Valley State University

Born and raised in Southeast Michigan. Attended Michigan Tech University, earning B.S. in Metallurgical Engineering. Earned M.S. and Ph.D. at Carnegie Mellon University in Materials Science and Engineering. Over 20 years industrial and academic experience.

Progress in K-12 Computer Science Education: Are Engineering Students Being Left Behind?

Catherine Molloseau, Ph.D.
Department of Engineering
Grand Valley State University
Grand Rapids, MI 49503
Email: mollosec@gvsu.edu

Abstract

Computer science education in the K-12 setting has evolved in the last decade with the development of new outreach programs, courses, and national standards. This ongoing effort has focused on both increasing the number and diversity of students pursuing computer science related fields as well as to help better prepare students for success where historically, it has been demonstrated that as many as a third of students fail introductory collegiate-level computer science courses worldwide. To determine if the implementation of these initiatives is having a positive impact on engineering students' performance in a first-year programming course, a correlational research study at a mid-size Michigan public university was completed over a three-semester period beginning in 2022. Students were surveyed regarding their prior experience with computer science at the beginning of the semester, and student scores on the first laboratory practicum and final course grade were recorded. The data demonstrates that nearly sixty percent of students had no prior experience with computer science and withdrew from the course at nearly double the rate as students with AP experience. For those that did complete the course, a Welch's t-test demonstrates that inexperienced students still passed at nearly the same rate as students with prior experience. The high withdraw rate suggests that engineering students that enter a first-year programming course without having completed an advanced computer science course in high school may be at a significant disadvantage to students that have had this opportunity. In addition, although nearly seventy percent of students attending public high schools in the state of Michigan have access to at least one computer science course, it needs to be better understood as to why more students planning to pursue engineering in college are not enrolling in these courses.

Background

Computer science education has been a topic of discussion since the mid twentieth century when the first computers were introduced, and computer science became recognized as a field of study at the university level.[1] At the birth of computer science education and throughout the last seventy years, there has been continued debate on what should be emphasized in the curriculum, theory versus application.[1,2] Some experts argued that a strong understanding of mathematics in the sense of using symbols, logic and number theory in the development of algorithms was the most important while others such as those in the computer industry believed training a workforce to develop the hardware and software using engineering principles was key.[1] In a study by Hromkovic and Lacher[3] that focused on the historical connections of human thinking and computer science education, it was proposed that in order to continue to advance science and technology emphasis should be placed on "better understanding mathematics and languages in their development,

understanding computer science research as a research instrument in science and humanities, and understanding our technical world and being able to control and develop it". Today, there is a general consensus among computer science educators that computer science education should focus on computational problem-solving (including theory) while also addressing the topics of social impacts and interdisciplinary needs such as in bioinformatics and cybersecurity in a technologically evolving world.[1,4-6] In doing so, there will be more opportunity to innovate in a manner that benefits society.

During the twenty-first century, the topic of pedagogical approaches to computer science education particularly at the K-12 level has emerged. Movements such as the CSforALL campaign that began in 2013 have created an explosion in the development of curriculum to reach a diverse set of learners.[7] The introduction of AP Computer Science Principles (AP CSP) by the College Board in 2016 represented the largest launch of an AP course in its history.[8] In 2017, in collaboration with industrial and government partners, the Computer Science Teachers Association (CSTA) published a set of national standards to be used in the instruction of computer science in K-12 schools.[4] In addition, nine policies were developed by leading organizations to guide states in implementing computer science education. In 2020, it was reported that all 50 states have adopted at least one of these policies while many have adopted 5 or more.[9] In the Midwest, the state of Michigan officially adopted the CSTA standards in 2019 and as of 2022, the following four policies: 1) define computer science and establish rigorous K-12 computer science standards, 2) establish dedicated computer science positions in state and local education agencies, 3) allocate funding for computer science teacher professional learning and 4) allow computer science to satisfy a core graduation requirement.[10] In addition, 46% of public high schools (serving 70% of the student population) within the state offered at least one computer science course as of 2022. Given this trend, it brings to question whether these initiatives are having an impact on student performance in computer science post K-12 such as in a collegiate setting. In particular, are there course offerings or approaches that lead to more success for engineering students in a first-year programming course? The goal of this study is to develop a better understanding of this situation with a focus on the implementation of computer science initiatives at the state level such as in the state of Michigan.

