Understanding Attrition Rates: Why the Drop-Out Rate Within Computer Science Degree Is so High

Project Objective

This project aims to study the quality of courses within the undergraduate Computer Science degree and how it could affect the high attrition rates found in higher education. Although abundant resources exist for Computer Science students, the material can be difficult to learn due to its conceptuality and the student's learning environment. This study will examine the Computer Science courses and labs by surveying students to see if there are any underlying differences between Computer Science and other degree fields that might be holding students back.

Project Background and Significance

In recent years, the drop-out rate of students within the Computer Science degree has been relatively high. Most of these drop-outs occur during the first two years of the degree program averaging a 40% drop-out rate and ranging from 30% to 60%, depending on the university (Giannakoset 2). This is significant because jobs requiring skills taught within the Computer Science curriculum are becoming more prevalent and in high demand, yet students are hesitant to pursue the field (Giannakoset 2). This may be due partially to the little exposure students from K-12 have to computing skills. Although initiatives like "CS for All" exist to help students in primary and secondary education systems acquire computing skills, the efforts are underdeveloped and not widely used (Jun 2). This dramatically affects the ability of incoming college students within Computer Science as they have little knowledge or background of computer technologies, limiting their growth (Jun 2). Due to this lack of exposure, incoming college students are unfamiliar with the concept of computational thinking, which employs the application of other concepts like "abstraction, problem decomposition, algorithms design, data collection and analysis, data representation, and simulation" (Jun 2). Through familiarization with computational thinking, many challenges faced by incoming undergraduate students are alleviated, which could result in lower drop-out rates (Takács 12).

Many approaches have been made to help improve Computer Science attrition rates in higher education by enacting curriculum adjustments or by adding supplementary lab courses. One method investigated was the implementation of group project presentations, where students collaboratively worked together to learn about conceptual Computer Science problems and presented their findings. This allowed the students to get active feedback from their instructors and peers, bolstering their confidence and their knowledge (Bakhry 4). Additional methods include using adaptive-learning programs online that aid students in how to code. The online programs allow the students access to instant feedback fostering quicker growth in their development of programming languages and being familiarized with the syntax (Anindyaputri 1). Another study shows that there is also an impactful difference depending on the teachers' teaching styles and the students' learning styles. It shows that students would perform much better when the teacher provides multiple presentation styles of the same information (Chetty 5). However, many of these methods are not widely utilized and are primarily used in study cases.

It is clear that there are variables that affect the student's ability to perform, which affects the attrition rates of Computer Science in higher education. That is why an analysis of students is necessary to help determine any avoidable determinants in the Computer Science field.

Research Methods

This study will implement a survey and a statistical analysis, examining random samples of students in Computer Science and observing what obstacles may be affecting them. One of the factors that students often cite is poor lecture quality, which can be measured by conducting a campus-wide survey. The survey will be conducted by utilizing a Google Form where students will rate their course experiences. From those that responded, one hundred students will be randomly selected to have the chance to win a $10 Amazon gift card. The advantage of sampling this data, as opposed to using a third-party source such as ratemyprofessor.com, would be that it avoids some voluntary response bias due to the incentivization of an award. In a conventional survey, a surveyor could only obtain data from people who responded willingly; however, they miss out on all the data from those who decided to abstain. Fortunately, with an award, students are more likely to fill out professor reviews, which means that the population sample that is gathered would be a more accurate representation of all students without being subject to any extreme voluntary response bias.

Continuing with this methodology, a unit of measurement is required. In this case, the reviews require each student to numerically rank, on a 1-10 scale, the quality of different aspects of each course. Some of these aspects would be lecture quality, professor explanations, material familiarity, the usefulness of labs, or homework fairness. These metrics would be tested to see if there is a significant drop in the average quality of C courses in comparison to other majors. In order to do this, the experiment would compile data and focus on two primary groups: Computer Science Courses and non-Computer Science courses. The Computer Science group would be the experimental group, while the non-Computer Science group would be the control group. Then, random samples of each group would be taken, and their means and adjusted standard deviations would be recorded. The optimal hypothesis test for this experiment would be a 1-tail, 2-sample null hypothesis test, with an alpha value of 0.05, to see if there is a statistically significant drop in ratings when comparing the C review samples with the non-Computer Science review samples.

