ASEE 2022 ANNUAL CONFERENCE
Excellence Through Diversity
MINNEAPOLIS, MINNESOTA, JUNE 26TH-29TH, 2022 ASEE

Paper ID #37738

Academic Help Seeking Patterns in Introductory Computer Science Courses

Lina Battestilli (Teaching Associate Professor)

Dr. Lina Battestilli is an Associate Teaching Professor in the Department of Computer Science at North Carolina State University(NCSU). Her Ph.D. dissertation in Computer Science from NCSU in 2005 and her industry experience at IBM Research were focused on novel networking technologies, cloud computing and performance evaluation. Since 2012, her research interests are in computer science education with a focus of broadening participation in computing. She teaches introductory computer science courses, networking electives and the Computer Science Capstone Course. She is a Program Chair for ACM's SIGCSE TS 2023-2024, and member of the Academic Committee of the Grace Hopper Celebration 2020-22. Dr. Battestilli is the Faculty Advisor for NCSU's ACM-W student chapter, 2020/2021 NCSU DELTA Faculty Fellow and 2021/2023 Cultural Competence in Computing (3C) Fellow, NCSU Equity for Women Award Nominee 2020 and Invited Member to NCSU's COVID Instruction and Student Support Task Force.

Matthew Stephen Zahn

Matthew Zahn is a PhD student in Computer Science at North Carolina State University. He received his bachelor's degree in Computer Science at NC State in 2021. He has worked as a Peer Teaching Fellow for the Computer Science department's Software Fundamentals course since 2019. His research interests include computer-based education, game development, and software engineering.

Sarah Heckman (Teaching Professor)

Describing Academic Help Seeking Patterns in Introductory Computer Science Courses

Matthew Zahn, Lina Battestilli, Sarah Heckman
Department of Computer Science
North Carolina State University

Abstract

This paper explores the utilization of help-seeking resources in two computer science courses across two semesters, taken at the same university: a CS1 for Engineering majors ($n = 326$) and a CS2 for Computer Science majors and minors ($n = 238$). Asking, receiving and processing academic help is considered an important self-regulated learning skill. The help-seeking interactions faculty encounter will vary depending upon the course structure and the student demographics. Our goal in this study is to explore differences to determine whether or not patterns exist in how students are seeking help. First, we group students based on their usage of an online discussion forum and their frequency of attending office hours. Next, we describe these help-seeking groups using prior programming experience, course performance and the students' confidence in their computing skills. Our results match expectations with help-seeking; students who participate on the course discussion forum tend to perform better than students who do not and students with low confidence in computer science skills in the CS2 class attend office hours more frequently. Practitioners can utilize these findings to make decisions about how to structure the help provided in their courses and determine ways to support students that need more help.

1 Introduction

The process of learning for mastery extends beyond the classroom. Students are exposed to material via lectures (live or recorded) and process the materials through completing assignments. When students have questions about the course material, their learning continues when they seek academic help. Historically, the typical way to ask for help has been via office hours when the students can synchronously interact with the course teaching staff. Online discussion forums for asynchronous help seeking are also common in higher education.

Some students take full advantage of the academic help provided in a course, while others do not seek any academic help. Students' help-seeking strategies have been linked to their cognitive engagement, course-related anxiety and academic performance[1]. Won et al.[2] noted that being aware of how to ask, receive and process academic help is considered an important self-regulated learning skill. They also found that students' perceptions of their social contexts inform if and how they seek help with their learning. Students who adopt mastery goals are more likely to engage in autonomous help-seeking, whereas those who adopt performance goals either avoid seeking help or seek expedient help[3,1]. Some students consider office hours a last resort when they anticipate a failing grade rather than a help resource to support learning[4]. There may also be a stigma that asking for help makes a student look incompetent because they cannot learn on their own[2].

The objective of this research is to identify patterns in academic help-seeking behavior in early CS courses (CS1 and CS2). Our goal is to improve the success rates of students taking CS1 and CS2 courses by connecting them to the academic help-seeking resources available in the class. Our main research questions are:

RQ1: Which distinct help-seeking patterns in introductory computer science courses can be identified from discussion board and office hour usage data, and how do these patterns affect student learning performance as measured by course grades?

RQ2: What are the computing attitude mindsets for the different help-seeking groups?

To our knowledge, there has not yet been a characterization of student help-seeking patterns using data about the usages of online discussion forums and office hours interactions. There are also differences in help-seeking behavior between students in different computing pathways (majors and non-majors) and how the CS course is structured (flipped or non-flipped format). The focus of this research is to investigate different help-seeking patterns and correlate them to the students' learning performance (i.e. grades), prior programming experience, and confidence in their CS skills.

