“Like Navigating A Dark Labyrinth”

Experiences of Minoritized Computer Science Students at Small Liberal Arts Colleges

Juno Bartsch
Computer Science
Virginia Tech
Blacksburg, Virginia, USA
junob@vt.edu

Christian Murphy
Computer Science
Columbia University
New York, New York, USA
cdm6@columbia.edu

Bob Edmison
Computer Science
Virginia Tech
Blacksburg, Virginia, USA
kedmison@vt.edu

ABSTRACT

To understand the experiences of minoritized students taking introductory Computer Science, we conducted a mixed-methods survey with 43 participants at two non-doctoral institutions. We found that minoritized students had decreased sense of belonging and persistence. First-generation and low-income students had significantly lower self-efficacy and reported feeling under-prepared. Students also mentioned access barriers, gender-based microaggressions, and a lack of identity-based community. Based on these findings, we make the following recommendations: implementing CS0, accessible course policies, recruiting and training diverse educators, and increasing collaboration across different types of universities.

CCS CONCEPTS

• Social and professional topics → Computer science education; CS1; Computational thinking; User characteristics; Race and ethnicity; Gender; People with disabilities

KEYWORDS

CS0, Self-efficacy, Minoritized learners, Neurodivergence, Transgender, Culturally Sustaining Pedagogy

ACM Reference format:

Juno Bartsch, Christian Murphy, and Bob Edmison. 2026 “Like Navigating A Dark Labyrinth:” Experiences of Minoritized Computer Science Students at Small Liberal Arts Colleges. In *Proceedings of 2026 Conference for Research on Equitable and Sustained Participation in Engineering, Computing, and Technology (RESPECT 2026). ACM, Chicago, IL, USA, 10 pages.* https://doi.org/10.1145/1234567890

1 Introduction

Minoritized learners are those who have been socially, politically, and/or economically disenfranchised by systems of power and privilege in our society [36]. These students are not merely “underrepresented,” they have been systematically excluded, and researchers have focused on their “deficits” rather than their assets [3, 17].

In this work, we highlight the experiences of many minoritized students in CS, including Women, Transgender and Gender Non-Conforming (TGNC), Black, Indigenous, and People of Color (BIPOC), First-Generation and/or Low-Income (FGLI), Neurodivergent/ND (Autistic, ADHD, Dyslexic, etc.), Disabled, and students with Mental Health Conditions (MHCs). While members of any of these groups may have vastly different experiences in the classroom, minoritized identities can result in compounding and intersectional experiences of oppression [17]. Addressing their needs is essential to diversify our field.

In 2023, non-doctoral institutions gave 80% of their CS degrees to men and 51% to white students, slightly higher rates than institutions overall [30]. We have little data on the retention or experiences of Non-Binary, Disabled, FGLI, and other minoritized groups in CS [25, 30]. Including these students introduces new perspectives, creativity, and innovation. However, it also means that we must adapt CS courses to embrace different ways of learning, knowing, and being. In higher education, this work starts with CS1.

2 Theoretical Framework

In this work, we intentionally center the voices of students who hold one or more minoritized identities. We draw from critical methodologies such as Critical Race Theory and Critical Disability Theory. Specifically, we believe that the issues raised in this work are the result of systemic issues, not individuals, departments, or institutions. We use an asset-centered approach to our analysis, meaning that we view diversity and differences in opinions as essential to our work. In our research, teaching, and learning environments, we believe in intentionally including people with different lived experiences and perspectives.

Our research was conducted at two predominantly white small liberal arts colleges. Due to the small size of these programs, the number of minoritized students can be heavily shaped by students feeling included by the community. Presenting the intersectional experiences of students with minoritized identities poses risks such as unintentional identification of participants. However, this should not justify overlooking the experiences of these minoritized students, nor should the relative lack of research activity compared to

larger universities. We believe we should be using the power, prestige, and wealth of these private colleges to actively combat inequities. Additionally, this research serves as a call to action for researchers at larger universities to partner with smaller colleges and their local communities to share strategies for inclusion, create forums for minoritized people to share their experiences, and foster a learning environment that embraces many ways of learning and knowing.

3 Research Questions

This work is necessary because broadening participation efforts have not been well studied at small private colleges, and minoritized students' needs are not being met [25]. Issues like preparedness for CS1 [33, 35], student retention [30], and faculty shortages [17, 34] are pervasive in CS. Given this state of institutional failure, we propose the following research questions (RQs):

1. How is self-efficacy impacted by identity, experience, and support? What factors lead students to continue in CS?
2. What factors deter students from retaining the CS major? How can they be mitigated?
3. How might minoritized students experience CS1 differently?
4. How can we improve the course content, structure, delivery, policies, and/or resources to support minoritized students?

Our contributions include highlighting the experiences of under-researched minoritized groups, identifying faculty-student connections as a key mitigator of the self-efficacy gap, and offering recommendations for liberal arts and general contexts. We hope this will lead to systemic changes in computing education and research.

4 Positionality Statement

Our research team is committed to amplifying the voices of minoritized students. Collectively, we have lived experience with minoritized identities including being Transgender, Disabled, living with Mental Health Conditions, being First-Generation, and being Low-Income. These experiences heavily influence how we understand structural inequities and exclusionary practices within the field of computing. We also acknowledge our positions of social and institutional privilege, being white researchers at wealthy and renowned institutions. These identities may shape our interpretations of the data, which is why we took deliberate steps to be reflexive and foreground our participants' perspectives.

Having an undergraduate and member of teaching faculty on our team at the time of this research, we understand that this creates a power imbalance and privacy concerns for some students. This is why our research team included dialogue with community members and establishing trust with our participants as crucial steps in the research process. We commit to representing their experiences as faithfully as possible, in their words, with minimal bias or assumptions.

5 Research Design and Methods

This study used a mixed-methods survey, which was developed with prior literature and feedback from a small pilot group. A mixed-methods design was chosen to gather trends about the minoritized student population (e.g. self-efficacy), while allowing students to self-describe their experiences.

