Computer Science Curriculum Redesign Proposal

CS4991 Capstone Report, 2025

Tyler Donovan
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
anm2vc@virginia.edu

ABSTRACT
UVA's computer science (CS) curriculum fails to equip students with the skills necessary to apply into the increasingly competitive tech industry. I propose restructuring the order of specific CS curricula to better fit application cycles, as well as adding CS courses specific to internship preparation. This approach would include delaying low-level programming and theory-heavy classes to be completed later in the program while prioritizing skills and project development classes. From this approach we can expect students to be more prepared for the time-sensitive internship application cycle before moving on to more CS theory centric classes. This proposal needs to be piloted to measure its effectiveness against students' current success. If the data shows positive results, it would need to be expanded and integrated into the actual curriculum.

1. INTRODUCTION

Internships play an integral part in allowing CS students to obtain competitive and high paying jobs directly out of college. However, with the increasing number of CS students, the competition has grown, it is becoming more difficult to land these internships and many students do not know when or where to start. In 2025 it is important that students start applying early, develop their own personal projects, study for technical interviews, and work in teams as early as 2nd semester freshman year if they want to remain competitive for the top positions. This is extremely stressful for students to navigate on their own and the CS program currently does not prepare them well to accomplish these goals.

Students come to a prestigious university like UVA for computer science not only to learn the theory and structure of computer science, but also to obtain a stable job outside of college. With the advantage of a restructured UVA CS program, students will be better prepared to tackle the difficulties of landing an internship and job.

2. RELATED WORKS

Microsoft, a pioneer in the computer science world, published a blog from 2023 highlighting the importance of CS in the current workforce as well as the positive impacts it has on the success of individuals in areas outside of CS. In their Microsoft Computer Science Guide (MCSG) they prioritize creative expression and development of computational thinking skills. They also note the importance of developing real-world problem-solving skills and hands-on activities in early learners (Microsoft Education Team, 2023). While this article alludes to younger learners, the principles stated apply to college, as well.

Much of the UVA CS curriculum features hands-on project-building and real-world problem-solving skills later in the curriculum

and it is easy to lose sight of the original purpose of learning CS. Applying these principles, moving project-based curriculum forward in the CS program can help students connect concepts to the real world while concurrently preparing them for internship applications. Further supporting this claim is an article by Hereford Sixth Form College in the UK deconstructing their CS curriculum. Their coursework involves learning problem-solving and computational thinking skills, programming techniques, standard and complex algorithms, software development methodologies, databases and networks within a 2-year program. Accelerated and compact, this program focuses on the key concepts that students should be focusing on to get a good foundation for computer science (Hereford Sixth Form College, 2014). We can apply these same concepts to the UVA curriculum to make sure that students have all the tools necessary, as well as some completed project experience to be a more competitive candidate for internships.

Another thing worth mentioning that is too often overlooked is the importance of developing soft skills for the job. Team projects, deadlines, organization, etc. are all important concepts that we learn in more project-oriented classes that really lead to success in jobs. Taking these classes earlier in the curriculum or having them integrated into basic classes can really help prepare students for the workforce (Durakovic, 2023).

PROJECT PROPOSAL

Algorithmic problem-solving skills have become increasingly difficult in current application cycles, so adding a class to address this and internship application guidelines (ideally in a student's 1st year) would substantially increase the chances of that student's success by at least making them aware of these processes. Additionally, restructuring the program so that classes surrounding project development with peers can help expose students to software development practices earlier and lead to better and thoughtful personal projects necessary to be a competitive applicant.

3.1 Review of Current CS Curriculum

Currently at UVA CS foundational courses include intro to programming (CS1110), data structures and algorithms 1 and 2 (CS2100; CS3100), discrete mathematics 1 and 2 (CS2120; CS 3120) computer systems and organization 1 and 2 (CS 2130, CS 3130) and software/advanced software development essentials (CS 3140; CS 3240). UVA states that "The 2000 level courses should be taken before the 3000 level courses… other prerequisites govern the order that these courses should be taken" (Engineering.Virginia.edu, 2025), showing that there is a current defined order that UVA enforces for CS students. The problem is that 2000's level courses will typically take up most of a student's 2nd year where that summer they will have to start applying for crucial 3rd year internships. Unless a student takes all 2000 level courses in a single semester followed by all 3000 level courses the next semester, there is no possible way the student will have anything beyond foundational knowledge when they need to be competitive for the recruitment process.

3.2 Proposed Solutions

Because of the unique problem of time constraints on CS recruiting season, changes need to be made to the CS curriculum to help prepare students for the job market. In what follows I present some potential solutions to this problem which include reordering of curriculum and the addition of essential recruiting skills classes.

3.2.1 Reordering of Classes

Taking the existing CS curriculum and simply reordering it could prove to be very beneficial

for students. The time frame I will be focusing on is what students take in their 2nd year as their 1st year (if engineering) will be filled with engineering-specific prerequisites. The primary components that students need to be comfortable with for CS interviews is the behavioral, technical, and sometimes system design interview. UVA does a great job providing opportunities to do practice behavioral interviews with real employers, but not so much with technical interviews. These technical interviews have heavy emphasis on data structures and algorithms as well as critical thinking and problem-solving skills. Given the time constraints of these interviews, python is also typically used (Tay, 2024).

This means it is essential for students to be familiar with CS2100 and CS 3100 content before their 3rd year. This also means students ideally take CS3140 and CS3240 for actual project development experience, team experience, and system design knowledge. A competitive student should have proficiency in programming languages, experience with tools like Git, and relevant coursework and projects that highlight their edge (Tealhq.com, 2025). Taking these courses up front not only better prepares them with these skills, but makes it easier for them to develop their own projects in a more professional way (like using a repository) that will allow them to gain a competitive edge.