2 Related Work

Increasing enrollments in CS [1] mean that more students are seeking academic help, especially before assignment deadlines. The two main types of help-seeking in CS are through asynchronous online discussion forums and synchronous office hours where teaching staff directly interact with one or more students.

2.1 Likelihood of Help-Seeking Interactions

Novice programmers struggle identifying when they need help for solving programming tasks; they may wait too long to ask or not provide enough detail[5]. Recent CS graduates join industry with adequate design and development skills but their communication, collaboration, and orientation skills are not as well addressed[6].

Karabenick[3],[1] describes many complexities that influence the likelihood a student will seek help. They describe two types of help-seeking: *autonomous help* which is help that will reduce the student's need for help later by discussing strategies for success and *expedient help* which is help that reduces the student's workload like receiving an answer to a question. Karabenick and Knapp[7] found that students who feel threats to their self-esteem are less likely to seek autonomous help. Karabenick[1] also found the likelihood a student will seek help is influenced by their perception of the course's goal orientation. A mastery orientation (comparing oneself with

[1] Average number of CS majors continued its rise in 2018-19, 2019 Taulbee Survey

oneself) was positively correlated with students seeking help while a performance orientation (comparing oneself with others) was negatively correlated.

2.2 Online Discussion Forums Interactions

Online discussion forums provide students the mechanism for asking questions asynchronously while they are actively engaged in their work. Students may feel lower pressure[8] and encounter fewer social barriers[9] to asking questions by using discussion forums, especially when they can appear anonymous to peers. The use of online discussion forums may even contribute to improved outcomes in CS[10].

Vellukunnel et al.[11] found that most student help-seeking activities in CS involve *constructive* questions related to finding and fixing faults in their programs and that asking constructive questions is correlated positively with course grades. Additionally, they found that 81% of students in introductory CS courses between two institutions posted at least once to the online discussion forum (Piazza) and that over 99% of the students viewed the posts[11]. Mihail et al.[12] found that students who scored better in a class post more to the discussion forum than students who scored worse in a class.

2.3 Office Hours Interactions

Office hours provide the opportunity for students to receive synchronous help from the teaching staff, both from instructors and Teaching Assistants (TAs). Provided help by the TAs is an effective help-seeking resource for students across disciplines that can decrease withdraw rates and increase retention in future courses for both students[13,14] and the TAs themselves[15].

Recently, there has been development of software tools that manage and collect analytics about the usage of office hours[16,17]. Using analytics from the *Queue* tool, Ozymko et al.[16] found that 20% of the students in a course ask 82% of the questions. Furthermore, office hours were used by 64% of the students and a few students had significantly more office hours interactions than the rest of the students. They also found that students who attend lots of office hours do well on graded assignments but not necessarily on proctored exams.

Ren et al.[18] analyzed the contents of pre- and post-office hours interactions surveys to identify the types of questions students asked during office hours and how those questions aligned to the steps of the Design Recipe[19]. They found that students tended to ask questions related to the implementation of a function and testing the correctness of a function. TAs tended to report that the actual interactions involved multiple steps of the Design Recipe, typically including understanding the problem and the data definition, which demonstrates the effectiveness of TAs in supporting students in working on the right part of the problem.

Smith et al.[17] analyzed office hours interactions as recorded via the MyDigitalHand (MDH) tool. Similar to Ren et al.[18], the MDH tool requests information about the problem the student is having and information from the teaching staff and student about the success of the help interaction. The study found that fewer than 50% of the students attended office hours. Of the students who attended office hours, 50% of the office hour time was utilized by 5% of the students.

In this research, we build upon the work of Smith et al.[17]. The novelty of our work is that we investigate analytics about office hours and discussion forums which gives us a more comprehensive way to describe the students' help seeking behaviour in introductory CS courses.

3 Method

We collected data about the help-seeking habits of students in an introductory CS1 course for engineering majors and a second-semester course for computer science majors and minors. The data was collected in Fall 2020 and Spring 2021 at a large public, research-intensive, university in the United States in two courses: CS1-Engineering (CS1-Eng) and CS2-Object Oriented Programming (CS2-OOP). The CS1-Eng course was taught by the same instructor both semesters. The CS1-Eng covers typical CS1 topics to undergraduate engineering students. The CS2-OOP course had a common instructor in five of the six sections, including a distance education offering each semester. The remaining section, in Fall 2020, was taught by another instructor using similar course materials and the same assessments.