In addition to developing curriculum for K-12 students, previous studies have been conducted to better understand influences on student performance in introductory level collegiate computer science courses where pass rates have been historically below 70%.[11,12] In a study by Burgiel et al.,[13] 2,871 students from 115 U.S. institutions were surveyed about their past experience in computer science (primarily high school). In addition, student SAT and/or ACT scores were obtained along with final course grade in the programming class. Overall, it was found that although high standardized math test scores correlated with higher scores in computer science, the best indicator for student success was when students had extensive opportunities to practice coding. In addition, students reported various pedagogical techniques used in the classroom such as group work, discussions of application of computer science, preparation work for standardized tests, and peer instruction, however, these techniques were not shown to have a significant impact. In a similar study by Umapathy et al.[12] involving 193 college students from a U.S. public institution in the southeast, it was found that when students were surveyed in regard to their preferred method of instruction in a computer science course, being provided the opportunity to practice coding while learning about the concepts and calculations involved was

more favorable than having to memorize or otherwise be formally tested. In yet another study, Chen et al.[14] investigated the influence of a student's first programming language and attitude towards computer science on success in introductory collegiate computer science courses. A total of 10,203 students from 118 two-year and four-year institutions in the U.S. participated, and overall it was found that 57% had prior programming experience (with 5% having graphical experience vs. 52% with textual based experience). There was general consensus that any prior experience was helpful and contributed to maintaining a positive attitude about the subject. There was not enough evidence to suggest that a specific style of programming (graphical vs. text-based) or language (eg. C vs. Python vs. Java) was most beneficial, however, there was some evidence that younger grades (6-10 years old) may benefit more from graphics-based programming. In each of the above-mentioned studies, providing students opportunities to practice coding was the leading indicator for success, but there was no specific focus on a student's college major, only that the students were enrolled in an introductory computer science course.

Two studies focused more on understanding the types of students that succeed in a first-year collegiate level programming course. Reynolds et al.[15] used attributes such as declared major (Computer Science (CS), Information Systems (IS), or non-majors), High School GPA, Class Rank, and ACT scores and compared them to the rate of success measured by the final course grade for 722 students at a mid-size public university. They found that although CS majors had a statistically significant higher rate of success than IS majors, there was no significant difference between CS and non-majors. In addition, the CS majors tended to have a higher average ACT score than other majors. Surprisingly, the non-majors tended to have higher average High School GPA, and ACT Math and Science scores compared to CS and IS majors. An important factor in this study is that they also used data from students that repeated the course, and the data demonstrated that these students had a higher rate of success in all cases. In a study by Chen et al.,[16] the focus was on determining if math and science courses (none, regular or AP) play a role in student success in a first-year programming class. The study involved 9,418 students from a stratified random sample of 118 U.S. colleges and universities. In addition to academic information, demographics and other background information were collected. Their study found that after compensating for student background, both AP Calculus and AP CS had a positive effect on student success in college CS and that a regular Calculus class also had a positive effect for students although students that only took regular Calculus would need to achieve high grades to make up for not taking AP CS.

Previous studies have contributed to the understanding that prior experience does influence a student's rate of success in a first-year collegiate level programming class, however, most of these studies were completed before or shortly after the launch of AP CSP, a course designed to be more accessible to a diverse set of learners than its previous counterpart, AP CSA. In addition, none of the studies looked specifically at another category of students that are typically required to complete a computer science course, namely engineering majors. The aim of this study is to determine if increased access to computer science in K-12 institutions (such as through the launch of AP CSP) is having a positive impact on the success of engineering students in a first year applied programming course. To do so, a correlational research study was conducted over the course of three semesters using data from a mid-size 4-year public university in the state of Michigan. Student prior experience with computer science along with performance throughout the course

was tracked and analyzed to determine if there are any statistical trends that indicate the K-12 computer science movement is having a positive impact on student performance and retention.

Method

Institution

All data was collected from a mid-size public comprehensive university in the state of Michigan. This regionally accredited university serves approximately 19,000 undergraduate students and 3,000 graduate students. The institution has an engineering department with more than 40 full-time faculty members serving approximately 2,000 students. Within the engineering department, students may earn a bachelor's degree in electrical, computer, mechanical, biomedical, product design and manufacturing, or interdisciplinary engineering. As part of the first-year sequence, all engineering students are required to complete and pass (grade of C or higher) an introductory applied programming course in C. Pre-Calculus is a prerequisite for the course. Final grades are assigned based on a combination of interactive reading assignments and laboratory activities as well as lab practicums, weekly take home quizzes and mid-term and final exams.

