

Course Placement Information

Based on a handout by Eric Roberts

Learning to program computers unlocks the full power of computer technology in a way that is both liberating and exciting. At the same time, programming is an intellectually challenging activity that comes easily to very few people. Taking a programming course requires a great deal of work and commitment on your part, but you will not be able to master programming without putting in that level of work somewhere along the way. The payoffs, however, are quite real. If you make the effort and keep up with the demands of the material, you will be able to make computers do amazing things. As you begin your journey in computing at Stanford, it would serve you well to decide what are the best options for you along this route.

What introductory programming course should I take?

A very large percentage of Stanford students take a programming course from the Computer Science Department at some point during their undergraduate career. Because we need to accommodate students with a wide range of backgrounds and interests, the CS department offers several different introductory classes:

- *CS 105—Introduction to Computing.* This course is designed as a general-education introduction to what this rapidly expanding field of computer science is all about. It attracts an audience of approximately 150 students a year, most of whom take the course primarily to meet the “Formal Reasoning” Ways of Thinking/Doing requirement. If your only interest is in meeting that requirement, CS 105 is likely to be the most appropriate course. Like any programming courses, CS 105 requires a reasonable amount of work, but not as much as CS 106A. CS 105 is offered this Spring quarter.
- *CS 106A—Programming Methodology.* This course is the largest of the introductory programming courses and is one of the largest courses at Stanford (taken by roughly 1,500 students last year). CS106A teaches the widely-used Python programming language along with good software engineering principles. The course is explicitly designed to appeal to humanists and social scientists as well as hard-core techies. In fact, most CS 106A graduates end up majoring outside of the School of Engineering. The course requires no previous background in programming, but does require considerable dedication and hard work.
- *CS 106B—Programming Abstractions.* This course is the natural successor to CS 106A and covers such advanced programming topics as recursion, algorithmic analysis, and data abstraction using the C++ programming language, which is similar to both C and Java. If you've taken the AP Computer Science A (not the CS Principles) exam and done well (scored 4 or 5) or earned a good grade in a college course, CS106B may be an appropriate course for you to start with. CS106B assumes that you already have familiarity with good programming style and software engineering issues (at the level of CS106A), and that you can use this understanding as

a foundation on which to tackle new topics in programming and data abstraction. CS106B is offered in Spring this year.

I already know how to program—shouldn't I skip the intro courses altogether?

Many students entering Stanford today have had considerable programming experience in high school or from their own independent work with computers. If you are in that position, the idea of starting with a beginning programming course—even an intensive one like CS 106B—seems like a waste of time. Your perception may in fact be correct. In my experience, there are at somewhere between 10 and 15 students in each entering class who should start at a more advanced point in the sequence. For most of you, however, the right place to start is with the CS 106 series. Most high-school computing courses are quite weak and provide very little background in modern software engineering techniques. By taking CS 106, you will learn how the CS department at Stanford approaches programming and get a solid foundation for more advanced work. If you're unsure where you should start the programming sequence, please talk with us.

Other courses

As computers become more powerful, it is possible to use them for increasingly sophisticated tasks without engaging in programming, at least in a traditional sense. The CS 106 courses teach you about programming, and not just about a particular programming language.

If you already have programming experience and want to learn about specific languages and tools, you should check out the following courses:

- *CS 142—Web Applications.* This course covers the concepts and techniques used in constructing interactive web applications, including the JavaScript. It requires prior programming experience at the level of CS107 and CS108. It is offered in Spring quarter this year.
- *CS 193C—Client-Side Internet Technologies.* This course covers such client-side web-oriented topics as JavaScript, HTML5, CSS, and Ajax. The course requires previous programming experience at the level of CS106A. It will likely only be offered during Summer quarter this year.
- *CS 193P—iOS Application Development.* As the name implies, this course charts the development path for iPhone and iPad applications. It requires prior programming experience (prerequisites: CS107 or CS108 as well as familiarity with the C programming language). It will be offered during Spring quarter this year.