3.1 Course Context

CS1-Engineering The CS1-Eng course is a *flipped* introductory programming course for engineering students. The course covers typical CS1 topics using the MATLAB programming language. The course is designed to follow weekly *learning cycles*. The weekly schedule per learning cycle is shown in 1. There are three tests in the semester, each focusing on the material from the previous learning cycles. There are also three larger programming projects due right before each test.

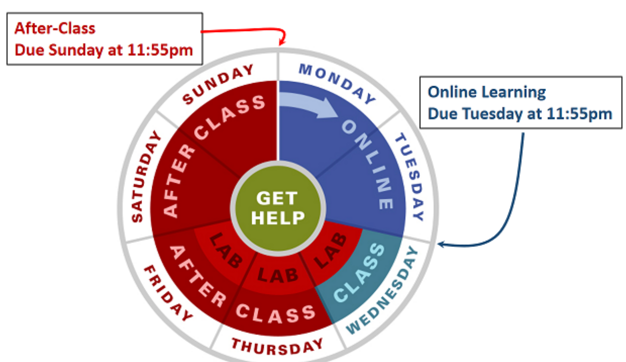


Figure 1: Schedule of a weekly learning cycle for CS1-Eng.

The *online* component of each cycle consists of about 3 hours of self-paced online learning where students are expected to watch pre-recorded videos and answer self-checked, multiple-choice questions on the topics covered in the videos. The *in-class* component of each cycle consists of 2-hr of class time with the instructor where students work on conceptual and simple programming questions that do not require them to write or submit code. Instead, students answer questions interactively via their mobile devices or laptops and work in small groups using peer instruction pedagogy[20]. The students then participate in a 3-hour *lab* led by undergraduate Teaching

Assistants (TAs) with a ratio of 20-30 students per 2 TAs, and consist of a multiple-choice quiz and solving programming exercises by writing and submitting code. In the *after-class* component students individually complete a set of online auto-graded programming problems.

CS2-OOP The CS2-OOP course is the second of a three-semester introductory sequence for computer science majors and minors and consists of two 75-minute lectures. There is a separate 110-minute lab course that is a co-requisite; forum and office hours resources are common for students in the lecture and/or lab. The course covers advanced OO (inheritance, interfaces, abstract classes, polymorphism); software engineering (design, testing, tools and practices); finite state machines; use and implementation of linear data structures (array-based lists, linked lists, stacks, queues, iterators); and recursion (general recursion overview and recursive lists). The programming language of instruction is Java.

Students complete three guided projects, which review prerequisite materials and progressively introduce tooling and new concepts through a combination of guided practice and independent tasks. Students additionally complete two, multi-part projects. During the academic year considered in this study, weekly quizzes with an end of semester retake policy were utilized in lieu of examinations due to the COVID-19 pandemic. In lab, students work on small teams to complete 12 lab activities that build over the course of the semester. Each lab focuses on a key topic in the course.

3.2 Help-Seeking Options: Online Discussion Forum and Office Hours

Students had the option to receive help asynchronously through the use of a Piazza online discussion forum. In Piazza, after a question has been submitted, the teaching staff can review the post and submit a collective answer in a section called “Instructor Answer”. In addition to the Instructor Answer, students can also submit answers to the “Student Answer” section of each question if they have encountered a similar issue and want to share their experience with the problem. For more difficult questions, Piazza allows for a “Follow-up Discussion” on each post where students and instructors can reply to the original poster and continue troubleshooting the issue. Finally, students can also submit “Notes” to the forum, which are posts without the Student and Instructor Answer sections. Notes are mainly used by the teaching staff to send announcements to all forum members.

Office hours were held via video conferencing in Zoom and were managed with the MyDigitalHand (MDH) tool[17]. MDH is a web-based tool which creates a queue and tracks one-on-one help interactions. Students log in and “raise” their hand, answering three to four multiple choice and free response questions describing the issue they need help with. The teaching staff member (instructor or TAs), who is holding the office hours, would then accept the student request from the queue and assist the student with their issue. Once the issue has been resolved, the teaching staff will mark the MDH ticket as complete and answer a few follow-up questions to describe how the help interaction went. If the student still needs more help, the teaching staff can re-enter the student into the MDH help queue or the student can “raise” their hand again so that their issue can be examined by a different teaching assistant or instructor.

3.3 Participants

This study considered students enrolled in CS1-Eng or CS2-OOP during the Fall 2020 or Spring 2021 semesters. The number of students from each course, consenting participants, and gender breakdown of consenting students is shown below in Table 1. Note that some participants chose not disclose their gender.

The participants were all at or over the age of eighteen and consented to the anonymized use of their semester-long course metrics. While the analysis of this research was carried out on the students solely it is important to recognize that the original anonymized data set also included teaching assistants and instructors, whose metrics were collected from their participation on the online forum and in office hours. The initial processing of the raw data set included the removal of any user who was classified as a teaching assistant or instructor of either course.