Students were considered eligible if they had completed at least two weeks of an XYCo CS1 course. This period marks the last date that students can freely add or drop a course. The recruitment materials emphasized that the study was for minoritized students but was open to all. Participants were recruited using the CS department mailing lists, flyers, word-of-mouth, and snowball sampling.

Students were asked about the workload, difficulty, grading, and support systems in CS1. They were also given the option to list their identities, discuss their personal experiences, challenges, learning needs, and their opinions on a hypothetical CS0 course. The self-efficacy scale was adapted from the New General Self-Efficacy Scale [5]. Self-described demographic factors were coded into categories to ensure anonymity. Race was coded as white, Asian, or BLI (Black, Latine, and Indigenous). Similarly, we used Women, Men, and TGNC as gender categories. Disabilities were categorized as psychological (MHCs and ND), physical, and learning disabilities. FGLI self-identification was closed-ended, as students widely identify with that label in the XYCo.

The study received IRB exempt status from CY (protocol #24-048). We were unable to offer compensation to the participants. All students were given informed consent and encouraged to give honest feedback, with no obligation to participate. All data were anonymized before analysis.

5.1 Background and Related Work

5.1.1. CS0 and CS1.

The typical entry point to the CS major is CS1, an introduction to programming in languages like Python or Java. This experience is intended to teach critical thinking and problem-solving skills that are necessary to create an algorithm [28]. While programming plays a significant role in modern CS, the design and logic of the code are just as important. Learners may benefit from alternate entry points that do not rely on "learning by doing."

Alternative entry points include CS0 and CS1X. We refer to CS0 as a "breadth-first" approach that surveys the discipline as a whole, prioritizing computational thinking and problem-solving with some exposure to programming [20]. CS0 would be designed for students with little experience and non-majors and has been shown to increase success rates in CS1 as well as interest in the CS major [7]. Another approach is to create sections of CS1 for students without prior experience, such as Cohoon and Tychonievich's CS1X

[8]. This course covered the same content as CS1 but significantly improved the enrollment and retention of minoritized students. Alternative entry courses bring students to similar, more consistent levels of preparedness before entering CS2 [18, 33].

5.1.2. Retention and Self-Efficacy.

Researchers have had concerns about retention in CS for decades, particularly for minoritized students. In 2007, it was reported that 33% of all CS1 students would drop or fail, which slightly improved to 28% in 2017 [2]. While this figure may not be as high as other STEM (Science, Technology, Engineering, and Mathematics) subjects, CS is not more inherently difficult to teach or learn [2, 4]. We should be asking how we can set our students up for success, not why they are failing.

Changes like re-explaining concepts, reducing anxiety around asking questions, and offering support for students without prior experience can improve student outcomes [16, 33, 35]. We can also increase their persistence by building their self-efficacy, or “judgment of their ability to execute tasks or achieve mastery” [12]. We can improve interest and performance in the field by fostering a sense of belonging, indicating that students feel valued and respected [22]. Additionally, we can utilize research-backed practices to improve pass rates and encourage interactivity, including the flipped classroom, problem-based learning, pair programming, project-based learning, feedback surveys, and instructional transparency [16, 22, 26, 28, 32].

5.1.3. Representation and Culture.

Broadening participation efforts have typically focused on Women and racially “underrepresented minorities” [17, 25]. Representation is a poor metric for student experiences because it is dependent on time and context [17]. For example, Asian students are better represented in CS than other BIPOC but have similar experiences with discrimination and low self-efficacy [23, 27]. Research has focused on building minoritized students' resilience rather than making necessary systemic change [6]. Initiatives to support intersectional diversity, often organized by multiply minoritized groups like Women of Color, have been dismantled in the name of “including everyone” [17].

Many faculty feel unprepared to address discrimination when it occurs [6]. They may not have training, guidance, or support from their departments in working with minoritized students. However, they can incorporate Cultural Humility and Culturally Sustaining Pedagogy (CSP) into their practices. Cultural Humility asks us to critique our practices and beliefs in the interest of creating connections and continued learning [6]. In the classroom, this can look like including minoritized identities in problem sets [24], partnering with community organizations for course projects, and encouraging discussions on ethical and sociopolitical impacts of technology [19]. In CS, these practices have mostly been applied in K-12 and informal learning environments [9, 19, 29]. Future work is needed to expand beyond individual professors or departments to address systemic injustices, such as in admissions, which impact access to CS [6]. Ultimately, we have a collective responsibility to promote diversity, equity, inclusion, and accessibility (DEIA) in academia.

5.2 The Bi-Co

To answer our RQs, we considered the case of College X (CX) and College Y (CY). These small liberal arts colleges in the U.S. Northeast make up the XYCo consortium. Both serve about 1400 undergraduate students on wealthy suburban residential campuses. CY is a Historically Women’s College (HWC) and CX is coeducational. While students in the XYCo may take courses or major in CS at either college, CY students are mostly Women or TGNC.

Like many undergraduate-only CS programs, both colleges have struggled to maintain enough professors to meet increasing demand [34], which resulted in the elimination of CX's CS0 course around 2010. In addition to CS1, CX offers a combined CS1+2 course for experienced students, and CY has a graphics-applied CS1X course [37]. Additionally, departments such as biology and chemistry offer CS+X programming courses (termed “affiliate courses” here). Although CS1, CS1X, and affiliate courses are considered “equivalent” entry pathways, they use different languages (Python, Java, Processing) and cover different concepts. CS students can seek support from peer educators (TAs and peer tutors) at no cost. At the time of this study, each year CX offered 1 section of CS1 with 40 students, CY had 2 sections with 25 students, and each offered 1 additional section (CS1+2 with 15 students and CS1X with 20 students, all estimates approximate).

