Eric Roberts
CS 106A

Handout #1
January 4, 2016

Course Placement Information

Parts of this handout were written by Mehran Sahami.

Computers are everywhere in today's world (probably even in places you wouldn't want them to be). The more you know about computers, the better prepared you will be to make use of them in whatever field you choose to pursue. 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 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 range of backgrounds and interests, the department offers several different introductory classes:

- *CS 101—Introduction to Computing Principles*/*CS 105—Introduction to Computing*. These courses are designed as a general-education introduction to what this rapidly expanding field of computer science is all about. They attract an audience of approximately 300 students a year, most of whom take the courses primarily to meet the "Formal Reasoning" Ways Requirement. If your only interest is in meeting that requirement, CS 101 or CS 105 is likely to be the most appropriate course. Like any programming course, both CS 101 and CS 105 require a reasonable amount of work, but not as much as CS 106A. CS 101 will be offered in spring quarter, and CS 105 was offered only in autumn quarter this year.
- *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 1,600 students last year). CS 106A teaches the widely used Java 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. The course requires no previous background in programming, but does require considerable dedication and hard work. CS 106A is offered in autumn, winter, and spring quarters this year.
- *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 Computer Science AP exam and done well (scored 4 or 5) or earned a good grade in a college course, CS 106B may be an appropriate course for you to start with, but sometimes CS 106X may be a better

choice. CS 106B assumes that you already have familiarity with good programming style and software engineering issues (at the level of CS 106A), and that you can use this understanding as a foundation on which to tackle new topics in programming and data abstraction. CS 106B is offered in the autumn, winter, and spring this year.

- *CS 106X—Programming Abstractions (Accelerated).* CS106X currently operates as an "honors" version of our CS 106B course. It is taught using the C++ programming language and covers the same topics as CS 106B but with more in-depth coverage in some areas. In order to get through that much material in a quarter, CS 106X moves at a very fast pace. Students are expected to have solid background comparable to our CS 106A course and should have sufficient maturity and dedication to tackle an intense challenge. If you've had previous programming experience, this class is an excellent way to learn C++ and brush up on your skills. If you haven't done much programming before or don't feel comfortable with your programming skills, you should take the CS 106A/B sequence instead. Don't let anyone tell you that "real engineers take CS 106X." These days, most computer scientists and engineers start with CS 106A, where they do just fine. The last thing you want to do is get in over your head. If you think you are ready for CS 106X, you might want to see how you feel about the course expectations and pace after attending the first few classes. CS 106X is offered in autumn and winter quarters this year. Note: Students who have previously taken CS 106A and wish to enroll in CS 106X in place of CS 106B are free to do so. You just need be prepared for a fast-paced course.

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 106X—seems like a waste of time. Your perception may in fact be correct. In my experience, there are 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 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 me.

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 about a particular programming language. If your goal is to know more about how to use computers, you should investigate the following courses:

- *CS 1C—Introduction to Computing at Stanford.* This one-unit course is offered in the autumn quarter only and makes sure you have a level of "computer literacy" that will allow you to function effectively at Stanford. It does not teach programming at all.
- *CS 2C—Multimedia Production.* This course covers topics related to editing and publishing multimedia (sound, image, and video), including using applications like Photoshop and iMovie. Like CS 1C, this course does not teach programming and requires no programming experience.

If, on the other hand, 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. It requires substantial prior programming experience (prerequisites: CS 107 and CS 108). It will be offered in winter and spring quarters this year.
- *CS 193C—Client-Side Internet Technologies.* This course covers such client-side web-oriented topics as JavaScript, XHTML, CSS, XML, and Flash. The course requires previous programming experience at the level of CS106A. It will probably be offered only during summer quarter this year.
- *CS 193P—iPhone and iPad Application Programming.* As the name implies, this course charts the development path for iPhone and iPad applications. It requires prior programming experience (prerequisites: CS106B/X as well as familiarity with the C programming language). It is offered during autumn and spring quarters this year.