Table 1: Consent and Demographics of Participants

Course	CS1-Eng	CS2-OOP
Total Enrolled Students	568	583
Consenting Students	326	238
Consent Rate	57.4%	40.8%
Male	75.5%	69.3%
Female	19.6%	20.6%

3.4 Metrics and Data collection

3.4.1 Help-seeking Metrics

Metrics about the students were collected from both Piazza and MDH. Using these metrics, we group students based on their help-seeking behaviors. The specific metrics considered are:

- **Forum Views (FV)**: the total number of posts the student *viewed* on the online discussion forum; including both student posts and notes.
- **Forum Posts (FP)**: the total number of posts the student *submitted* to the online discussion forum; including questions, answers, follow-up discussions, and notes.
- **Office Hours Interactions (OHI)**: the number of Office Hour Interactions that were marked as completed by the teaching staff (TAs or instructor) in MDH.

3.4.2 Survey Metrics

Additionally, within the first two weeks of each semester, students were asked to complete a survey to determine their prior programming experience and their confidence in their CS skills. The survey for the CS2-OOP course included 26 questions from the Computing Attitudes Survey (CAS)[21]. The CAS questions were cross-validated through confirmatory analysis that was found to be consistent with the theoretical framework which drives instrument development[21]. The CAS questions were not included in the CS1-Eng Fall 2020 and Spring 2021 survey. Therefore we were only able to analyze the computing attitude mindset of students enrolled in CS2-OOP. In the survey, for both courses, we asked the students to select all their prior programming experience from a list including AP Computer Science, Self-Taught, Professional Experience, and attending classes at another university or community college.

3.5 Help-Seeking Groups

Using the help-seeking metrics from Section 3.4.1, we grouped the students based on how they utilized the help-seeking resources into the six groups in Table 2. The *Non-Users* are students who never contributed to the online forum and viewed less than 25% of the number of views. The *Lurkers*, are students who also never contributed to the online forum and viewed greater than or equal to 25% of the number of views. The *Contributors* are students who have asked a question or submitted a student answer or responded to a post in a follow-up discussion. In the CS2-OOP course, 84.5% of students contributed to the forum while less than half, or 44.8% of the students contributed to the CS1-Eng forum. Notably, a large number of the students in the flipped CS1-Eng course were in the *Lurkers* group.

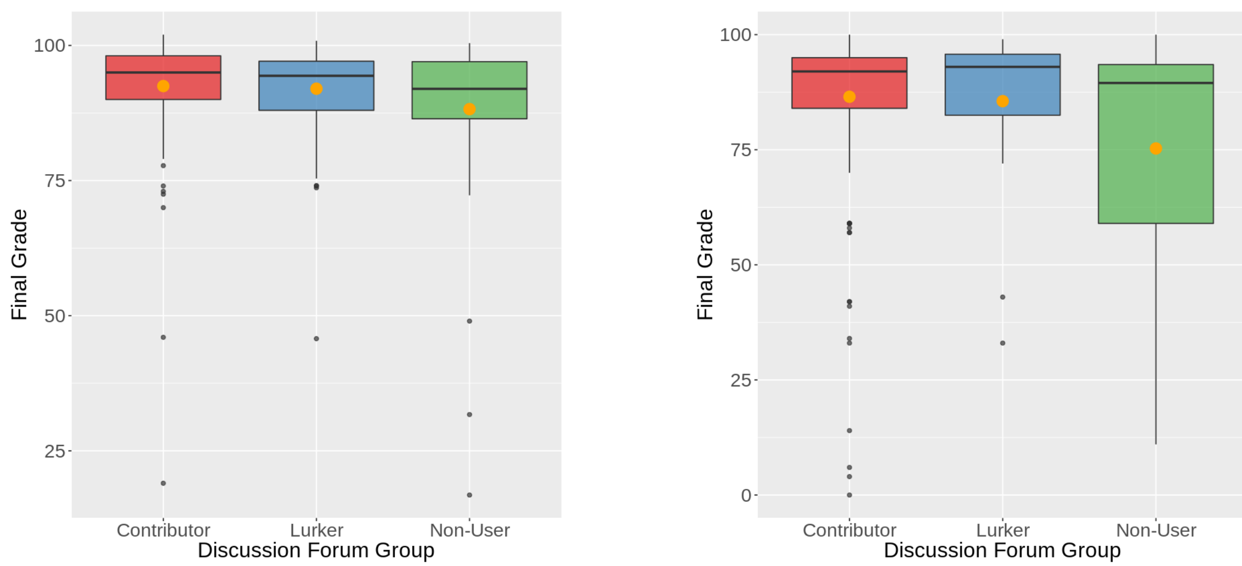
For office hours, the *Non-Goers* are students that never attended a single OHI while the *Goers* are students that attended at least one OHI during the semester. Finally, the *Super-Goers* are students that utilized office hours at higher levels compared to their peers To determine the cutoff of interactions for *Super-Goers*, the number of OHI of each student were graphed on a scatter plot and horizontal lines were drawn to visually separate students that deviated from the majority.