5.2.1. The Strike and Project Reboot.

CX and CY have legacies of student activism on their campuses, including the Fall 2020 Strike led by BIPOC students [31]. The majority of students attended teach-ins and rallies rather than classes during this time. Additionally, a CY CS faculty member and students developed a set of goals for the department to work toward. These included increasing sense of belonging and retention for underrepresented students. They identified numerous challenges to achieving these goals, including students’ preparation before college, how time-consuming CS courses are, Imposter Syndrome, the lack of representation in the department, inaccessible course policies, and the lack of communication and student voice in departmental decisions. Despite these issues being brought to the CY CS department, no action was taken.

Similarly, at CX, a student organization called XCode compiled recommendations for the CS department entitled Project Reboot [13]. XCode mentioned issues including a faculty shortage, visiting instructors teaching CS1, an uneven

distribution of experience in CS1, untrained TAs, and CS1 over-enrollment which led to students abandoning the major. XCode argued that discrepancies in topics covered in the intro sequence, such as recursion, could be addressed with a "streamline set" of topics in CS1/2. They also argued for large lecture-style CS1 sections, which faculty believed would worsen outcomes. Finally, XCode argued for training peer educators, creating a TA handbook, and minimally subjective rubrics. Having untrained peer educators disproportionately impacts first-generation students, who are less likely to understand the "hidden curriculum"—including the social norms of higher education and the value of resources like office hours [14].

6 Data Analysis

All statistical tests were performed in Jamovi [15] using a significance level $\alpha = .05$ and non-directional hypotheses. The research team created a codebook and used thematic analysis for the short-answer responses. After reading through the data, initial codes were created inductively (e.g., access barriers under "learning needs") and deductively (e.g., students lamenting the "toxic" culture). These were used to develop statements based on shared student sentiments. For example, participants were marked in agreement with the statement "I found the TAs helpful" if they said, "the TAs helped me get the information MUCH clearer" (P24) or "I went to enough TA hours to get all my questions answered" (P25). These estimates are conservative, as the authors only indicated agreement when responses were clearly in favor. Percentages were calculated out of all participants who answered at least one short-answer question (n = 36). Some limitations to our approach include students opting out of the mailing lists, skipping some questions, and the lack of feedback from students who dropped CS1 (at least 90.7% of our sample had completed CS1 or were enrolled during the study).

7 Findings and Discussion

In this study, we collected data from a diverse group of 43 minoritized students who took CS1 in Fall 2020-Spring 2024. We are unable to report an accurate response rate due to the lack of demographic data on minoritized groups.

For the class years 2024-2027, CY students made up 18.6%, 9.3%, 14.0%, and 4.7% of respondents. Similarly, CX students made up 14.0%, 4.7%, 23.3%, and 11.6% of respondents. Most of the participants in this study (62.8%) were CS majors or minors, with the remaining participants being another major, undecided, or undeclared. Of the students who reported their CS1 course, 44.2% took CS1 at CY, 27.9% took CS1 at CX, and 18.6% took CS1+2 at CX. The other identity-based demographics are as follows:

- Race/ethnicity: White (58.1%), Asian (25.6%), Black, Latine, and Indigenous (11.6%), prefer not to say (4.7%)
- Gender: Women (44.2%), Transgender and Gender Non-Conforming (30.2%), Men (20.9%)
- LGBT: Yes (51.2%), No (37.2%), Unsure (7.0%)
- FGLI: No (69.8%), Yes (25.6%), Unsure (2.3%)

A chi-square test of association found significant correlation between BIPOC and FGLI identities, $\chi^2(1, N = 40) = 8.03$, $p = .005$. Additionally, 20.9% of respondents identified as Disabled and 9.3% were unsure or questioning. Of those who felt comfortable with disclosure, 72.7% had psychological disabilities, 36.4% had physical disabilities, and 9.1% had a learning disability. More than half of students had multiple disabilities. Many students also reported Neurodivergence, including ADHD (18.6%), Autism (11.6%), and Dyslexia/ Dyscalculia/ Dysgraphia (1.5%). MHCs like anxiety (54.8%) and depression (42.9%) were prevalent; only 38.1% had no MHCs.

In the following subsections, we present findings from both quantitative and short-answer responses, divided into four themes: Who feels ready for CS1?, When does rigor become gatekeeping?, Who is in your corner?, and, Who gets to belong in CS?. Each theme covers general student sentiments, impacts on minoritized groups, as well as recommendations to improve the way we teach intro CS.

7.1 Who feels ready for CS1?

Our respondents expressed varying levels of prior experience in CS and motivations including interest in the major, general education requirements, and career applicability. However, students from each of these groups expressed frustration with CS1—non-majors who "don't want to be programmers" (P16), students wanting a course "catered to students with no CS experience" (P43), or CS majors preparing for the "difficulty spike" (P18) in CS2. Students from each of these backgrounds raised concerns about preparedness, which is troubling given that most students coming into CS1 had very little or no prior experience in CS (41.9%). This was closely followed by those with some experience (40.0%). Only 18.6% of our participants reported significant prior CS experience, which almost perfectly overlapped with the CS1+2 students. In the XYCo, there are no math or CS prerequisites for CS1, meaning students have varied previous experience.

Next, we investigated the impact of prior experience on time completing programming assignments. Students with significant, some, and little programming experience spent about 3-5, 4-6, and 5-7 hours per week on assignments, $F(2, 24.1) = 3.62$, $p = .042$, with a mean difference of 2.17, $p = .033$. Notably, 20.9% of participants reported spending 8 or more hours per week on their programming assignments. If students spend too much time on assignments instead of reading their textbook and understanding the concepts, this prevents them from effectively learning the material. This impact compounds, making it more difficult to complete future assignments, especially for students with jobs, sports, and other extracurricular obligations.

We also measured student self-efficacy as an average of the following scales relating to intro CS (reverse-coded as necessary): I cannot be successful, I received enough support to do well, I am less capable than my peers, It took me more time to understand concepts than my peers, and I felt confident in my ability to overcome challenges. There was high internal consistency among the items, Cronbach's α = .89, and the average score was 3.38 (SD = 1.19). On average, self-efficacy was slightly higher than neutral.