Participants

All students enrolled in the required applied programming course in C were invited to participate in the study. As this study focused on the effect of initiatives in the state of Michigan, only data from students that graduated from a Michigan high school was included. In addition, the study did not collect data on demographics such as age, gender, or race as this is not the focus of the study, and the population of some groups was projected to be too low to avoid individual identification.

Instruments and Measures

All students that volunteered to participate were provided a survey at the start of the semester that addressed the following topics:1) most recent type of school attended, 2) year of graduation from high school, 3) place of graduation (state or country), 4) programming experience prior to college, and 5) programming languages used. In terms of measuring performance, the scores on the first lab practicum as well as the final course grade were collected.

Assumptions and Limitations

This is a correlational research study that commenced over three semesters. The programming course used as a tool for measurement had up to six different instructors for a given semester. The instructors used shared resources as well as common assessments and met weekly to discuss the course. Although there is a possibility that the instructor's own teaching style may have influenced student performance, it was assumed to be insignificant. As part of this study involved student participation via survey, it is possible that student response was not accurate. In addition, due to its voluntary nature, participation may not reflect the full population of students enrolled in the course for a given semester. There are also other factors that can affect a student's performance such as course load, personal work schedule, participation in a sport or other activity, or a student's

overall health and wellness. These factors cannot be controlled but were considered as a possible limitation when analyzing the data and drawing any conclusions. Finally, this study involved only one institution in the state of Michigan and results may not represent all engineering students enrolled in an introductory programming course within the state. It was chosen to focus on one institution, however, to limit additional variables such as those noted above. Future studies involving all engineering institutions within the state may lead to a more thorough understanding of the impacts of state initiatives on student performance.

Results

This is a correlational research study, and data was collected over three semesters from 2022 to 2023 where a total of 172 of the 405 enrolled students (42%) volunteered to participate. After review, 35 participants were disqualified from the study due to either having previously taken the course or graduating from a high school outside the state of Michigan.

Initial Survey Results

Data from the initial survey is summarized in Table 1 and Figures 1-3 below. According to Figure 1, 59% of the students had no prior computer science experience before entering college while 26% reported having some level of experience such as through a semester/trimester elective, FIRST Robotics or summer camp (Table 1). The remaining 15% of students reported having taken an AP Computer Science course (note: students that reported both an AP course and any other experiences were placed in the AP category). Figure 2 demonstrates that students that have had exposure to computer science prior to enrolling in the course have worked with a variety of programming languages with Java being the most prevalent (27) and JavaScript, Python, and Scratch/Block-Based languages as the next four most popular languages (20-23). In terms of the student's high school graduation year, Figure 3 shows that 74% of the participants graduated in 2021 or 2022.

Table 1: Summary of student computer science experience prior to college

Programming Experience	# students
None	81
AP CSP/AP CSA	20
trimester/semester elective	22
Other	17
FIRST Robotics	12
Girls Who Code	0

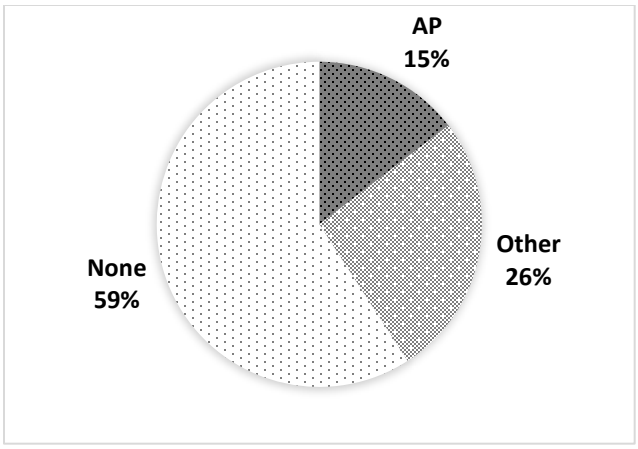

Figure 1: Distribution of student participants based on computer science experience prior to college (N = 137)

