Course Placement Information

Based on a handout by Eric Roberts and Mehran Sahami.
Thanks to Mehran Sahami, Cynthia Lee, Nick Parlante, and Patrick Young for their input!

The CS department offers many programming courses and it can be difficult to know exactly where to begin. This handout provides a general overview of these classes and some advice about selecting the class that's right for you.

If you have any questions, please feel free to come talk to the course staff! We'd be happy to offer one-on-one advice.

Getting Started in CS: CS106A

CS106A is our first-quarter programming course. If you're interested in learning how to program a computer, this is the place to start. CS106A has no prerequisites – it's open to everyone! Most students interested in learning computer programming will start here, including many eventual CS majors.

CS106A focuses on fundamental concepts in computer programming (`if` statements, methods, arrays, etc.) that arise in many different programming languages. Although CS106A is taught in Python, it is not designed as a class in standard Python programming.

In our experience, we've found that some high school CS classes are more rigorous than others. If you do have high school CS experience, we recommend reviewing the syllabus for CS106A to get a sense of what it covers. If you feel that you've seen all these topics before, you may want to start off in CS106B or CS106X. If you've covered some, but not all, of the topics from CS106A, you might want to look into CS106AX as a starting point. As always, feel free to ask the course staff if you have questions about where to begin!

We recommend that you take CS106A if

- You are interested in learning to program computers.
- You have little to no prior programming experience.

We recommend that you ***not*** take CS106A if

- You have prior programming experience at a level comparable to an introductory college course (for example, if you scored a 4 or 5 on the AP CS exam).
- You meet the criteria for taking CS106AX and are interested in a more accelerated treatment of the material.
- You have prior programming experience and want to learn how to program in Python.

In the 2019-2020 academic year, CS106A is offered every quarter.

Sharpening Your Prior Experience: CS106AX

If you have some prior programming experience but haven't covered everything that's normally explored in CS106A, then our CS106AX course may be for you. It functions as an introduction to programming in both JavaScript and Python, allowing for a deeper exploration of programming fundamentals than what's typically explored in CS106A. Think of it as a more intensive version of CS106A for folks who have some past experience – it's a great way to learn and solidify the fundamentals for future study, though it's not designed to serve specifically as a deep dive into either JavaScript or Python.

We recommend that you take CS106AX if

- You are interested in learning to program computers.
- You have some prior programming experience, but not at the level required for CS106B/X.

We recommend that you ***not*** take CS106AX if

- You have no prior programming experience.
- You have sufficient prior programming experience to jump right into CS106B or CS106X (for example, you've scored a 4 or 5 on the AP CS exam).
- You are taking it because you'd really like to take CS106A, but have a scheduling conflict that prevents you from doing so. *(Please don't do this! This is a very different class than CS106A.)*
- You have prior programming experience and are specifically interested in learning Python or JavaScript.

In the 2019-2020 academic year, CS106AX is offered in Fall and Spring quarters.

The Next Step: CS106B

CS106B is our second course in computer programming. It focuses on techniques for solving more complex problems than those covered in CS106A and for analyzing program efficiency. Specifically, it explores fundamental data types and data structures, recursive problem solving, graph algorithms, and basic algorithmic analysis.

CS106B assumes you have programming experience at the level of CS106A, though you don't necessarily have to have taken CS106A in order to take CS106B. If you've seen basic control structures (loops, `if` statements, etc.), variables, arrays, hash tables, and program decomposition, then you should be ready to take CS106B.

CS106B is taught using the C++ programming language, but uses C++ more as a vehicle for ideas than as an object of study. If you're interested in studying C++ in depth, consider looking into CS106L.

We recommend that you take CS106B if

- You have prior programming experience at the level of CS106A or CS106AX.
- You are interested in learning more about problem-solving with computers.
- You've programmed before but have not seen recursion, data structures, or algorithmic analysis.

We recommend that you ***not*** take CS106B if

- You already have completed equivalent coursework elsewhere.
- You have little to no prior programming experience.
- You have prior programming experience and want to specifically learn C++ programming.

In the 2019-2020 academic year, CS106B is offered every quarter.

The Next Step, And Then Some: CS106X

CS106X is the “honors” version of CS106B. It covers the same material as CS106B at a slightly faster pace, plus some additional topics not normally covered in CS106B. Depending on the offering, CS106X sometimes has an open-ended final project that culminates in a poster session. In the past, CS106X used to be CS106A and CS106B combined into a single quarter, but that is not currently the case.

You shouldn't feel pressure to take CS106X if you're thinking that at some point you might go on to major in computer science. Most CS majors go through the CS106A/B sequence, with only a small number of students opting to take CS106X. Rather, we'd recommend taking CS106X only if you're really interested in the material and want to explore it in more depth.