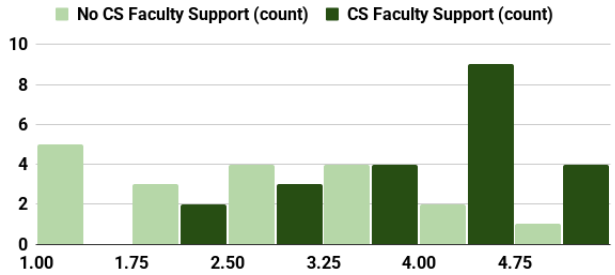

Figure 1: Self-efficacy for students who self-reported having (dark green) or not having (light green) the support of one or more CS faculty.

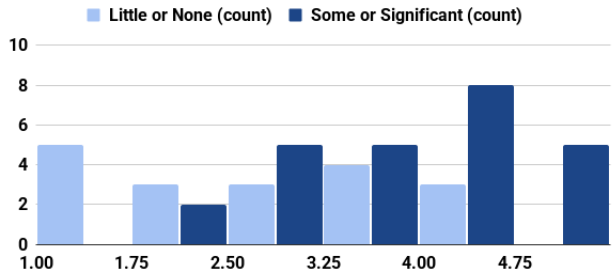

Figure 2: Self-efficacy for students with little or none (light blue) versus some or significant (dark blue) CS experience before CS1 (self-reported).

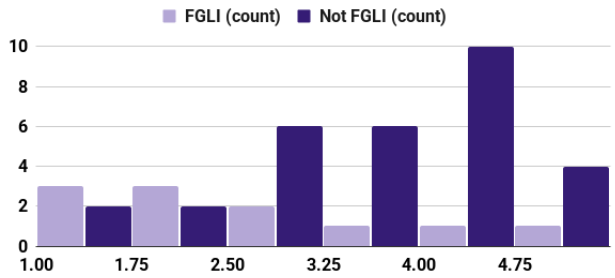

Figure 3: Self-efficacy for students who did (light purple) and did not (dark purple) identify as First-Generation or Low-Income (FGLI).

Self-efficacy was most strongly correlated with support from a CS professor, $t(39) = 3.74$, $p < .001$, Cohen’s $d = 1.17$. Self-efficacy was also significantly higher in students with prior experience, $t(41) = 3.51$, $p=.001$, Cohen’s $d = 1.09$. There was little difference observed between `some’ and `significant’ experience—this may change in larger controlled samples [23]. FGLI students had lower self-efficacy, $t(39) = 2.87$, $p = .008$, Cohen’s $d = 0.99$ (Fig. 1-3). These results indicate that students without faculty support, prior experience, and FGLI students have lower self-efficacy, are less likely to believe they are capable of success in CS, and less likely to persist in the major.

Self-efficacy was not correlated with other demographic factors, including gender, LGBT identity, Disability, and MHCs. Although BIPOC self-efficacy was 0.62 points lower than white students, $p = .110$, and ND students scored 0.58 points higher than neurotypicals, $p = 0.164$, the differences were not statistically significant. This could be due to sample size, as lower self-efficacy has been found in Women, Non-Binary, and Disabled students [22, 23].

7.1.1. Impacts on FGLI Students.

Short-answer responses from FGLI students confirm that math/CS preparation and self-efficacy were key barriers to their success. Participants like P20 found that their “biggest challenge was not having the math skills of my peers.” Similarly, P43 noted that being “the only student who came into the class with 0 coding experience... made me feel left out.” Two-thirds of students felt that they struggled more in CS due to being FGLI.

Students also mentioned situations where being FGLI made it challenging to fit in and communicate with professors. P16 said, “I was told to quit my job if it interfered with attending TA hours, despite that being my only source of income to sustain myself.” P5, P9, and P16 all discussed how their sense of belonging was reduced due to being FGLI. Specifically, P5 said they did not “feel... allowed to ask questions” due to their “ethnicity and income.” P9 said that CS1 “felt like navigating a dark labyrinth,” where “it was assumed that I had some level of understanding in a concept” when they asked questions, making it feel inaccessible to non-STEM students.

7.1.2. The Need for CS0.

CS1 seeks to provide a strong coding foundation but may not be the best fit for students without experience. In anticipation of this finding, we asked students for their opinions on a hypothetical CS0 course. Our participants expressed unanimous support for CS0, saying it “would be AMAZING!!” (P3). CS0 would begin to address inequities in prior math and CS experience [33], build student confidence [7], and improve retention of minoritized students [8]. In turn, this would create a more even distribution of experience in CS1, enabling instructors to focus on challenging concepts and improving the CS2 transition.

Drawbacks to implementing CS0 include course

availability and the major sequence length. As mentioned in Project Reboot, CS1 is often over-enrolled, and CS0 would likely take away a CS1 section. However, we cannot overlook the needs of minoritized students because of longstanding issues with instructional capacity. Instead, CS departments in the liberal arts should collaborate on curriculum development for affiliate courses and offer joint peer educator training. This will facilitate external educators to teach CS well and ensure students learn the same core principles in CS1 "equivalent" courses. The issue of major sequence length can also be addressed by clearly communicating course offerings, prerequisites, and the major pathway. These need to be accessible to deans and non-CS faculty, as major advisors tend to be assigned in students' second or third year. While small CS departments are acutely impacted by faculty availability, they do have the opportunity to work with students and staff more directly than at larger colleges.

7.2 When does rigor become gatekeeping?

It is unsurprising that many students struggle with the pace and workload of CS1, including 41.7% of our sample. More than half found CS1 time-consuming (57.5%), and 42.1% said it was more difficult than other introductory courses. When asked how they spent their time outside of class, students said they spent a median of 9 hours (IQR = 5.00) on CS1. This included 2 hours studying, 2 hours working with peer educators, and 5 hours programming. Although the median is in line with expectations for 3 credit-hour courses, the large IQR suggests variability in time spent programming.

