Modernizing UVA CS Curriculum: Emphasizing AI

CS4991 Capstone Report, 2025

Sulav Regmi
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
yjb9kv@virgnia.edu

ABSTRACT
The University of Virginia's computer science program emphasizes low-level computing, requiring two separate courses: Computer Systems and Organization 1 and 2. I propose combining these into a single course allowing for machine learning to be introduced as a required class. This new curriculum would offer hardware-related topics as electives, ensuring interested students can still engage with such material. Implementing this change includes condensing the existing low-level computing content, mandating a machine learning course, and ensuring department approval to provide hardware electives. This restructuring would target current industry focus, providing students with education in modern fields such as artificial intelligence, while also remaining open for students interested in hardware. These additions should be implemented while observing changes in job opportunities and role placements for computer science graduates.

1. INTRODUCTION

The University of Virginia's computer science (CS) curriculum splits core topics into paired sequenced courses. This approach is used with topics such as algorithms, C-based programming, and discrete mathematics, separating introductory material from advanced concepts. While this approach is beneficial, it has also led to inefficiencies, particularly in the Computer Systems and Organization (CSO) sequence. The first course in this pair covers a variety of topics, jumping from bit operations to logic circuits before spending a large amount of time on assembly programming. Realistically, the only portion of the course that transitions directly into its succeeding one is the C programming at its end. The second course provides much of the crucial content that students need for understanding low-level computing. This uneven distribution of material suggests an opportunity to streamline the two courses into a single, more cohesive class.

Machine learning, one of the most transformative fields in computing, remains absent from UVA's required coursework. Beyond a brief introduction to machine learning and clustering, students receive little exposure to AI concepts unless they pursue electives.

Figure 1: Required ML Topics in DSA2

Figure 1 above illustrates how little importance is given to this topic; a mere two or three lectures are required by UVA's curriculum. Given the increasing demand for AI knowledge, it is incredibly important that UVA's CS curriculum adapts to reflect industry.

By streamlining and combining CSO, the department could introduce machine learning as a required course. This would occur without sacrificing hardware-related content, which could instead be offered as electives for interested students, the courses being similar to computer engineering. This restructuring would better align UVA's program with evolving industries, while still maintaining flexibility for students with different interests.

2. RELATED WORKS

Professors Mark Sherriff and Luther Tychonievich, architects of UVA's current CS curriculum, note their changes originated from an outdated department. Updates had not been made in two decades, and major complications arose with increasing enrollment and variations of degrees being available (Tychonievich and Sherriff, 2022). The eventual changes resulted in the previously discussed paired course and prerequisite system, this now being reflected in CS major requirements, as seen in Figure 2 below:

Foundation Courses

These courses are the next set of courses students take after finishing Introduction to Programming and comprise the set of prerequisites needed for upper-level courses. The 2000 level courses should be taken before the 3000 level courses and note that there are other prerequisites that govern the order that these courses should be taken. An example schedule can be found in the Degree Handouts and Resources section.

- CS 2100 - Data Structures and Algorithms 1 (Credits: 4)
- CS 2120 - Discrete Mathematics and Theory 1 (Credits: 3)
- CS 2130 - Computer Systems and Organization 1 (Credits: 4)
- CS 3100 - Data Structures and Algorithms 2 (Credits: 3)
- CS 3120 - Discrete Mathematics and Theory 2 (Credits: 3)
- CS 3130 - Computer Systems and Organization 2 (Credits: 4)
- CS 3140 - Software Development Essentials (Credits: 3)

Figure 2: Required CS Undergraduate Courses

The plan succeeded in updating the previous course system, in which many courses would overlap, and where course progression was unclear. However, these course requirements, as identified, lack any exposure to machine learning or AI concepts, resulting in a gap in knowledge that may not always be addressed with electives.

In contrast, Carnegie Mellon University (CMU, 2025), widely regarded as one of the most prestigious and rigorous CS programs in the United States, mandates AI coursework as part of its core curriculum, as shown in Figure 3 below:

One Artificial Intelligence elective (min. 9 units). Students will be able to tackle complex, real-world problems using techniques from Artificial Intelligence, including symbolic and probabilistic reasoning, machine learning, optimization, and perception.		
10-315	Introduction to Machine Learning (SCS Majors)	12
11-411	Natural Language Processing	12
11-485	Introduction to Deep Learning	9
15-281	Artificial Intelligence: Representation and Problem Solving	12
15-386	Neural Computation	9
16-384	Robot Kinematics and Dynamics	12
16-385	Computer Vision	12
others as designated by the CS Undergraduate Program		

Figure 3: Carnegie Mellon AI Required Coursework

CMU ensures that students are exposed to AI fundamentals, reinforcing its importance in modern computing.

Stanford University (2025) follows a specialized track system for CS students; yet even in the general track, AI remains a required component. Stanford also offers a dedicated AI track, further emphasizing the growing necessity of AI education. These examples highlight the need for UVA to integrate AI and machine learning into its required coursework to remain competitive and aligned with industry expectations.

3. PROPOSAL DESIGN

To implement the proposed curriculum change, the Computer Systems and Organization (CSO) sequence must be carefully combined to ensure students still gain fundamental low-level computing knowledge. By restructuring the sequence, machine learning can be introduced as a required course without sacrificing critical content, which will instead be offered as electives for students who wish to pursue these topics in greater depth.