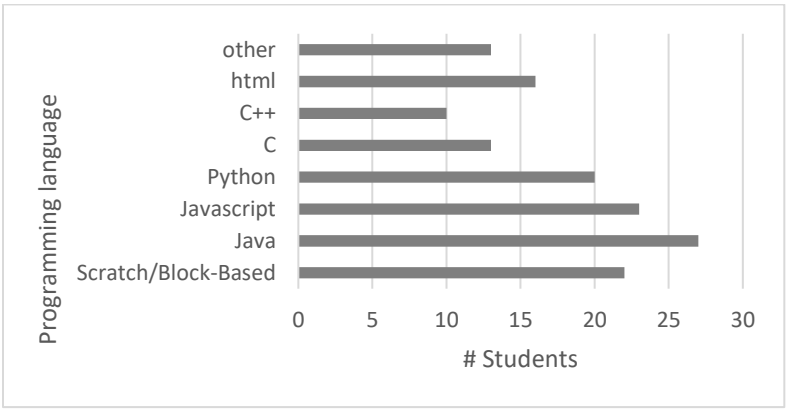

Figure 2: Distribution of programming languages students worked with prior to course enrollment

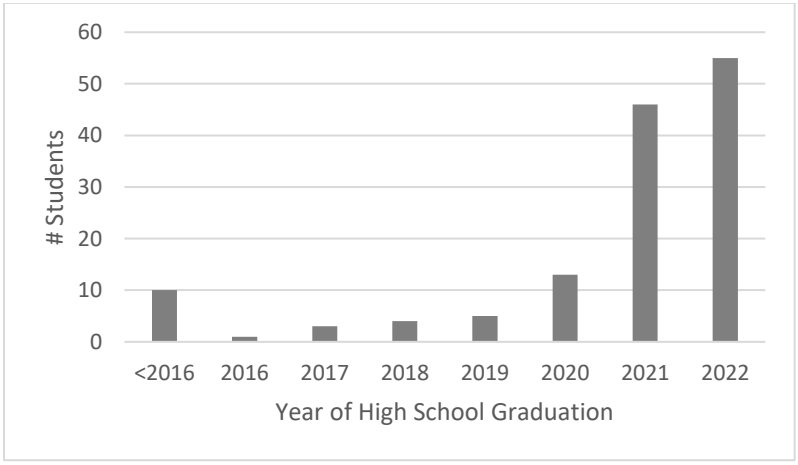

Figure 3: Distribution of student year of high school graduation (N = 137)

Course performance results:

During the study, 33 students withdrew from the course (3/20 from the "AP", 7/36 from the "other" and 23/81 from the "none" category). The percentage withdrawal for each category is represented in Figure 4 and demonstrates that the withdraw rate for students with no prior coding experience was 30 to 50% higher than those with some level of experience. The scores for the first practicum and final course grade were therefore collected for the remaining students that completed the course (N = 104). Figure 5 (a and b) represent scores based on the year of high school graduation and prior computer science experience (note: students that graduated in 2016 or earlier were assigned the year 2016). Although there was no significant trend observed with the year of high school graduation, it was found that all students that had taken an AP computer science course graduated in 2020 or later. In looking at the entire population, 80 students (77%) passed the first practicum while 79% (82 students) passed the course.

