

Course Information

Instructor	Keith Schwarz
Email	htiek@cs.stanford.edu . Don't hesitate to send me emails! I enjoy helping out and answering questions. Even if you have a C++ question that pertains to something we didn't cover in class, send it to me and I'll be more than happy to answer.
Office Hours	While I don't have "office hours" in the conventional sense, I will try to stay around after lectures to answer questions. If this doesn't work out, send me an email and I can try to meet with you at another, more convenient time.
Lectures	Tuesdays and Thursdays, 4:15 – 5:05 in room Educ 128.
Units	1 unit satisfactory/no credit. However, that's one unit jam-packed with useful C++ and you will have a high ratio of content to units.
Prerequisites	This class is designed to be taken concurrently with CS106B or CS106X, but anyone with equivalent experience should be able to benefit greatly from the material. If you have completed CS106B/X in a previous quarter, that's fine as well.
Website	The course website is cs106l.stanford.edu and it's chock-full of announcements, handouts, and readings. I will upload all lecture code and handouts after each section, so don't worry about frantically copying down the code we'll be writing.
Readings	All of the course reading materials will be available online at the course website. I'll be posting relevant readings as the course progresses, so be sure to periodically check the course website.

C++ is an enormous language and while we'll cover a good deal of material in this course, there are several topics we won't have a chance to cover. If you're interested in getting some other perspectives on the material, the following books are excellent resources. You will not be disappointed if you check them out:

Accelerated C++: Practical Programming by Example by Andrew Koenig and Barbara Moo. This introductory C++ textbook is an invaluable resource for aspiring C++ programmers and is perhaps the best introductory C++ text available. While not an exhaustive reference, *Accelerated C++* will quickly get you up to speed on professional-level C++ topics.

Effective C++, *More Effective C++*, and *Effective STL* by Scott Meyers. These books are incredibly useful and will change the way you think and program in C++. The books in the *Effective C++* series are collections of useful pieces of advice, so you can quickly navigate to relevant sections. While Meyers' books are targeted at audiences with a solid understanding of core C++ language features, they should still be quite useful in the second half of the course.

The Design and Evolution of C++ by Bjarne Stroustrup. This most excellent book by the creator of C++ provides insight into the design decisions and overarching philosophy of C++. If you're interested in how the language came to be, this is the definitive resource.

In addition to the above books, I strongly recommend picking up a C++ language and library reference book for use in this class. Stroustrup's *The C++ Programming Language* is an excellent choice.

Grading

There will not be any exams or midterms in this class. Instead, there will be three small programming assignments that will exercise different aspects of the material. These assignments are designed to be about a quarter the size of a CS106B/X assignment, and hopefully will be quite entertaining. In order to receive credit for the class, you will need to complete the three assignments satisfactorily.

On some of the assignments, we will be holding IGs so that you can get some one-on-one feedback about your solution. These IGs are not mandatory, but I think you'll get a lot more out of this course if you decide to attend them.

Late Policy

I understand that this is a one unit class and that you are likely to have many other commitments, and so my late policy is reasonably flexible. If you feel like you need more time for any of the assignments, please let me know and we can try to work something out. I'm fairly generous about granting extensions, so don't hesitate to ask.

Honor Code

This one should be pretty simple. Don't copy someone else's programs, or post the solutions to any of the assignments online where everyone else can see them. You are free to discuss the assignments with other students, but your submissions **must** be your own work. Also, you **must** make a note in your submission if you collaborate with any other students. If you have any questions about what's permitted, feel free to send me an email.

Tentative Syllabus

C++ is an enormous language and unfortunately we will not have time to explore it in its entirety. However, there are several key aspects of the language that I believe will best prepare you to build large software systems in C++. Below is a tentative (and ambitious) class syllabus, which is subject to change based on how quickly we're able to move through the material:

Tuesday, March 30	Introduction to C++, Streams
Thursday, April 01	Streams
Tuesday, April 06	The Preprocessor and Multi-File Programs
Thursday, April 08	STL Sequence Containers Assignment 0 Out
Tuesday, April 13	STL Iterators and Associative Containers
Thursday, April 15	Example: CityFinder
Tuesday, April 20	STL Algorithms
Thursday, April 22	Example: Hearts Assignment 0 Due Assignment 1 Out
Tuesday, April 27	Beyond the STL with Boost
Thursday, April 29	Pointers and References

Tuesday, May 04	C Strings
Thursday, May 06	Example: Vector Assignment 1 Due
Tuesday, May 11	Templates
Thursday, May 13	const, Member Initializer Lists
Tuesday, May 18	Copy and Conversion Functions Assignment 2 Out
Thursday, May 20	Operator Overloading
Tuesday, May 25	Inheritance
Thursday, May 27	Inheritance
Tuesday, June 01	C++0x Assignment 2 Due