The first step in this process is merging CSO 1 and CSO 2 into a single, more cohesive course.

The current two-course system covers a wide range of topics, but some of the content in CSO 1 does not directly connect to the more industry-applicable concepts in CSO 2. This is because much of CSO 2 is true C-based coding, achieving the understanding of key computing concepts such as functional page tables, memory caches and the use of threads. While CSO 2 does build on CSO 1, much of CSO 1 relates to CSO 2 only through UVA's desire to work from the smallest level of computing to the full picture throughout this lecture series. Realistically, practice with logic gates or two homework assignments about coding in binary are contextually useful when progressing to CSO 2, but are not skills that will be built upon in the second course. An example of these homework assignments can be seen below.

The following are assignment writeups:

- Bit fiddling
- Gates
- Binary coding 1: Mult
- Binary coding 2: Fib
- x86-64 Assembly
- Assembly Debugging
- Small C Functions
- Linked List in C
- Postfix Calculator
- Socket-based chat

Figure 4: Computer Systems and Organization 1 Assignments

A majority of these assignments have no application of C, this being the primary skill needed in CSO 2, as well as professional roles. While previous concepts of gates, binary and assembly are undeniably necessary to learn, some content could be trimmed to allow concepts from CSO 2 to be included, as well. While this transition would be difficult, introducing students to C earlier in the semester and progressing quickly into the applicable concepts in CSO 2 would greatly streamline the low-level computing education offered at UVA. More specialized topics, such as in-depth assembly programming and logic circuits, would be condensed or moved into elective courses for students with an interest in hardware-focused fields. This same reallocation of material would also be done to CSO 2, loosely-related units such as Networking and Cryptographic protocols removed but still accessible in the already-present Cyber-Security elective choices.

With space created by consolidating the CSO sequence, a machine learning course can be introduced as a core requirement. This course already exists at UVA, labeled CS 4710, but only has two sections in an average semester. By making this a required course, all CS students, regardless of specialization, would graduate with a basic understanding of AI concepts, making them more competitive for industry roles.

4. ANTICIPATED RESULTS

This proposed restructuring of UVA's computer science curriculum is expected to produce significant benefits both for students and the university. The major advantages of this change would be a stronger alignment with the curriculum of nationally recognized CS programs, as well as future industry demands. By introducing a required course, students will gain foundational AI knowledge, preparing them for careers in data science, AI development and other rapidly growing fields. As AI and machine learning continue to shape the future of computing, it falls to UVA to ensure their graduates are equipped with skills that translate to success with emerging technology.

Another expected benefit is a streamlined approach to teaching low-level computing. The current two-course system has overemphasized concepts that do not necessarily translate to practical application in later coursework or industry roles. The introduction of C earlier in the semester will allow for a smoother transition into the more applicable topics of CSO 2, ensuring that students develop skills that will be directly useful in their careers. A further benefit of this restructuring is the increased flexibility it offers students. By providing structured electives that supplement the core topics that remain in the new CSO course, the same knowledge is available to interested students.

Furthermore, this restructuring would help UVA maintain its competitive edge among top CS programs. Many leading universities, such as the previously mentioned CMU and Stanford, have already integrated and mandated AI exposure for students. This demonstrates how UVA benefits from this restructuring, an increase in academic recognition and success of graduates attracting more promising students.

5. CONCLUSION

This proposal presents an opportunity to modernize UVA's computer science curriculum by increasing AI-related content. By merging the Computer Systems and Organization sequence into a single, more efficient course, a required machine learning class can be implemented. This will ensure that all CS graduates will leave UVA with foundational AI knowledge. With the increasing relevancy AI has in industry and academia, students will be better prepared for their future careers.

Additionally, this restructuring would be done without sacrificing low-level computing knowledge in UVA's curriculum. The topics removed from the CSO sequence will remain present in electives for interested students. This proposed curriculum aligns UVA with top computer science programs while also addressing inefficiency in the current CSO sequence. With this change, UVA can provide students with a streamlined and industry applicable education.

6. FUTURE WORK

The next step in implementing this proposal involves finding an optimal way to merge content across the CSO courses. Although specific topics were previously mentioned, faculty and department leaders would need to be consulted to assess the logistics of this combination. The inclusion of experienced members of academia will prevent the exclusion of key learning outcomes, as well as allow collaboration for new elective offerings.

Continuous monitoring would also have to be conducted during this transition. This long-term observation should collect student outcomes, such as job placements or other career opportunities, this measuring the effectiveness of the new program. If successful, further emphasis on AI could be placed, changing UVA's curriculum to have greater specification on subtopics within the broad subject.

7. ACKNOWLEDGMENTS

I would like to thank Professors Mark Sherriff and Luther Tychonievich for their foundational work in structuring UVA's current CS curriculum, which informed this proposal.

REFERENCES

Stanford University. (2025). BS: degree requirements: Computer science. https://www.cs.stanford.edu/bachelors/degree-requirements

Tychonievich, L. and Sherriff, M. (2022). Engineering a complete curriculum overhaul. In Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1 (SIGCSE 2022), Vol. 1. Association for Computing Machinery, New York, NY, USA, 453–459. https://doi.org/10.1145/3478431.3499287

Carnegie Mellon University. (2025). *Undergraduate catalog*. Computer Science Program & lT. http://coursecatalog.web.cmu.edu/schools-colleges/schoolofcomputerscience/undergraduatecomputerscience/#bscurriculumtextcontainer