Students reported feeling expected to work "10+ [hours] a week" (P43) on their assignments, suggesting they may be unnecessarily arduous, improperly scaffolded, and/or unaligned with the learning objectives. Moreover, a significant portion of respondents (37.2%) reported that the content was "poorly taught" (P24) or not presented in a way they understood. They were especially frustrated when their efforts did not align with their final grades.

7.2.1. Impacts on Disabled and ND Students.

Disabled and Neurodivergent (ND) students made up 41.9% of respondents. Senior students were slightly overrepresented, which is likely due to access to diagnosis [3]. Many of these students detailed course policies and practices that impacted their experience in CS1. Some policies they benefited from included having "lecture/notes/examples... posted online" (P21), "having a full week to do assignments so they overlap" (P37), "the labs [being] open 24/7" (P37), and acceptance of late work. Some things students asked to be implemented included "asynchronous assignments/tests" (P21), "flexible extensions and grading policies" (P16), clearer and less vague instructions, assignments being "shorter/gone over step by step" (P5), as well as typed rather than handwritten feedback (P18). These policies could mitigate the impacts of executive dysfunction, sensory overwhelm, and information processing differences [3, 11]. Without equitable application of these policies across the department, it can lead to students being "ostracized" and "discriminated against... for attempting to use [their] accommodations" (P16).

Students listed Neurodivergence as a barrier to connecting with their peers and TAs. P18 said that being ND "affects my ability to outreach and [connect] with cohorts," and "the CS department could have had events or avenues for that." Similarly, P26 "rarely" chose "to collaborate with classmates." Communication barriers can prevent students from making connections and support systems that they need to persist in the major. Additionally, some students were frustrated by the lack of time for participation in lecture, questions being "immediately directed towards the TAs," and confusing rules about collaboration during assignments. Several students mentioned feeling reliant on the TAs as some professors were difficult to reach outside of class. While peer educators are an invaluable resource, Disabled and ND students deserve support and clear communication from CS faculty.

Students also addressed the impacts of CS1 on their mental health. P5 said, "mental health issues made it difficult to get to class every day, or get assignments in on time, and [CS professors] have been hit or miss in... [working] with students." This inconsistency between professors can significantly impact the well-being and academic records of students with MHCs. Similarly, P16 claimed that CS courses "have destroyed my physical and mental health," forcing their peers to "pull all nighters and [be] constantly on the verge of a breakdown." These responses highlight the urgency of implementing accessible policies and practices.

7.2.2. Accessibility by Default.

Accessibility should be considered in all course decisions, materials, and the environment. Accessibility is a mindset and involves intentionally including Disabled, ND, and students with MHCs (72.7% of our sample). While some of these students received accommodations and understanding, others were met with harassment and microaggressions. Research has shown that students benefit from equitable grading practices, accessible course materials, and disability awareness [3, 11].

CS departments should work with disability services offices to educate and support their faculty in implementing accommodations individually and at scale. For example, turning on captions can benefit everyone, especially students who do not have access to formal accommodations. More broadly, recognizing the strengths and diversity that these groups bring to CS can help students feel more comfortable disclosing these identities [3]. Maintaining rigor in an equitable course does not mean constructing an overwhelming workload or artificial grading curves. Instead, professors should create manageable challenges while supporting collaboration, collecting feedback, and encouraging learning from mistakes [4].

7.3 Who is in your corner?

Although support from a CS faculty member was essential for building student self-efficacy, 43.6% of students did not receive this support. In fact, 34.2% reported not having the support of any faculty, staff, or administration. Students relied heavily on peer educators, with 61.0% of students considering them a key part of their support structure in CS1. While most students felt supported by peer educators, some participants said they did not have enough chances to get questions answered, and P22 stated, “many TAs are not helpful.” This sentiment was supported by P27, a former TA, who claimed they did not get “a good enough rubric to grade fairly.”

7.3.1. Impacts on TGNC Students and Women.

Social support systems are especially impactful for minoritized genders in CS. Women are more likely to attend department social events [27], and the lack of these events led our participants to feel less belonging in the department. Although our survey did not mention sense of belonging, 20.0% of minoritized students felt it was a major barrier to their success. The findings in this study are somewhat unique given that about half of the participants attend CY, where Women and TGNC students constitute a significant majority. P19 appreciated the environment at CY because she had previously been “isolated and... harassed by one of the older guys,” on her high school robotics team until she “abandon[ed] trying to learn CS altogether.”

However, issues of gender-based discrimination in the XYCo were reported by Women at CX and TGNC students. P32 remarked that they were “often talked over by (cis)male classmates.” This was echoed by P16, who said that they “have been constantly misgendered in [their] CS courses, more than any other department.” Additionally, P40 felt their professor was dismissive of their ideas and pushed them “to do the easier assignment instead” because of their gender. These findings show that gender-based discrimination occurs in HWCs and coeducational environments. Therefore, professors in gender biased fields like CS must take an active role in addressing microaggressions and creating a more welcoming culture.

7.3.2. Recruiting and Training Diverse Educators.

Building connections with faculty requires students to feel safe sharing feedback, raising issues, and asking questions. At a departmental level, hiring compassionate, engaging faculty should be a top priority. Additionally, training these faculty on how to teach equitably, communicate, and practice reflexivity is essential to the long-term success of DEIA initiatives. Faculty should be held accountable but also supported and given mechanisms to build community. Priority and resources should be given to faculty initiatives that encourage the retainment and growth of minoritized communities [17]. Support should also be given to offset the invisible labor that minoritized faculty take on in supporting students with similar experiences [17]. Centers for teaching and learning may also provide opportunities for student or staff consultants to improve professors' engagement with students [10]. CS educators should strive to be collegiate and collaborative, with everyone working toward DEIA, not just a select few.