We recommend that you take CS106X if

- You meet all criteria for taking CS106B.
- You want to get a more in-depth treatment of the topics from CS106B.
- You are willing to put in more work than is necessary for CS106B.

We recommend that you ***not*** take CS106X if

- You have no programming experience and are hoping to complete the equivalent of CS106A/B in a single quarter.
- You are taking it because you'd really like to take CS106B, but have a scheduling conflict that prevents you from doing so. *(Please don’t do this! Most people who try going down this route end up in Frown Town.)*
- You really want to take CS106B, but are concerned that you need to take CS106X to avoid falling behind everyone else (trust us, that's not going to happen.)

In the 2019-2020 academic year, CS106X is offered in Fall quarter.

How it All Works: CS107

CS107 is designed as a first course in computer systems programming. It explores how high-level programming constructs are represented internally inside the computer and how those internal representations affect program behavior and performance. Along the way, it provides programming maturity and exposure to developing software in a Unix environment.

CS107 has either CS106B or CS106X as prerequisites and assumes an understanding of fundamental programming techniques and good programming style. As a result, it's rare for incoming students to jump directly into CS107 and to skip the CS106 series entirely. Typically, we'd only recommend this to students with a background comparable to CS106A/B and who already have good programming style. Most students, even those who go on to be CS majors, usually begin in CS106A/B/X.

We recommend that you take CS107 if

- You have completed CS106B or CS106X or have the equivalent programming background, including familiarity with recursion and fundamental data structures (binary trees, dynamic arrays, linked lists, graphs, etc.)
- You have experience writing readable code – writing comments, decomposing problems into smaller pieces, etc.

We recommend that you ***not*** take CS107 if

- You have never before taken a class in computer programming.
- You have prior programming experience, but have not met the postconditions of CS106B/X.

In the 2019-2020 academic year, CS107 is offered every quarter.

How it All Works on a Tiny Computer: CS107E

CS107E is version of CS107 that covers similar topics but which focuses on programming a small computer that can easily fit into the palm of your hand. The class is smaller and more project-oriented than CS107 and lets you play around with small embedded devices to see how low-level systems concepts directly let you control physical devices.

We recommend that you take CS107E if

- You meet all the requirements for CS107.
- You enjoy working on open-ended projects.

We recommend that you ***not*** take CS107E if

- You're nervous about taking CS107 and want to satisfy that requirement in a different way.

In the 2019-2020 academic year, CS107E is offered in Winter and Spring quarters.

An Alternative Introductory Course: CS105

If you're looking to learn more about computers or computing (for example, you want to learn how to make a web page, or how the Internet works, or how a computer is put together) without focusing too much on programming, you may want to try CS105. This course is designed to give a broader introduction to computing than CS106A at the expense of more significant programming depth.

We recommend that you take CS105 if

- You have no prior background in computer programming.
- You are interested in learning about computers and how programs control computers.
- You are interested in exploring programming in less depth than what's covered in CS106A.

We recommend that you ***not*** take CS105 if

- You know for certain that later on, you'll be taking CS106A.
- You have prior programming experience at the level of CS106A or higher.

In the 2019-2020 academic year, CS105 is offered in Fall and Spring quarters.

Learning a Language: CS106L

CS106L is a one-unit companion course to CS106B/X that focuses purely on the C++ programming language. Unlike CS106A and CS106B, which focus more on general programming skills and fundamental programming concepts, CS106L is specifically designed to focus on language features particular to C++ and how to use the C++ programming language to solve problems. Although CS106L is designed as a companion course to CS106B and CS106X, it's open to anyone with a comparable background.

We recommend that you take CS106L if

- You have prior programming experience at the level of CS106B or CS106X (or are currently enrolled in those courses.)
- You are interested in learning more about the C++ programming language.

We recommend that you ***not*** take CS106L if

- You want a deeper understanding of topics like recursion, data structures, or big-O notation.
- You want to learn programming at the level of CS106B or CS106X, but don't have the time to take those courses.

In the 2019-2020 academic year, CS106L is offered year-round.

An Alternative Terminal Course: CS106E

CS106E is a terminal course for non-CS majors who have already taken CS106A and want to have a solid understanding of computer technology in addition to the programming the other CS106 courses focus on. The course is particularly useful to anyone that might work in the tech industry in a non-programming capacity. The course covers topics such as how the internet works, what an operating system does, how websites are built, cloud computing, artificial intelligence, computer security, and privacy.

We recommend that you take CS106E if

- You meet all the requirements for taking CS106B.
- You are fairly certain that you won't be continuing onward to a CS major in the future.

We recommend that you ***not*** take CS106E if

- You are thinking about majoring in CS, but would like an easier alternative to CS106B/X.

In the 2019-2020 academic year, CS106E is offered in Spring quarter.