Expected Outcome

Once the testing is complete, the resulting p-value would determine whether rejecting the null hypothesis is appropriate. In order to get accurate results, the experiment would have to be repeated multiple times with different random samples. During this process, type I and type II errors would also be considered to correct any false assessment of the p-value. It is expected that the findings reject the null hypothesis in favor of the alternative hypothesis, which states that there exists a significant drop in quality in Computer Science courses. This would mean that there are obstacles that stand in the way of many Computer Science students, and that there would be a further need to counteract these obstacles and better the quality of the Computer Science curriculum. The findings would be published in a research article and shared with the University of Central Florida and other institutions.

The resulting findings can have a major impact on institutions because of the potential growth that their departments could have. In order to do this, the research article will be presented to institutions with high dropout rates. Solutions to remedy the dropout rates would be proposed based on the statistical evidence gathered in the experiment. Improving factors such as study environment, lecture quality, and homework

fairness can all assist students in maintaining their computer science coursework until graduation. Overall, all academic institutions of any department can benefit from the research, not just computer science. This would grant new insight on optimizing the students' probability for success.

Literature Review

Anindyaputri, Natasha A., et al. "Enhancing Students' Ability in Learning Process of Programming Language Using Adaptive Learning Systems: A Literature Review." *Open Engineering*, vol. 10, no. 1, 2020, pp. 820–829., https://doi.org/10.1515/eng-2020-0092.

Bakhru, Sunil A., and Rashesh P. Mehta. "Assignment and Project Activity Based Learning Systems as an Alternative to Continuous Internal Assessment." *Procedia Computer Science*, vol. 172, 2020, pp. 397–405., https://doi.org/10.1016/j.procs.2020.05.073.

Chetty, Nithya D., et al. "Learning Styles and Teaching Styles Determine Students' Academic Performances." *International Journal of Evaluation and Research in Education (IJERE)*, vol. 8, no. 4, Dec. 2019, pp. 610–615., https://doi.org/10.11591/ijere.v8i4.20345.

Giannakoset, Michail N., et al. "Understanding Student Retention in Computer Science Education: The Role of Environment, Gains, Barriers and Usefulness." *Education and Information Technologies*, vol. 22, no. 5, 19 Oct. 2016, pp. 2365–2382., https://doi.org/10.1007/s10639-016-9538-1.

Jun, Andrew, et al. "CS for All: Introducing Computational Thinking with Hands-on Experience in College." *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2017, https://doi.org/10.1109/csci.2017.187.

Takács Rita, et al. "Successful Steps in Higher Education to Stop Computer Science Students from Attrition." *Interchange*, 18 Oct. 2022, https://doi.org/10.1007/s10780-022-09476-2.

Preliminary Work and Experience

As Computer Science majors, we have adequate experience regarding the quality of the Computer Science degree curriculum. Given that we share the same major, we have also shared many of the same courses with the same lecturers. Overall, we have come to the conclusion that there are many barriers that inhibit the learning potential of many students, ourselves included. These barriers are, but not limited to unfair assignments, difficulty of understanding lecturer, and intense workload. We all share these same experiences, despite being at different years in our academic career.

Additionally, other students within the computer science department also take issue with the same grievances that we have. It is a popular claim to assume that computer science lecturers are poor lecturers. Because most computer science students, including us, agree with the same general claim, is can be said that the quality of the computer science curriculum is lacking in many regards.

IRB/IACUC statement

An IRB statement will be required as individuals will be surveyed

Budget

$1000 - 100 $10 Amazon gift cards

Total Funding Amount: $1000