Table 2: Help-Seeking Groups

Group	Description	CS1-Eng	CS2-OOP
Discussion Forum Groups			
Non-Users	$FP = 0$ and $FV <$ the first quartile number of views	19.3%	8.0%
Lurkers	$FP = 0$ and $FV >=$ the first quartile number of views	35.9%	7.6%
Contributors	$FP > 0$	44.8%	84.5%
Office Hour Groups			
Non-Goers	$OHI = 0$	61.0%	34.9%
Goers	$OHI > 0$	39.0%	65.1%
Super-Goers	OHI are outliers	2.8%	5.0%

4 Analysis and Results

To better understand help-seeking behaviour, we describe the help-seeking groups by looking at the students' course grades on a 100% scale, prior programming experience, and computing attitudes mindset.



(a) CS1-Eng, Kruskal-Wallis $p = 0.029$

(b) CS2-OOP,Kruskal-Wallis $p = 0.1511$

Figure 2: Grades for each Discussion Forum Group

4.1 Course Grade (RQ1)

First, we compared the final course grade of each help-seeking group to find of if there is any relationship between the help-seeking behavior and course performance. Due to the presence of non-normal data, a Kruskal-Wallis test was utilized to determine whether or not there was any statistically significant relationship between the help-seeking groups and the course grade of students.

Discussion Forum: Figure 2a shows that students in the CS1-Eng flipped course, who either actively or passively used the online discussion forum on average received higher final course grades than students who do not make use of this help resource ($p = 0.029$). Both the mean (depicted by the orange dot) and median final grade for “Contributors” and “Lurkers” are higher than for the “Non-User” behavioral group.

Figure 2b shows the final course grades for help seeking discussion forum groups for the CS2-OOP course. The plot shows a difference in the mean and variance of the groups, confirming that students who utilize the provided help-seeking resources will, on average, perform better in the course than students who are not using the online discussion forum at all. We did not find any statistically significant relationships for the CS2-OOP course when comparing the final course grade of the students and their help-seeking behavioral based on their usage of the discussion forum. While the averages are different, the medians are not. Additionally, a majority of students in CS2-00P were contributors (84.4%) while very few were non-users (7.9%). The large difference in group sizes and the large variance in the non-users group contributes to the non-significant result.

Office Hours: For CS1-Eng and CS2-OOP, the Kruskal-Wallis tests did not report any statistical significance with the median final grades for the office hour behavioral groups. Despite the lack of statistical significance, comparing the medians of each group revealed that students who utilized the office hour resources are more likely to score higher in the course than their peers who chose not to utilize the office hour resources. These findings are in-line with the comparisons between discussion forum behavior and final grades, but not significantly different.

4.2 Prior Programming Experience (RQ1)

Next, we compared the Prior Programming Experiences (PPE) of each group, collected from the survey responses, to find any relationships with help seeking behavior. Since the data set was not normally distributed, Pearsons' chi-squared test was used to determine whether or not there is a significant association between prior experience and student help-seeking behavior. Due to the differences between the two courses, one being CS1 and the other a CS2 course, we evaluated the PPE differently in each course. In CS1-Eng, we were only interested in knowing whether or not students had at least one form of PPE. In CS2-OOP we wanted to know how many students started this course with PPE centered around the previous course, CS1-Major. To achieve this we determined whether or not students had one of the following PPE categories: 1) had CS1-Major as their *only* form of PPE, 2) had CS1-Major and *other forms* of PPE, or 3) had *any other form* of PPE that was not CS1-Major, which includes credit for CS1-Major through the AP CS A exam or equivalent courses elsewhere. Table 3 shows PPE of the study participants.[2]

Table 3: Help-Seeking & Prior Programming Experience

Group	**CS1-Eng**		**CS2-OOP**		
	PPE	**No PPE**	**CS1-Major Only**	**CS1-Major & Other PPE**	**Other PPE**
DF:Non-Users	9.8%	9.5%	.4%	1.7%	3.8%
DF:Lurkers	13.8%	22.1%	3.4%	2.5%	.8%
DF:Contributors	19.0%	25.8%	29.8%	28.1%	22.7%
OH:Non-Goers	28.2%	32.8%	10.1%	7.6%	13.0%
OH:Goers	14.4%	24.5%	23.5%	24.8%	14.3%
OH:Super-Goers	1.5%	1.2%	1.7%	1.7%	1.3%

In CS1-Eng, results indicated that there were no significant associations between student help-seeking behaviors and whether or not students had any form of PPE. Out of the 326 students in CS1-Eng, 57.4% started the course with no form of PPE. When examining the distribution of PPE within each help-seeking group, there is an even spread of students who do and do not have PPE within each group.

