

Considering CS

This handout is taken from the web site at <http://csmajor.stanford.edu/Considering.shtml>.

Considering CS?

Below is an introduction to the undergraduate CS major at Stanford. It briefly covers the program requirements, what you can do with a CS degree after graduation, and how to get started in the major. The following links might also be useful.

If you are a High School student considering becoming a CS major at Stanford, you might also want to check out the High School FAQ.

If you are enrolled at Stanford and considering majoring in CS but are not quite ready to declare, you can sign up for the `considering_cs` list. We forward selected departmental announcements, social events, and class changes to this list periodically.

If you are ready to declare, you may want to look at the pages on Choosing an Advisor or Declaring CS/CSE.

So you want to major in Computer Science?

In comparison to most academic disciplines, computer science is an unusual field. When Stanford was founded over 100 years ago, most of the scientific disciplines we study were already long established. The first computers, unbelievably primitive by today's standards, would not be developed for another 50 years, and it took another 25 years for computer science to become established as an independent discipline. Stanford's Computer Science Department was founded in 1965 and has consistently enjoyed the reputation of being one of the top computer science programs in the world.

In the last quarter of a century, progress in computer science has been enormously rapid, and computers have had a profoundly transforming impact on society. Computer science research has provided much of the intellectual underpinning and creative energy that have fueled that transformation, and it continues to be an extremely exciting field.

Exciting aspects of Stanford CS

Research

Undergraduates in CS have opportunities to get involved in research. With faculty and resources that are among the strongest in the world, a great deal of leading-edge academic research has been and continues to be carried out at Stanford. If you show that you can do the work, you can get involved in this type of research as an undergraduate, which will provide you with extraordinarily valuable training for future work in the field. The department also has a special program, called CURIS, which provides research opportunities for students during the summer.

Teaching

Undergraduates in CS have opportunities to get involved in teaching. The discussion sections for the introductory CS106 courses are led primarily by undergraduates. As a section leader, you will have the chance to teach the next generation of CS majors and get them excited about programming. If you have done well in your CS courses and can demonstrate both an aptitude for and an interest in teaching, you should contact the

coordinators of the program by sending mail to the CS198 Coordinators to arrange an interview.

Internships and Jobs

The Stanford CS department is right in the middle of Silicon Valley. The department has excellent connections to local companies (many of which were founded by people connected with the department). These connections provide help in getting summer internships as well as permanent positions after graduation.

Getting started

The usual first step in learning about computer science is to take CS106. Even if you don't end up majoring in CS, this class will serve you well in most engineering and scientific disciplines. Because students enter Stanford with varying backgrounds, there are two different paths through the CS106 course.

The standard introductory sequence for students without significant background in computing is to take CS106A followed by CS106B. For those with some experience in programming (or who have passed the CS AP exam), you can reach basically the same point by taking the one-quarter course, CS106X.

CS106X covers the same topics as CS106B, but with more in-depth coverage and at a faster pace. Note that it is also possible to take CS106X (rather than CS106B) after taking CS106A. Even if you've had a considerable amount of prior programming experience, you might still want to take CS106X, particularly if you do not know C++. By doing so, you will learn a lot more about how to develop well-engineered software and get a more solid foundation for more advanced work than you are likely to have from high school programming courses.

If you haven't done much programming before or don't feel comfortable with your programming skills, take the CS106A/B sequence. Don't let anyone tell you that "real engineers take CS106X." Many student who take the CS106A/B sequence go on to be very successful computer scientists, and the last thing you want to do is get in over your head.

No matter which course you take, you will learn about the C++ programming language along with many fundamental programming concepts and software engineering techniques.

Mathematical foundations

Along with the programming concepts covered in the CS106 courses, computer science requires the ability to think abstractly, often using the tools of mathematics. Unlike much of engineering, however, computer science depends heavily on discrete mathematics (as opposed to the continuous mathematics you study in calculus courses). Students take the CS103 course, which provides a solid foundation in the mathematics and theoretical topics needed for continued work in computer science.

The only prerequisite for taking CS103 is CS106A (or equivalent prior experience). As a result, you may want to consider taking CS103 early in your academic program to get more exposure to the theoretical foundations of computer science, which will give you a more holistic view of the discipline.

The CS major core

In addition to CS103 and CS106, the CS core is comprised of two additional courses in programming and systems (CS107 and CS110) as well as two additional courses in theoretical aspects of computing (CS109 and CS161).

Completing the CS major

After you complete the core courses, the CS program offers a variety of options for pursuing work in different areas within computer science. Students can choose to complete the requirements for any one track area within CS. The current tracks are listed below.

- Artificial Intelligence
- Biocomputation
- Graphics
- Human-Computer Interaction
- Information
- Systems
- Theory
- Unspecialized
- Individually Designed

For details about individual tracks and advice for choosing a track, see the tracks page. For more details about the specific academic requirements for each track, please refer to the relevant program sheet.

In addition to completing the requirements for a track (4-5 courses, depending on the track), students also have options for elective courses in the CS program, including both courses in the CS department as well as other related disciplines. In this way, students may engage in more multi-disciplinary work as part of their program in the CS major.

Requirements outline

Mathematics (26 Units)

- *Calculus* (MATH41, MATH42)
- *Mathematical Foundations of Computing* (CS103)
- *Introduction to Probability for Computer Scientists* (CS109)
- Two math electives from a restricted set

Science (11 Units)

- *Mechanics* (PHYSICS41)
- *Electricity and Magnetism* (PHYSICS43)
- One science elective from the School of Engineering list

Engineering Fundamentals (13 Units)

- *Programming Abstractions* (CS106B or CS106X)
- *Introductory Electronics* (ENGR40)
- One engineering fundamental elective from the School of Engineering list