Course Information

Instructor	Keith Schwarz
Email	htiek@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 try to offer some advice.
Office Hours	While I don't have "office hours" in the conventional sense, I will 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	Mondays and Wednesdays, 4:15 – 5:05 in Hewlett 103.
Units	1 unit satisfactory/no credit. However, that's one unit jam-packed with useful C++ information and you will have a high ratio of content to units.
Prerequisites	Enrollment is limited to students in CS106B/X. If you are not enrolled in CS106B/X but are still interested in the material, you are still welcome (and encouraged to) attend lectures and read the lecture notes. The class assumes familiarity only with the C++ concepts taught in CS106B/X and should be appropriate for students of almost any programming background.
Readings	I will distribute handouts at each lecture that should serve as a good starting point for the material. However, we cannot possibly cover all of the C++ language or libraries in this class, so I strongly recommend that you pick up a reference text. If you're interested in supplemental readings, you can consider: *Accelerated C++: Practical Programming by Example* by Andrew Koenig and Barbara Moo. This is a great book that prioritizes the material quite well and gives a good treatment of some of the most important aspects of C++. *Effective C++*, *More Effective C++*, and *Effective STL* by Scott Meyers. These books are incredibly useful and can completely change the way you program in C++. They're a bit advanced and require a pretty solid understanding of the material, but they're well-worth the read. *C++: The Complete Reference* by Herbert Schildt. This is my personal "reference of choice," and has boatloads of information. It is a reference, not a textbook, but is a great source for complete information on various topics. *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++.
Website	The class website is http://www.stanford.edu/class/cs106b/cs106l/ and it's loaded with announcements and handouts. I will upload the lecture code and handouts after each section, so don't worry about frantically copying down the code I'm writing.

Grading

There will not be any exams or midterms in this class. Instead, there will be three small programming exercises. To receive a passing grade, you need to complete **two** out of the three with passing grades. This means that you can entirely skip an assignment without affecting your grade. Please try to complete assignments on time – while I don't have a formal late policy, I most likely will not accept late assignments. Since you only need to complete two of the three assignments, if you are unable to complete one of the assignments on time, you can just skip it and wait for the next assignment to roll around. If you are concerned about missing a deadline and want to discuss extensions with me, please send me an email.

Unlike CS106B/X, we will not hold interactive grading sessions to go over your assignments. However, if you are interested in discussing your grade with me, we can meet after class or by appointment.

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. I have absolutely no problem with you collaborating and working together – in fact, I encourage you to do so – but please make a note of it on your assignment before you turn it in. If you are ever concerned about whether something is permissible, please email me and ask.

Syllabus

The syllabus is subject to change based on how the class progresses. Here is our tentative goal:

Week 0: No class.
Week 1: IOStream library
Week 2: C strings and pointer arithmetic
Week 3: STL containers and iterators
Week 4: STL algorithms
Week 5: `const`, `static`, initializer lists, conversion constructors
Week 6: Exception handling
Week 7: Copy constructors and assignment operators
Week 8: Operator overloading, functional programming with the STL
Week 9: Inheritance, advanced C++ topics