In CS2-OOP, each of the three previously mentioned categories of PPE were tested separately against the help-seeking behavior groups. When evaluating students who had CS1-Major as their *only* form of PPE, tests indicated that there was a significant association with discussion forum help-seeking behavior. When examining students who had CS1-Major as well as other forms of PPE, chi-squared tests showed that there was a significant association with office hour help-seeking behavior. Finally when analyzing students who had forms of PPE that were not

[2] Student falls in a DF and an OH group. Not all students completed the survey.

CS1-Major, chi-squared tests indicated that there were significant associations with how students seek help through both the discussion forum as well as office hours.

4.3 Computing Attitudes Mindset (RQ2)

Finally, we took the calculated percentage score of student answers to the CAS questions that aligned with how a computing expert would agree with the statement and compared the scores across each group in the CS2-OOP course[21]. This comparison was done only in the CS2-OOP course as the CAS was not utilized in CS1-Eng. Since the percentage scores are not normally distributed, the Kruskal-Wallis test was utilized to determine whether or not any statistically significant relationships exist between student's computing attitudes mindset and their help-seeking behavior.

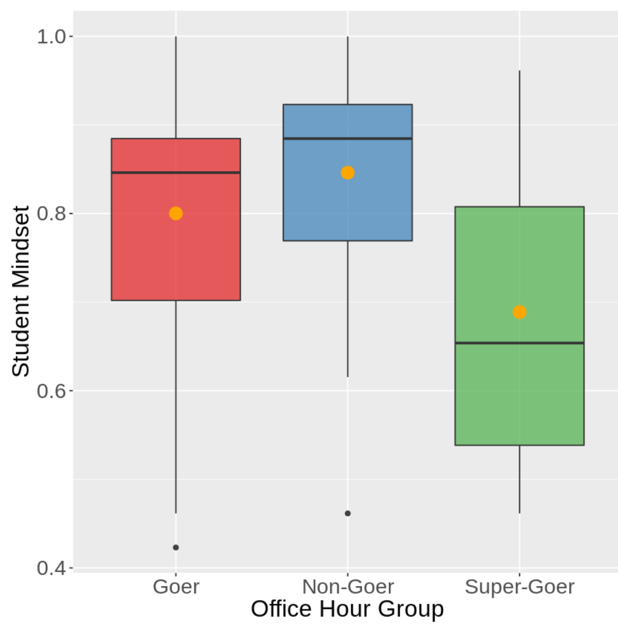


Figure 3: CS2-OOP: Mindset based on Office Hour Behavior ($p = 0.002$)

Our results indicate that there is a statistically significant relationship between student mindset and office hour behavior ($p = 0.00203$). Following this significant p-value, pair-wise Wilcoxon Rank-Sum tests were utilized to determine which office hour group contributed to the significant p-value. The results show that all pair-wise groupings between the three office hour groups contributed to the Kruskal-Wallis significance.

Examining the relationship more closely, we can see that the students who are less confident in their computing ability are the students who are making the most use of the office hour resource. In CS2-OOP, the "Super-Goers" group had an median 65.4% confidence level, the lowest of the three groups, followed by the "Goers" group with a median of 84.6%, and finally the "Non-Goers" group had the highest confidence with a median of 88.462%.

When analyzing the comparison between student's computing attitudes mindset and their discussion forum behavior, the Kruskal-Wallis test did not report any statistically significance differences between the medians of each group. ”Contributors”, ”Lurkers”, and ”Non-Users” alike all had nearly identical median scores and variance. From this we can conclude that the computing attitudes mindset of each student does not contribute to their discussion forum behavior compared to that of their office hour behavior.

5 Discussion

We find that the help-seeking groups vary between the two introductory CS classes. These are also differences between the help-seeking groups when considering final grades, prior programming experience, and computing attitude mindset.

