THE COMPUTER SCIENCE

SUCCESS BLUEPRINT

BY AMAN MANAZIR

@AmanManazir

TOP 5 CLASSES EVERY COMPUTER SCIENCE STUDENT SHOULD TAKE

You need to pick what classes to prioritize as part of your computer science degree. This guide will tell you what classes to take as well as which ones to avoid.

Programming I

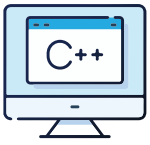
(Introduction to Java)

This class teaches fundamental computer science topics along with the process of incrementally developing small (200-500 line) programs. It was invaluable to me because it provided a rigorous study of computer science from first principles, despite my prior programming experience. I highly recommend starting with your university's entry-level computer science class instead of skipping ahead.

Programming III

(Data Structures and Algorithms)

The culmination of the programming fundamentals sequence, this intensive course covers data structures and algorithms. I firmly believe every STEM major should complete their university's introductory programming sequence, as it lays a strong foundation applicable to nearly any field.



Machine Organization and Programming

(Intro to C)

My introduction to C as well as the first (and only time) I ever touched Assembly in a computer science course, this class introduced me to memory management and so many important programming concepts. The problem with only knowing high-level languages is that you end up missing tons of low-level, important details — and C really teaches you those.

Operating Systems

Operating Systems was one of the final classes I took in my degree, and I can say without a doubt that it was my favorite class. I think it's safe to say that every single computer science student needs to take this class. You will discuss memory virtualization, concurrency, and storage, but more than that, it really teaches you how a computer works under the hood. I use these concepts every day as a software engineer. Don't let its reputation scare you — this class is 100% worth the time and effort.

Algorithms

This was one of the toughest classes I took in my degree, but one of the most worthwhile. The main advantage of an algorithms course is that it prepares you incredibly well for coding interviews. Before you take this class, you almost assuredly don't have a rigorous understanding of most algorithms, which can make LeetCode and online assessments difficult. However, after taking this course, you will be introduced to the mathematical foundations of many important algorithms which will allow you to practice them effectively for interviews.

EVERY OTHER CLASS, RANKED (BEST → WORST)

Look at the bottom of this list if you want to know what classes to avoid.

1. Programming III
2. Introduction to Artificial Intelligence
3. Intro to Computer Engineering
4. Introduction to Discrete Mathematics
5. Foundation of Mobile Systems and Applications
6. Introduction to Human-Computer Interaction
7. Software Engineering

CLASSES I WISH I TOOK

This list will give you a brief glimpse into very valuable classes that I didn't end up taking, mostly because I ran out of years of school. I would highly recommend these.

1. **Building User Interfaces**
2. **Introduction to Computer Networks**
3. **Database Management Systems: Design and Implementation**
4. **Introduction to Programming Languages and Compilers**
5. **Introduction to Software Security**

I attended the University of Wisconsin-Madison —
if you're interested in more details on all of these classes, visit this link:
https://guide.wisc.edu/courses/comp_sci/