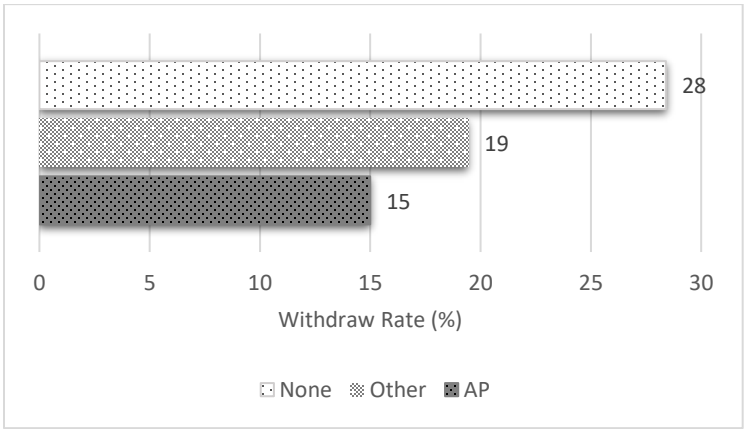

Figure 4: Rate of student withdraw (%) based on prior experience

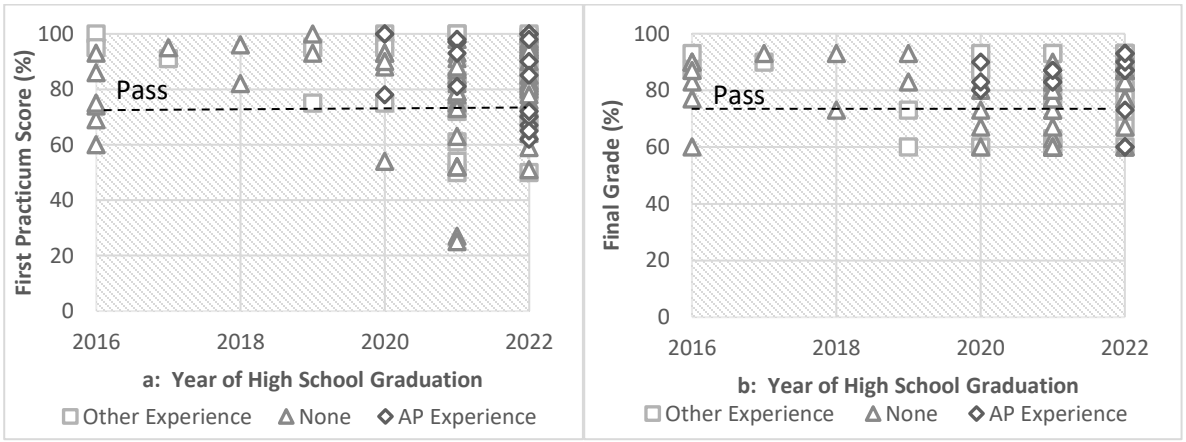

Figure 5: Relationship between score on first practicum (%) and year of high school graduation (a) and final grade (%) (b) based on computer science experience prior to high school (N = 104)

To compare the distribution of the scores per experience category, box plots as well as Welch's one-tail t-tests for unequal variances were generated; see Figures 6 and 7 and Tables 2 and 3. The box plots demonstrate that for both the first practicum score and final course grade, the mean for each category was above the passing score of 73%. In terms of whether the students with AP or other experience had a higher mean score than the students with no prior experience, the t-test demonstrates that in all cases the p-value is greater than 0.05 and therefore the difference is not statistically significant.

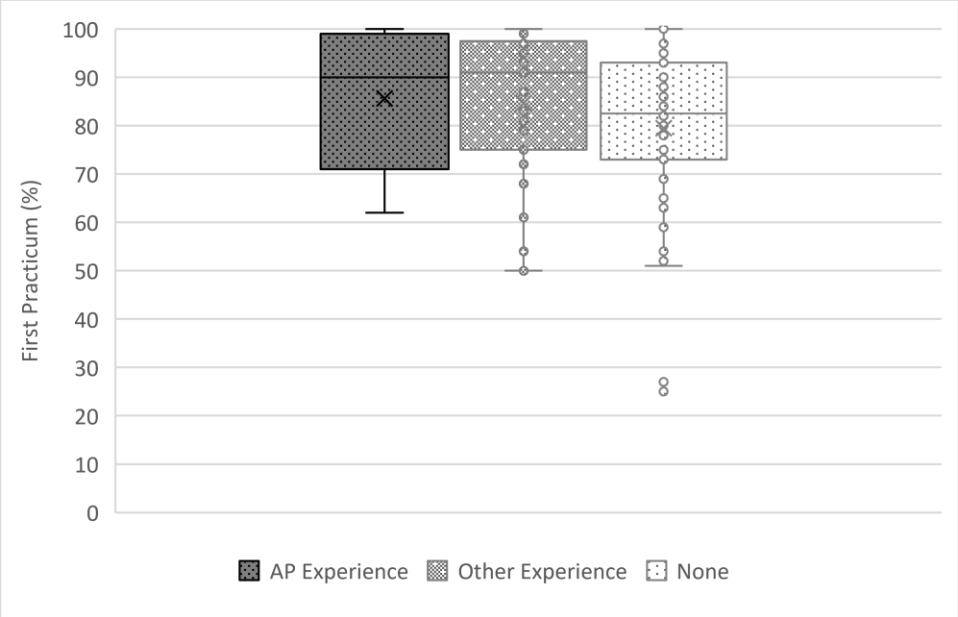

Figure 6: Box plots demonstrating the distribution of scores on the first practicum (%) as a function of prior computer science experience

Table 2: Welch's t-test results for first practicum score (Note: hypothesized mean difference = 0)