Course Delivery: Students in the traditional CS2-OOP course utilized more help-seeking resources that the students in the flipped CS1-Eng course. This may be attributed to the fact that the CS1-Eng students spent nearly 5hrs per week with the instructor and TAs and thus their help-seeking is not recorded via our tools. This is inline with one of the key goals of flipping a course, which is to move passive learning to be independent (i.e. viewing of the lectures) and to provide the students access to the teaching staff so they can seek help when they are actively working on assignments.

Course Grades: Our results show that discussion forum “Contributers” and “Lurkers” had higher grades than “Non-Users” for the CS1-Eng, similar to the findings in[11]. However, there are no differences in course grades by discussion forum usage for the CS2-OOP students, likely due to the heavy forum utilization. Additionally, there were no statistical differences in grades based on office hours usage, however, students who attended office hours did have a higher median grade than those who did not, suggesting further study might be needed.

Prior Programming Experience: Students in CS2-OOP who took CS1-Major only utilized the forum, students with CS1-Major and other experience utilized office hours, and students who took a CS1 equivalent in some other way utilized both the discussion forum and office hours. *This suggests that the help-seeking utilized in earlier experiences may impact how students seek help in CS2*. There may be a emphasis on having students ask questions in the forum over in-person office hours in CS1-Major so students where that is their only prior experience default to that form of help-seeking. There were no major differences in CS1-Eng, so help-seeking expectations may not yet be fully formed for programming classes.

Computing Attitude Mindset: The investigation of mindset and CS2-OOP help-seeking found that *students who are less confident are more likely to attend office hours*. And those who are the least confident are “Super-Goers”. This is encouraging that students who are less confident recognize the need for help, however, the “Super-Goers” may be too reliant on office hours help. *Effort should be made to help the “Super-Goers” increase their confidence in completing computing tasks independently.* There were no differences between forum help-seeking groups, likely due to the high forum utilization.

These empirical results show that *help-seeking can benefit student academic success, but that these benefits may vary by course type and delivery mode*. There is only a relationship with grades for the CS1-Eng Piazza groups, so promoting help-seeking behaviors may not be the best intervention to improve final grade outcomes.

5.1 Limitations and Threats to Validity

1) The data in this study was collected during COVID-19 pandemic and courses were taught fully online. The differences in teaching and learning style between fully online, hybrid, and in-person modes of learning impact how students seek help throughout the semester. Therefore, the results of this study cannot describe what student help-seeking behaviors were like prior to the COVID-19 pandemic.

2) The survey given to students in CS1-Eng did not include the CAS instrument, which at the time was an intentional choice to reduce the length of the survey. The relationships between student attitudes and office hours help-seeking behavior cannot, for the purposes of this study, be expanded and compared between the two courses. Future work could consider the CAS instrument to determine if the office hours help-seeking behavior observed in CS2-OOP is seen in other course contexts.

3) The courses were taught by different instructors and the results may not generalize to larger populations.

4) There may be unmeasured confounding variables behind some of the presented results.

5.2 Future Work

Using this research, we intend to suggest improvements for the workflow of MDH to improve the effectiveness of office hours for CS courses. Our ultimate effort is to increase the effectiveness of help-seeking interactions that focus on debugging, and create TA training materials that support effective help-seeking interactions The data from instructors and teaching assistants is still available in the original raw data and can be utilized in future research. Among the metrics are custom responses from the teaching staff, submitted after completed office hour interactions, which may be useful in gauging help-seeking effectiveness from a faculty point-of-view.

6 Conclusions

Our research question asked if students have different patterns of help-seeking behaviors in introductory Computer Science courses. We were able to categorize students into help-seeking groups and studied the relationship between final course grades, prior programming experience, and for CS2-OOP, computing attitudes and mindset. Our findings show that help-seeking behavior varies between the CS1-Eng and CS2-OOP courses when taught in an online format. This may impact how help-seeking is discussed and resources are allocated in similar courses. Additional research is needed to understand these relationships in non-pandemic semesters.

This material is based upon work supported by the National Science Foundation under Grant Nos. [anonymized for review]. We would like to thank [anonymized student] for his initial work on the dataset.

References

[1] Stuart A Karabenick. Perceived achievement goal structure and college student help seeking. *Journal of educational psychology*, 96(3):569, 2004.

[2] Sungjun Won, Lauren C. Hensley, and Christopher A. Wolters. Brief research report: Sense of belonging and academic help-seeking as self-regulated learning. *The Journal of Experimental Education*, 89(1):112–124, 2021. doi: 10.1080/00220973.2019.1703095.

[3] Stuart A Karabenick. Seeking help in large college classes: A person-centered approach. *Contemporary educational psychology*, 28(1):37–58, 2003.