Peer educators also play a crucial role as a “bridge” between instructors and students [1]. At many institutions, they receive insufficient guidance to manage themselves, their role, and their connections with peers. Instead of learning on the job, peer educators could work with experienced tutors, practice teaching scenarios, learn about cultural barriers, and develop conflict resolution strategies [1]. Peer educators are an extension of the instructors that train them and must be similarly coached to combat bias and promote equity. This includes everything from the community they build with students to the rubrics they are expected to adhere to when grading. Peer educator training should reflect commitments to DEIA and relationship building with minoritized students.

7.4 Who gets to belong in CS?

In our findings, multiply minoritized students most frequently expressed despair, dejection, and hopelessness. In particular, BIPOC FGLI students felt overwhelmed with the workload, outside jobs, discrimination, and lack of connections with their peers. They discussed “work and family obligations” (P43) and the “financial burden [and] not having enough nutritious food” (P38) as critical roadblocks in their CS journey. Similarly, they felt there was “an expectation to have more exposure before arriving” (P5) which impacted some participants' mental health. P9 “felt like I was an outsider in [CS1] and at [College] as a whole” because CS1 was presented as “super easy,” and their peers with STEM experience did not seem to struggle. They also said:

> “I wasted a semester learning about all the ways in which I did not stack up to the students in my class... the rhetoric that prefaced the course made me feel less intelligent than the students around me.”

While there were no direct mentions of racial discrimination, BIPOC students were significantly less likely to continue in CS. In our sample, only 68.7% of BIPOC students continued in CS, compared to 88.0% of white students. Of the BIPOC students who took CS2, 54.5% expressed feeling unprepared for the content and/or workload, compared to only 13.6% of white students. Several BIPOC students who persisted cited faculty support, such as P19, who said, “I never had to reach out to anyone else because [Professor was] always there to support me.” However, many students were unable to find support, and 61.1% of minoritized students did not feel there was a community in XYCo CS for students with their identities.

7.4.1. Culturally Sustaining Pedagogy and Practices.

Some computer scientists believe in “race evasive” or “race

neutral" teaching approaches [29]. However, ignoring inequities reinforces discrimination and the lack of material support for minoritized students [29]. We must be intentional about the examples we use to teach computing concepts to ensure students feel represented and valued [24].

Some examples used in XYCo computing courses include a dataset on baby names separated by biological sex, having students print a "fortune" similar to the cookies popular at Chinese American restaurants, graphing the population of incarcerated individuals by race, and facial recognition algorithm accuracy by race and gender. In these examples, it is important to recognize and interrogate systemic biases as part of the learning objectives.

Incorporating Cultural Humility and Culturally Sustaining Pedagogy would move our field toward restorative justice. Institutional changes would need to follow, such as incorporating ethics and interdisciplinary studies into the curriculum, updating mission statements, starting working groups, maintaining systems of accountability, and including students in departmental meetings. These steps are designed to combat "sociotechnical oppression," create communities of care and joy, as well as expand culturally sustaining approaches with abolitionist and critical race theories [29].

8 Conclusion

With these findings and recommendations, we hope to enable CS educators to support students who have been historically overlooked in classrooms and research. Specifically, we found that the support of just one CS faculty member greatly improved self-efficacy and helped students persevere in CS. Even when these efforts are not supported by others in their department or university, faculty should strive to make their classrooms and offices welcoming toward minoritized students, especially as identity-based support and resources are being dismantled at some institutions [21].

By giving students alternate paths to entry, accessible courses by default, training CS educators, and building learning environments that embrace all identities, we can break down systemic barriers for minoritized students. We should encourage diverse approaches to teaching that drive student connections without compromising accessibility or rigor. In addressing the RQs, we found the following:

1. Self-efficacy is higher in students with CS faculty support and prior CS experience; it is lower for FGLI students. Student retention requires social support and a sense of belonging.
2. Factors contributing to retention include course structures, support systems, and sense of belonging. Instructors and pedagogies must respond to students' access and equity needs.
3. Minoritized students may have less preparation and motivation to continue in CS due to systemic inequities.
4. Solutions include CS0, accessible and Culturally Sustaining pedagogies, as well as recruiting/training diverse educators.

While this study had many meaningful findings, it was limited by sampling and offers a somewhat narrow view of the experiences of CS1 students. The demographic results are unique due to the demographics of CS in the XYCo. This partially explains the over-representation of white, Women, and TGNC respondents. Many of the analyses relied on coding self-described variables, which may not capture the full range of participant experiences. This was done to ensure that participants could express their identities without being restricted to non-representative categories, such as "other."

It is essential to continue collecting student feedback and understanding minoritized student experiences, from large-scale retention data to in-depth qualitative work. Additionally, we must understand the experiences of students who are minoritized in ways we have not always considered, including socioeconomic class and outside the gender binary. We currently do not have sufficient data to understand their retention or experiences at scale [25]. Care should be taken to recruit more BIPOC students and emphasize intersectionality, when sample size allows. We acknowledge that some of these results may not be directly transferable to large R1 universities, however, issues of inaccessibility, gender, and class-based discrimination are pervasive across institutions. Further studies are also necessary to understand the benefits and challenges of implementing the strategies proposed in this work. Finally, researchers could work to optimize combinations of introductory course options (CS0, CS1, CS1X, CS1+2, and affiliate courses) at colleges with varying capacities and student needs.

ACKNOWLEDGMENTS

This work would not have been possible without the immense labor and dedication of BIPOC students. We are incredibly grateful to Haverford College's Women of Color House, Black Students Refusing Further Inaction, and Black Student League, as well as the Core Bryn Mawr College Strike Collective. We also appreciate Deepak Kumar, Sorelle Friedler, Kelly Gavin-Zuckerman, and Jacob Chan for their feedback and support.