	AP	Other		AP	None		Other	None
Mean	85.6	84.7		85.6	79.4		84.7	79.4
Variance	198.5	247.9		198.5	291.3		247.9	291.3
Observations	17	29		17	58		29	58
df	37			31			60	
tStat	0.2206			1.517			1.413	
P(T<=t) one-tail	0.4133			0.0697			0.0814	
t Critical one-tail	1.687			1.696			1.671	

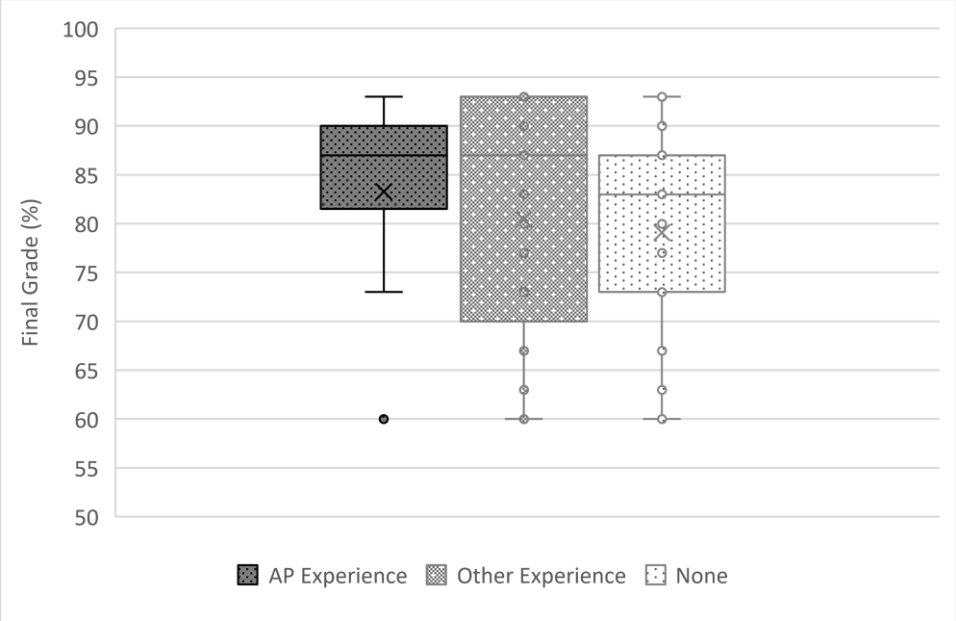

Figure 7: Box plots demonstrating the distribution of final course grades (%) as a function of prior computer science experience (Note: Final Grades were converted from letter grades to percent based on a standard 4.0 grading scale. A score of "A" was assigned 93% and an "F" was assigned 60%).

Table 3: Welch's t-test results for final course grade (Note: hypothesized mean difference = 0)

	AP	Other		AP	None		Other	None
Mean	83.3	80.4		83.3	79.1		80.4	79.1
Variance	102.7	154.8		102.7	120.5		154.8	120.5
Observations	17	29		17	58		29	58
df	39			28			50	
tStat	0.8539			1.4827			0.4939	
P(T<=t) one-tail	0.1992			0.0747			0.3118	
t Critical one-tail	1.685			1.701			1.676	

Discussion

The goal of this study was to determine if the initiatives to increase access to computer science education at the K-12 level are positively affecting the performance of first-year engineering students in an applied programming course. Data was collected over three semesters during 2022 and 2023 with over 40% of students enrolled volunteering to participate. Of the 104 students that met the criteria for the study (Michigan high school graduates and first-time enrollment in the course), 72% graduated from high school in either 2021 or 2022. Despite 46% of public high schools (serving approximately 70% of the state's population) offering a computer science course

as of 2022, it was found that only 41% of the participants received any computer science training prior to college. Although the Welch's t-tests did not demonstrate that the average final grade for students that completed the course for the three test groups (none, other, and AP) was different, the lack of experience may have contributed to the high withdraw rates as shown previously in Figure 4. For further analysis of this potential relationship, the pass rate by level of experience was determined assuming students that withdrew would not have passed the course, Figure 8. The data suggests that students that had taken an AP computer science course had a significant advantage over both students with other types of experience (elective, extracurricular activity, etc.) and those with no experience. This data supports the study by Chen et. al.[16] in that students that completed an AP Computer Science course were more likely to be successful in an introductory collegiate level computer science course than those that had not.