[4] M. A. Smith, Yujie Chen, Rachel Berndtson, K. Burson, and Whitney Griffin. "office hours are kind of weird": Reclaiming a resource to foster student-faculty interaction. *InSight : A Journal of Scholarly Teaching*, 12: 14–29, 2017.

[5] Jason Carter and Prasun Dewan. Design, implementation, and evaluation of an approach for determining when programmers are having difficulty. In *Proceedings of the 16th ACM International Conference on Supporting Group Work*, GROUP '10, pages 215–224, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0387-3. doi: 10.1145/1880071.1880109.

[6] Andrew Begel and Beth Simon. Struggles of new college graduates in their first software development job. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '08, pages 226–230, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-799-5. doi: 10.1145/1352135.1352218.

[7] Stuart A Karabenick and John R Knapp. Relationship of academic help seeking to the use of learning strategies and other instrumental achievement behavior in college students. *Journal of educational psychology*, 83(2):221, 1991.

[8] D. Wu and S. R. Hiltz. Predicting learning from asynchronous online discussions. *Journal of Asynchronous Learning Networks*, 8 2:139–152, 2004.

[9] Natalie K. Person, Arthur C. Graesser, Joseph P. Magliano, and Roger J. Kreuz. Inferring what the student knows in one-to-one tutoring: The role of students questions and answers. *Learning and Individual Differences*, 6 2:205–229, 1994.

[10] Brenda Cantwell Wilson and Sharon Shrock. Contributing to success in an introductory computer science course: A study of twelve factors. *SIGCSE Bull.*, 33(1):184–188, February 2001. ISSN 0097-8418. doi: 10.1145/366413.364581.

[11] Mickey Vellukunnel, Philip Buffum, Kristy Elizabeth Boyer, Jeffrey Forbes, Sarah Heckman, and Ketan Mayer-Patel. Deconstructing the discussion forum: Student questions and computer science learning. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, pages 603–608, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4698-6. doi: 10.1145/3017680.3017745.

[12] Radu P. Mihail, Beth Rubin, and Judy Goldsmith. Online discussions: Improving education in cs? In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, SIGCSE '14, page 409–414, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450326056. doi: 10.1145/2538862.2538898.

[13] K. J. Topping. The effectiveness of peer tutoring in further and higher education: A typology and review of the literature. *Higher Education*, 32(3):321–345, Oct 1996. ISSN 1573-174X. doi: 10.1007/BF00138870.

[14] Gloria Childress Townsend, Suzanne Menzel, and Katie A. Siek. Leveling the cs1 playing field. *SIGCSE Bull.*, 39(1):331–335, March 2007. ISSN 0097-8418. doi: 10.1145/1227504.1227428.

[15] Sarah Hug, Heather Thiry, and Phyllis Tedford. Learning to love computer science: Peer leaders gain teaching skill, communicative ability and content knowledge in the cs classroom. In *Proceedings of the 42Nd ACM*

Technical Symposium on Computer Science Education, SIGCSE '11, pages 201–206, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0500-6. doi: 10.1145/1953163.1953225.

[16] Natalia Ozymko, Matthew Allan McCarthy, Wade Fagen-Ulmschneider, Karin Jensen, and Karle Flanagan. Work in progress: Analysis of the impact of office hours on graded course assessments. In *2020 ASEE Virtual Annual Conference Content Access*, Virtual On line, June 2020. ASEE Conferences. https://peer.asee.org/33960.

[17] Aaron J. Smith, Kristy Elizabeth Boyer, Jeffrey Forbes, Sarah Heckman, and Ketan Mayer-Patel. My digital hand: A tool for scaling up one-to-one peer teaching in support of computer science learning. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, SIGCSE '17, pages 549–554, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4698-6. doi: 10.1145/3017680.3017800.

[18] Yanyan Ren, Shriram Krishnamurthi, and Kathi Fisler. What help do students seek in ta office hours? In *Proceedings of the 2019 ACM Conference on International Computing Education Research*, ICER '19, page 41–49, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361859. doi: 10.1145/3291279.3339418.

[19] Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, and Shriram Krishnamurthi. *How to Design Programs: An Introduction to Programming and Computing*. The MIT Press, 2018. ISBN 0262534800.

[20] Leo Porter, Dennis Bouvier, Quintin Cutts, Scott Grissom, Cynthia Lee, Robert McCartney, Daniel Zingaro, and Beth Simon. A multi-institutional study of peer instruction in introductory computing. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, SIGCSE '16, page 358–363, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450336857. doi: 10.1145/2839509.2844642.

[21] Brian Dorn and Allison Elliott Tew. Empirical validation and application of the computing attitudes survey. *Computer Science Education*, 25(1):1–36, 2015. doi: 10.1080/08993408.2015.1014142.