Moving these courses up will be more challenging on the students but otherwise these are things they will be forced to do on their own to get these internships. The courses that should be shifted back into 3rd year as a result of moving these classes up would be CS2120, CS2130, CS3120, and CS3130. While these do teach crucial skills, it is not necessary for students to know this information for the vast majority of internships. The students can still benefit from learning lower-level programming and CS theory later, while also gaining the benefits of being well-prepared for recruiting season, making it a win-win.

3.2.2 Potential Additional Courses

No course currently exists to help students prepare their technical and soft skills for recruitment season, even though it is difficult and hard to prepare for alone. Therefore, offering a required or elective course to help students in this process could be incredibly beneficial. Content for this class could include practicing technical interview problems (similar to leetcode), mock technical interviews with classmates, learning soft skills and communication, systems design review, tools and systems to help with the application process, resume-building, and general interviewing skills. This would provide immense support to students who need more guidance beyond the career center to land a job. Additionally, making this a required course would increase every student's chances of getting an internship and job rather than leaving it up to chance and making them figure out everything themselves.

While this is heavy guidance and there is an argument that it is important for students to become self-sufficient, sometimes just making students aware of these opportunities can make all the difference in their future success. If put into effect, all of this content could be expanded upon and made into lessons and assignments for students to complete. Most importantly it would provide them with some guidance about crucial information needed to help secure their futures.

4. ANTICIPATED RESULTS

Since only six people reported for 2021-2023, it is hard to quantify the impact these changes may have on CS student outcomes (Career.Virginia.edu, 2025). However, this Reddit post from 2024 describes the frustrations CS students currently are

experiencing surrounding recruitment. Anecdotally from this student and myself, there are many people who didn't even know how to navigate recruiting and almost 90% of what had to be done to become a competitive applicant was done individually without support from the school (Reddit.com, 2024).

The proposed changes could have a major impact on CS graduate students' success. Students will have adequate preparation but also awareness going into recruiting season, increasing their chances of success in landing an internship. Many 3rd-year internships lead to return offers from those same employers, so improvements in post-graduate employment for CS could also increase as a result. CS students becoming better competitors in the job market as a result of preparation and the school could lead to connections with larger employers, expanding UVA's network and prestige within CS nationally. Overall, CS students will be able to take tangible skills learned in these programs, apply them to get better jobs and internships, with a downstream effect that the view of UVA's CS program improves as their graduates are landing jobs and making changes within the software industry.

5. CONCLUSION

The current UVA CS program does a fantastic job teaching us how to think about computer science and theory, but still lacks balance in teaching students how to obtain internships and jobs, which is most students' ultimate goal. Through the restructuring of the CS curriculum, UVA will not only be able to maintain the ability to teach high quality CS theory, but also prepare students for internship and job application seasons in an adequate and timely manner. Moving forward, algorithm and application heavy classes into 2nd and 3rd semester of the CS curriculum will allow students to build technical skills for technical interviews and also team/soft skills that cannot be taught in a lecture. Additionally, a potential internship preparation elective could really help boost a students' ability to obtain an internship in their second year or better prepare them to obtain one in their third year. Together, these changes will allow students to be better prepared to get internships, gaining industry experience, while still learning important CS theory when not pressed for time. This not only benefits the students' future developments and security but will also reflect very positively on the school if things work out as anticipated.

6. FUTURE WORK

The next steps for this proposal would be to pilot some of these changes and evaluate outcomes to see if there are meaningful improvements. This could be accomplished gradually by introducing an internship development elective first for students, then seeing how much demand it has and how useful students find it. Another way this could be done is by having a subset of students complete the newly-proposed curriculum then measure how successful they are compared to peers on the traditional curriculum path. The primary success measurements would be students' preparedness, internships and post-graduation work success. Using these measurements and these proposed next steps, UVA will be able to evaluate the viability of the newly-proposed curriculum. Should good and statistically significant results emerge, it would be recommended expanding the pilot for a larger round of testing for extra measurements, or rolling it out into actual curriculum.

REFERENCES

Engineering.Virginia.edu. (2020). *B.S. in Computer Science | University of Virginia School of Engineering and Applied Science*. Virginia.edu. https://engineering.virginia.edu/department/computer-

science/academics/undergraduate-programs/bs-computer-science

Career.Virginia.edu. (2021). *Career Data - Student Outcomes | UVA Career Center*. (2021). Virginia.edu. https://career.virginia.edu/CareerData/StudentOutcomes

Durakovic, M. (2023, October 17). *10 hard-to-swallow truths they won't tell you about software engineer job*. MensurDurakovic.com. https://www.mensurdurakovic.com/hard-to-swallow-truths-they-wont-tell-you-about-software-engineer-job/

Hereford Sixth Form College. (2014). *Computer Science A Level*. Hereford.ac.uk. https://www.hereford.ac.uk/explore-courses/courses/computer-science/

Microsoft Education Team. (2023, March). *Why students need Computer Science to succeed | Microsoft Education Blog*. Microsoft Education Blog. https://www.microsoft.com/en-us/education/blog/2023/03/why-students-need-computer-science-to-succeed/

Reddit.com. (2023). *I feel like UVA does not prepare its CS majors enough for job searching*. Reddit.com. https://www.reddit.com/r/UVA/comments/18l6o8n/i_feel_like_uva_does_not_prepare_its_cs_majors/

Tay, Y. (2024, October 8). *Software Engineer interviews: Everything you need to prepare. | Tech Interview Handbook*. Tech Interview Handbook. www.techinterviewhandbook.org. https://www.techinterviewhandbook.org/software-engineering-interview-guide/

Tealhq.com. (2025). *2025 Computer Science intern resume example (+Free Template)*. Tealhq.com. https://www.tealhq.com/resume-example/computer-science-intern