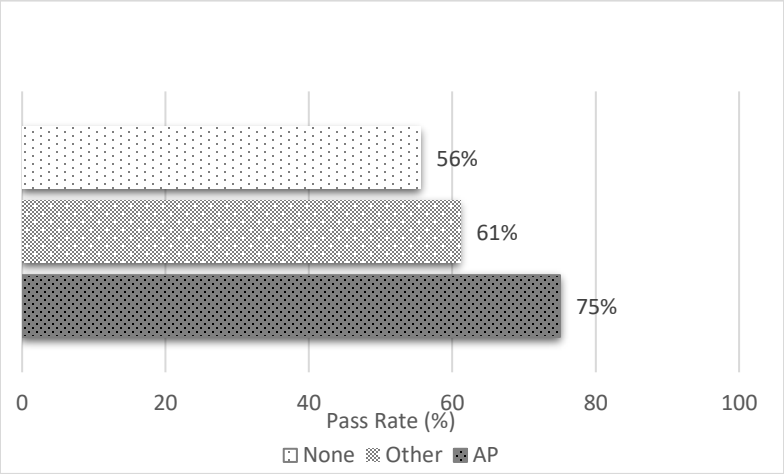

Figure 9: Overall course pass rate (%) based on the level of prior experience (N = 104)

If successful completion of an AP level computer science course (CSA or CSP) is an indicator of success for engineering students enrolled in a first-year programming course, then more high schools in the state of Michigan need to offer this level of course. As of 2020, only 237 schools (31% with AP programs) offered an AP computer science course with more offering AP CSP than AP CSA.[17] Because the state of Michigan no longer requires an endorsement in computer science, teachers of any discipline may receive training through organizations such as Code.org which could help increase the number of schools offering a course.[10] It may also help if the state develops an official plan or requires high schools to at least offer a foundational computer science course such as several other Midwest states do. Yet another strategy would be for college counselors to encourage students planning to pursue an engineering degree to take an AP computer science course. Students can even count this or other approved computer science courses towards their fourth math credit for graduation. Any of the above initiatives may help further increase student access to computer science thus helping them to be better prepared for a collegiate level course.

Another factor tracked was the programming languages students reported having worked with prior to enrolling in the study. Although Java was the most reported, there was no clear indication that experience with a given language correlated with student success. However, it was found that 33 students had experience with two or more languages. Of these students, 82% passed the course with 50% earning an 'A' grade. In the study by Chen et. al.[14] involving over 10,000 students from

more than 100 institutions, it was also found that the type of language did not have an impact on student performance. More data that focuses on the impact of the number of programming languages on student success may lead to a better understanding.

During the initial survey, students were asked about the types of experiences they had with computer science prior to college. Given that FIRST robotics is well-known for promoting STEM, the state of Michigan has the most FIRST FRC teams in the country with 458 in 2022, and this state has a unique district and state competition structure that spans the entire state,[18] it is surprising that 1) less than ten percent of the students reported having participated with FIRST, and 2) of those students, only 50% passed the course used for this study. It is possible that although the number of teams are high, the average number of students per team is relatively low and/or students that responded did participate in FIRST but did not gain computer science training through the experience. Another computer science related outreach program geared towards promoting female students, Girls Who Code, had zero reports of participation. Although gender was not a focus of this study, this was an unanticipated observation. A follow up study to explore low participation in these outreach programs and whether they affect student performance in an introductory collegiate level computer science course may be warranted.

This study has demonstrated that completion of an AP computer science course is the strongest indicator of a student's success in an introductory collegiate level computer science course for engineers. In addition, it was found that only 15% of students that participated in the study gained this experience. Although these results may be significant, it must be noted that there are some limitations. First, the study involved only one university in the state of Michigan and survey data was collected voluntarily over a three-semester period with 74% of the students graduating from high school in 2021 or 2022. Although the survey participation rate represented 42% of those enrolled, it is possible that the data does not fully represent the population. Also, since the state of Michigan adopted the national computer science standards in 2019, it is possible that the effects on student performance may not be fully realized for several years. Finally, there are additional factors that may have affected a student's performance such as overall course load, outside commitments such as work or extracurricular activities, health and wellness, and the disruption students may have experienced during the global pandemic. Although this cannot be controlled, future studies that focus on these aspects may shed further light into a student's performance.