REFERENCES

[1] Grace Barkhuff, Ian Pruitt, Vyshnavi Namani, William Gregory Johnson, Rodrigo Borela, Ellen Zegura, Anu G. Bourgeois, and Ben Rydal Shapiro. 2025. Exploring the Humanistic Role of Computer Science Teaching Assistants across Diverse Institutions. In Proceedings of the 56th ACM Technical Symposium on Computer Science Education (SIGCSE '25). Association for Computing Machinery, New York, NY, USA, 67–73. https://doi.org/10.1145/3641554.3701861

[2] Jens Bennedsen and Michael E. Caspersen. 2019. Failure rates in introductory programming: 12 years later. ACM Inroads 10, 2 (2019), 30–36. https://doi.org/10.1145/3324888

[3] Valeria Borsotti, Andrew Begel, and Pernille Bjørn. 2024. Neurodiversity and the Accessible University: Exploring Organizational Barriers, Access Labor and Opportunities for Change. Proceedings of the ACM on

Human-Computer Interaction 8, CSCW1, Article 172 (2024), 27 pages. https://doi.org/10.1145/3641011

[4] Center for Research on Teaching and Learning. 2023. Reframing Rigor to Promote Equity in Teaching and Learning. https://crlt.umich.edu/blog/reframing-rigor-promote-equity-teaching-and-learning.

[5] Gilad Chen, Stanley M. Gully, and Dov Eden. 2001. Validation of a New General Self-Efficacy Scale. Organizational Research Methods 4, 1 (2001), 62–83. https://doi.org/10.1177/109442810141004

[6] Anita Chikkatur and Stephanie Valle. 2023. Rethinking “Damage-Centered” Research and Individual Solutions: Cultural Humility as a Framework to Increase Student Diversity in Undergraduate STEM Departments. International Journal for the Scholarship of Teaching and Learning 17, 1 (2023), Article 19. https://doi.org/10.20429/ijsotl.2023.17119

[7] Daniel C. Cliburn. 2006. A CS0 course for the liberal arts. In Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education (Houston, Texas, USA) (SIGCSE ’06). Association for Computing Machinery, New York, NY, USA, 77–81. https://doi.org/10.1145/1121341.1121368

[8] James P. Cohoon and Luther A. Tychonievich. 2011. Analysis of a CS1 approach for attracting diverse and inexperienced students to computing majors. In Proceedings of the 42nd ACM Technical Symposium on Computer Science Education (Dallas, TX, USA) (SIGCSE ’11). Association for Computing Machinery, New York, NY, USA, 165–170. https://doi.org/10.1145/1953163.1953217

[9] Jazmin Collins, Kami Otero De Owens, Vensan Cabardo, Sarah Hug, Raena Cota, and Enrico Pontelli. 2021. Culturally Sustaining Pedagogies in CS: Modern K12 Outreach and its Adaptability in a Global Crisis. In Research in Equitable and Sustained Participation in Engineering, Computing, and Technology (RESPECT). IEEE, Philadelphia, 1–5. https://doi.org/10.1109/RESPECT51740.2021.9620563

[10] Alison Cook-Sather. 2025. Students as Teachers and Learners Program. https://tli-resources.digital.brynmawr.edu/.

[11] Adrienne Decker, Stephen H. Edwards, Brian M. McSkimming, Bob Edmison, Audrey Rorrer, and Manuel A. Pérez Quiñones. 2024. Transforming Grading Practices in the Computing Education Community. In Proceedings of the 55th ACM Technical Symposium on Computer Science Education (Portland, OR, USA) (SIGCSE ’24). Association for Computing Machinery, New York, NY, USA, 276–282. https://doi.org/10.1145/3626252.3630953

[12] Jamie Gorson and Eleanor O’Rourke. 2020. Why do CS1 Students Think They’re Bad at Programming? Investigating Self-efficacy and Self-assessments at Three Universities. In Proceedings of the 2020 ACM Conference on International Computing Education Research (Virtual Event, New Zealand) (ICER ’20). Association for Computing Machinery, New York, NY, USA, 170–181. https://doi.org/10.1145/3372782.3406273

[13] HaverCode. 2022. Project Reboot. https://github.com/junob-vt/reboot/blob/main/ProjectReboot.pdf. Unpublished Manuscript

[14] Anthony Abraham Jack. 2019. The Privileged Poor: How Elite Colleges Are Failing Disadvantaged Students. Harvard University Press, Cambridge, MA.

[15] Jamovi. 2024. Jamovi (Version 2.3) [Computer Software]. https://jamovi.org

[16] David A. Joyner, Lily Bernstein, Maria-Isabelle Dittamo, Ben Engelman, Alysha Naran, Amber Ott, Jasmine Suh, and Abby Thien. 2020. Attitudinal Trajectories in an Online CS1 Class: Demographic and Performance Trends. In Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (Trondheim, Norway) (ITiCSE ’20). Association for Computing Machinery, New York, NY, USA, 335–341. https://doi.org/10.1145/3341525.3387429

[17] Julia Rose Karpicz, Tomoko M. Nakajima, and Justin A. Gutzwa and. 2024. Challenging Normalized Gendered Racism in Departmental Efforts to Broaden Participation in Computer Science. Journal of Women and Gender in Higher Education 17, 4 (2024), 357–379. https://doi.org/10.1080/26379112.2024.2373479

[18] Michael S. Kirkpatrick and Chris Mayfield. 2017. Evaluating an Alternative CS1 for Students with Prior Programming Experience. In Proceedings of the 48th ACM Technical Symposium on Computer Science Education (Seattle, Washington, USA) (SIGCSE ’17). Association for Computing Machinery, New York, NY, USA, 333–338. https://doi.org/10.1145/3017680.3017759

[19] Kevin Lin. 2022. CS Education for the Socially-Just Worlds We Need: The Case for Justice-Centered Approaches to CS in Higher Education. In Proceedings of the 53rd ACM Technical Symposium on Computer Science Education (Providence, RI, USA) (SIGCSE ’22). Association for Computing Machinery, New York, NY, USA, 265–271. https://doi.org/10.1145/3478431.3499291

