

Course Placement Information

With extensive updates by Keith Schwarz and Mehran Sahami.

CS 106A, CS 106B, CS 106L, CS 106X, CS 107, CS 101, CS 105, CS 107E, CS 142—there are a lot of programming classes to choose from, and it can be hard to know which ones to take! This handout provides a general overview of these classes and some advice about how to choose which class to take. If you're still unsure which class would be the best fit for you, please come talk to us—we'd be happy to help out!

CS 106A

CS 106A is our first-quarter programming course. If you're interested in learning how to program a computer, this is the place to start. CS 106A has no prerequisites whatsoever—you're not expected to have ever programmed a computer before—and is our recommendation for anyone who wants to learn to program and has no prior experience. CS 106A focuses on fundamental concepts in computer programming (including control structures, program decomposition, and standard data types) that arise in many different programming languages. Although CS 106A is taught using the Java programming language, it is not designed as a class in Java programming.

We recommend that you take CS 106A if

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

We recommend that you *not* take CS 106A if

- You have programming experience at around the level of the CS AP exam or prior college coursework in computer programming.
- You have prior programming experience and are considering CS 106A primarily as a way to learn Java.

CS 106B

CS 106B is our second course in computer programming. It assumes that you have programming experience at the level of CS 106A, though you don't necessarily have to have taken CS 106A in order to take CS 106B. CS 106B assumes that you're familiar with basic control structures (loops, `if` statements, etc.), variables, arrays, hash tables, and program decomposition. It explores fundamental data types and data structures, recursion, algorithmic analysis, graphs, and graph algorithms.

CS 106B is taught using the C++ programming language but is not designed to teach standard C++ programming techniques. If you're interested in studying C++ in depth, consider looking into CS 106L.

We recommend that you take CS 106B if

- You have prior programming experience at the level of CS 106A.

- You are interested in learning more about general computer problem-solving.
- You have programming experience, but have not seen recursion, data structures, or algorithmic analysis.

We recommend that you *not* take CS 106B if

- You already have completed equivalent coursework elsewhere.
- You have little to no prior programming experience.
- You have prior programming experience and are considering CS 106B primarily as a way to learn C++.

CS 106X

CS 106X is the “honors” version of CS 106B. It covers the same material as CS 106B at a slightly faster pace, plus some additional topics not normally covered in CS 106B. In the past, CS 106X used to be CS 106A and CS 106B combined into a single quarter, but that is not currently the case. You shouldn’t feel any pressure to take CS 106X if you’re thinking that at some point you might go on to major in computer science. Most CS majors go through the CS 106A/B sequence, with only a small number of students opting to take CS 106X.

We recommend that you take CS 106X if

- You meet all criteria for taking CS 106B.
- You really enjoy programming and want to get a more in-depth treatment of the concepts from CS 106B.

We recommend that you *not* take CS 106X if

- You are taking it because you’d really like to take CS 106B, but have a scheduling conflict that prevents you from doing so.
- You have no programming experience and are hoping to complete the equivalent of CS 106A/B in a single quarter.
- You really want to take CS 106B, but somehow imagine that you need to take CS 106X to avoid falling behind everyone else (trust us, that’s not going to happen.)

CS 106L

CS 106L is a one-unit companion course to CS 106B and CS 106X that focuses purely on the C++ programming language. Unlike CS 106A and CS 106B, which focus more on general programming skills and fundamental programming concepts, CS 106L is specifically designed to focus on language features particular to C++ and how to use the C++ programming language to solve problems. Although CS 106L is designed as a companion course to CS 106B and CS 106X, it’s open to anyone. Historically, many students take CS 106L take after having completed CS 106B or CS 106X.

We recommend that you take CS 106L if

- You have prior programming experience at the level of CS 106B or CS 106X (or are currently enrolled in those courses.)
- You are interested in learning more about the C++ programming language.

- You're interested in seeing aspects of C++ that are not covered in CS 106B or CS 106X.

We recommend that you *not* take CS 106L if

- You are interested in getting a deeper understanding of topics like recursion, data structures, and algorithmic analysis.
- You want to learn programming at the level of CS 106B or CS 106X, but don't have the time to take those courses.

CS 107

Although CS 107 is numbered so that it comes after CS 106, CS 107 is a very different style of class than CS 106. While CS 106A/B focus on general programming skills, CS 107 explores how the computer actually executes programs, what goes on internally inside the machine, how compilers work, and so on. In a few cases, it makes sense for incoming students to skip the CS 106 series entirely and jump directly into CS 107. Typically, we recommend this option only to students with a significant programming background, who have managed to develop good programming style, and who are comfortable with recursion, fundamental data structures, and algorithmic analysis. Most students—even those who go on to be CS majors—start off in one of the CS 106 courses. Each year only a dozen or so incoming students will start off in CS 107. For comparison, CS 106A typically enrolls about 1,800 students each year.

We recommend that you take CS 107 if

- You have completed CS 106B or CS 106X or have the equivalent programming background, including familiarity with recursion and fundamental data structures.

We recommend that you *not* take CS 107 if

- You have never before taken a class in computer programming.
- You have prior programming experience, but have never taken a college-level course in computer programming.

CS 107E

CS 107E is a new, experimental course that covers similar topics as CS 107 but focuses more on programming the computer “from the bare metal” (that is, directly interfacing with the hardware). The class is small and enrollment for the first offering is limited, so we recommend contacting the instructors directly if you're interested in taking this course.

We recommend that you take CS 107E if

- You meet all the qualifications for CS 107.
- You are interested in taking a new class from the systems faculty.
- You have very strong programming fundamentals.

CS 101 or CS 105

If you're looking to learn more about computers or computing (for example, you want to learn how to make a webpage, or how the internet works, or how a computer is put

together) but don't necessarily want to invest a lot of time learning how to program computers, you may want to look into CS 101 or CS 105. These courses are designed to give an introduction to computers that goes into more breadth than the CS 106 courses about the bigger picture. CS 101 and CS 105 do provide some programming experience, though not at the level of what's covered in CS 106A.

We recommend that you take CS 105 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 computer programming in less depth than what's covered in CS 106A.

We recommend that you **not** take CS 105 if

- You know for certain that later on, you'll be taking CS 106A.
- You have prior programming experience at the level of CS 106A.

CS 108

CS 108 is a class in object-oriented systems design. The class covers software engineering and software design techniques for working on large projects in groups. It's taught using Java and, unlike CS 106A, does go into some depth about the Java programming language. At the end of the course, you'll be asked to complete a final project where you'll build a large, complex software system from scratch.

We recommend that you take CS 108 if

- You have completed CS 106B or CS 106X or have the equivalent programming background, including familiarity with recursion and fundamental data structures.
- You are interested in learning how to design and test large software systems.
- You have prior programming experience and want to learn how to build large Java programs.

CS 142

CS 142 is a first course in web programming. This course, which has grown in popularity each year, gives a complete picture of how web applications are built, including front-end and back-end technologies, databases, and security concerns. This class assumes that you have a reasonable programming background, and we strongly recommend completing at least one of CS 107 or CS 108 before attempting this course.

We recommend that you take CS 142 if

- You are interested in learning how to design and build web applications.
- You have a programming background comparable to having completed at least one of CS 107 or CS 108.

We recommend that you **not** take CS 142 if

- You are currently enrolled in CS 106A/B/X or have not completed courses at a similar level.