Conclusion

The purpose of this study was to determine if recent initiatives in computer science education at the K-12 level are having a positive impact on the performance of engineering students in a first-year applied programming course. This study demonstrated that a student's lack of experience increased their likelihood of withdrawing from the course and as past studies have shown, students that completed an AP level computer science course were more likely to pass. This study focused on students from one university that attended high school in the state of Michigan and given that almost sixty percent reported not having received any computer science education prior to enrolling in the course it is suggested that more efforts to increase access to these courses needs to be provided. In addition, further studies that involve students from more engineering programs within the state as well as continuing to gather data over the next three to five years may help to

develop a better picture of the state of computer science education for students pursing engineering majors in the state of Michigan.

Bibliography

[1] M. Tedre, S. and L. Malmi, "Changing aims of computing education: a historical survey," *Computer Science Education*, vol.28 no.2, pp. 158-186, 2018.

[2] C.V. Ramamoorthy, "Computer Science and Engineering Education", *IEEE Transactions on Computers*, vol. C-25, Dec., pp. 1200-1206, 1976.

[3] J. Hromkovic and R. Lacher, "The Computer Science Way of Thinking in Human History and Consequences for the Design of Computer Science Curricula", ISSEP, LNCS 10696, pp. 3-11, Springer International Publishing, 2017.

[4] Computer Science Teachers Association, "CSTA K-12 computer science standards, revised 2017", 2017. [Online]. Available: http://www.csteachers.org/standards. [Accessed Oct. 15, 2021].

[5] Team ISTE, "ISTE Announces New Computational Thinking Standards for All Educators", 2018. [Online]. Available: http://www.iste.org/explore/Press-Releases [Accessed Jan. 20, 22].

[6] M. Daugherty, V. Carter, and A. Sumner, "standards for technological and engineering literacy and STEM education", *technology and engineering teacher*, Feb., pp. 32-37, 2021.

[7] CSforALL, "About CSforALL", [Online]. Available: https://www.csforall.org/about/. [Accessed Dec. 10, 2021].

[8] College Board, "About AP Computer Science Principles", [Online]. Available: https://apcentral.collegeboard.org/courses/ap-computer-science-principles. [Accessed Jan. 10, 2022].

[9] Code.org, "2020 State of Computer Science Education Illuminating Disparities", 2020. [Online]. Available: https://advocacy.code.org/2020_state_of_cs.pdf. [Accessed Oct. 15, 2021].

[10] Code.org, "2022 State of Computer Science Education Understanding our National Imperative", 2022. [Online]. Available: https://advocacy.code.org/2022_state_of_cs.pdf. [Accessed Jan. 26, 2023].

[11] A. Vihavainen, J. Airaksinen, and C. Watson, "A systematic review of approaches for teaching introductory programming and their influence on success", In Proceedings of the 10th Annual Conference on International Computing Education Research, 2014, New York, NY, pp.19-26.

[12] K. Umapathy, A.Ritzhaupt, and Z. Xu, "College Students' Conception of Learning of and Approaches to Learning Computer Science", *Journal of Educational Computing Research*, vol.58 no.3, pp. 662-686, 2019.

[13] H. Burgiel, P. Sadler, G. Sonnert, "The Association of High School Computer Science Content and Pedagogy with Students' Success in College Computer Science", *ACM Transactions on Computing Education*, vol.20 no.2, Article 13, Apr., 2020.

[14] C. Chen, P. Haduong, K. Brennan, G. Sonnert, and P. Sadler, "The effects of first programming language on college students' computing attitude and achievement: a comparison of graphical and textual languages", *Computer Science Education*, vol.29 no.1, pp. 23-48, 2018.

[15] J. Reynolds, R. Ferguson, and D. Linville, “Be Careful What You Wish For: The Continued Limiting of Computing Majors”, In Proceedings of the EDSIG Conference, 2017, Austin, TX, pp.1-9.

[16] C. Chen, J. Kang, G. Sonnert, P. Sadler, “High School Calculus and Computer Science Course Taking as Predictors of Success in Introductory College Computer Science”, ACM Transactions on Computing Education, vol. 21, no. 1, Article 6, Dec., 2020

[17] Code.org, “Support K-12 Computer Science Education in Michigan”, 2020. [Online]. Available: https://code.org/advocacy/state-facts/MI.pdf. [Accessed Dec. 15, 2023].

[18] FIRST Robotics, “2022 FIRST in Michigan District”, 2022. [Online]. Available: https://frc-events.firstinspires.org/2022/district/FIM. [Accessed Dec. 15, 2023].