[20] Nancy L. Martin. 2015. IT0: Discrete Math and Programming Logic Topics as a Hybrid Alternative to CS0. Information Systems Education Journal 13, 1 (2015), 30–44. https://files.eric.ed.gov/fulltext/EJ1137190.pdf

[21] Eddy Ng, Terrance Fitzsimmons, Mukta Kulkarni, Mustafa Bilgehan Ozturk, Kurt April, Rupa Banerjee, and Sara Louise Muhr. 2025. The anti-DEI agenda: navigating the impact of Trump’s second term on diversity, equity and inclusion. Equality, Diversity and Inclusion: An International Journal 44, 2 (2025), 137–150. https://doi.org/10.1108/EDI-02-2025-011

[22] Vidushi Ojha, Andrea Watkins, Christopher Perdriau, Kathleen Isenegger, and Colleen M. Lewis. 2024. Instructional Transparency: Just to Be Clear, It’s a Good Thing. In Proceedings of the 2024 ACM Conference on International Computing Education Research (Melbourne, Australia) (ICER ’24). Association for Computing Machinery, New York, NY, USA, 192–205. https://doi.org/10.1145/3632620.3671091

[23] Vidushi Ojha, Leah West, and Colleen M. Lewis. 2024. Computing Self-Efficacy in Undergraduate Students: A Multi-Institutional and Intersectional Analysis. In Proceedings of the 55th ACM Technical Symposium on Computer Science Education (Portland, OR, USA) (SIGCSE ’24). Association for Computing Machinery, New York, NY, USA, 993–999. https://doi.org/10.1145/3626252.3630811

[24] Trysten Scott Richard, Eliane S. Wiese, and Zvonimir Rakamarić. 2022. An LGBTQ-Inclusive Problem Set in Discrete Mathematics. In Proceedings of the 53rd ACM Technical Symposium on Computer Science Education (Providence, RI, USA) (SIGCSE ’22). Association for Computing Machinery, New York, NY, USA, 682–688. https://doi.org/10.1145/3478431.3499330

[25] Linda J. Sax, Chantra Nhien, and Kaitlyn N. Stormes. 2024. A Quantitative Methodological Review of Research on Broadening Participation in Computing, 2005-2022. In Proceedings of the 55th ACM Technical Symposium on Computer Science Education (SIGCSE ’24). Association for Computing Machinery, New York, NY, USA, 1182–1188. https://doi.org/10.1145/3626252.3630768

[26] Amber Settle. 2012. Turning the tables: learning from students about teaching CS1. In Proceedings of the 13th Annual Conference on Information Technology Education (Calgary, Alberta, Canada) (SIGITE ’12). Association for Computing Machinery, New York, NY, USA, 133–138. https://doi.org/10.1145/2380552.2380594

[27] J. McLean Sloughter, Agnieszka Miguel, Mara Rempe, and Katie Kuder P.E. 2019. Survey Analysis of Student Experiences for Underrepresented Populations in Engineering and Computer Science. 2019 CoNECD - The Collaborative Network for Engineering and Computing Diversity. https://doi.org/10.18260/1-2--31794

[28] Sónia Rolland Sobral. 2021. Teaching and Learning to Program: Umbrella Review of Introductory Programming in Higher Education. Mathematics 9, 15 (2021), 23 pages. https://doi.org/10.3390/math9151737

[29] Tiera Tanksley. 2023. Employing an Abolitionist, Critical Race Pedagogy in CS: Centering the Voices, Experiences and Technological Innovations of Black Youth. Journal of Computer Science Integration 6, 1 (2023), 9. https://doi.org/10.26716/jcsi.2023.12.27.49

[30] Jodi L. Tims, Cindy S. Tucker, Mark A. Weiss, and Stuart H. Zweben. 2024. Student Enrollment and Retention in 2022–23 U.S. Undergraduate Computing Programs. ACM Inroads 15, 4 (2024), 20–46. https://doi.org/10.1145/3700774

[31] TriCollege Libraries. 2020. Haverford 2020 Student Strike Collection. https://digitalcollections.tricolib.brynmawr.edu/collections/haverford-2020-student-strike-collection.

[32] Arto Vihavainen, Jonne Airaksinen, and Christopher Watson. 2014. A systematic review of approaches for teaching introductory programming

and their influence on success. In Proceedings of the Tenth Annual Conference on International Computing Education Research (Glasgow, Scotland, United Kingdom) (ICER '14). Association for Computing Machinery, New York, NY, USA, 19–26. https://doi.org/10.1145/2632320.2632349

[33] Chris Wilcox and Albert Lionelle. 2018. Quantifying the Benefits of Prior Programming Experience in an Introductory Computer Science Course. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE '18). Association for Computing Machinery, New York, NY, USA, 80–85. https://doi.org/10.1145/3159450.3159480

[34] Craig E. Wills. 2022. Outcomes of Advertised Computer Science Faculty Searches for 2022. Technical Report. Worcester Polytechnic Institute. https://web.cs.wpi.edu/~cew/papers/outcomes22.pdf

[35] Brenda Cantwell Wilson and Sharon Shrock. 2001. Contributing to success in an introductory computer science course: a study of twelve factors. In Proceedings of the 32nd SIGCSE Technical Symposium on Computer Science Education (Charlotte, North Carolina, USA) (SIGCSE '01). Association for Computing Machinery, New York, NY, USA, 184–188. https://doi.org/10.1145/364447.364581

[36] Erik Wingrove-Haugland and Jillian McLeod. 2021. Not "Minority" but "Minoritized". Teaching Ethics 21, 1 (2021), 1–11. https://doi.org/10.5840/tej20221799

[37] Dianna Xu, Ursula Wolz, Deepak Kumar, and Ira Greenburg. 2018. Updating Introductory Computer Science with Creative Computation. In Proceedings of the 49th ACM Technical Symposium on Computer Science Education (Baltimore, Maryland, USA) (SIGCSE '18). Association for Computing Machinery, New York, NY, USA, 167–172. https://doi.org/10.1145/3159450.3159539