CS106B: Programming Abstractions in C++
Course Information and Syllabus
Stanford University - Spring 2016

Instructor		TA	
name:	Cynthia Lee	**name:**	Nhien Tran
email:	cbl@stanford edu	**email:**	nhient@stanford.edu
office:	Gates 191	**office:**	Gates B02

Course Website

Main site: http://cs106b.stanford.edu/
Q&A forum: https://piazza.com/class/im56jsir5ii5jw (self-register)

Course Overview

Students coming to CS106B are expected to have substantial prior experience with programming constructs such as loops, functions/methods, arrays, console and file I/O, standard data types, and classes. In other words, students coming to CS106B are expected to know how to write code. In this course, we transform you from someone who knows how to write some code into someone who can wield the power of the computer in a more expansive, versatile, and fluent way. This course adds powerful new tools to your skillset: managing complexity through abstraction, using recursion to crack big, tough problems by breaking them into smaller solvable problems, and applying analytical tools to evaluate the efficiency of different software design choices.

Units

Undergraduates must take the course for 5 units. Graduate and professional students may opt for 3-5 units. Course requirements and grading are the same regardless of number of units.

Who Should I Email/Contact?

- **Miscellaneous basic policy questions** (When is the midterm? How many late days do I get?): Re-read this course information document and the course website. Any additional questions post to Piazza (unless sensitive or confidential).
- **Help with assignments or course topics:** Piazza or email SL.
- **Assignment special circumstances/requests for additional late days:** email TA (first use all your existing late days, then only once they are exhausted email the TA).
- **Anything very sensitive or confidential:** email TA or Cynthia Lee.
- I'd be happy to talk to any student any time in office hours regarding career or course advising, questions about research and other opportunities at Stanford, concerns about performance in the class, suggestions for improving the class, good CS-related jokes, etc. Homework help is also welcome in office hours. You don't really need a "reason" to come to office hours—I like to just get to meet as many of you as possible over the 10 weeks, so stopping by to introduce yourself is most welcome! I usually have high quality chocolate of some kind in my office, so that's reason enough, right?

Prerequisites

Substantial prior programming experience can take the form of strong performance in Stanford's CS106A or a rigorous high school AP CS course or equivalent. No extracurricular coding experience is necessary or expected, though students may be successful in 106B with that background and *without* having prior formal coursework. This course is taught in C++, but no prior exposure to C++ is expected. Most students come into 106B knowing Java, though others have experience in Python or other languages and not Java. Students whose prior experience is in non-object-oriented languages (*e.g.*, Matlab, Perl, C) may find that CS106A is better suited to their needs.

- **Extensive further advice on choosing between CS106A and CS106B, and selected other CS courses, is available from the CS106 course placement document:**

  http://stanford.edu/~cbl/SSEA/Course%20Placement.pdf

Topics

The following is our *approximate* course topics schedule for the quarter (subject to change). The terms on this list might not entirely make sense to you until we begin to cover them, but I list them here so that those who might be candidates for skipping CS106B entirely, or for whom other classes might more directly address their educational goals, can make that determination:

- Week 1: Transition to C++ (very quick, anticipating that most of the audience will have used Java and not C++); Get started on the Basic ADTs (see week 2)
- Week 2: Basic ADTs: Vector, Grid, Stack, Queue, Map, Set, Lexicon.
- Week 3: Recursion, advanced recursion, backtracking recursion
- Week 4: "Big O" mathematical performance analysis; Classes in C++
- Week 5: Pointers and dynamic memory allocation (new/delete); Linked lists
- Weeks 6-7: Basic ADTs revisited: how do we implement them behind the scenes? Linked lists, trees, Binary Search Trees, tries, hashing
- Week 8: Graphs, graph search algorithms (BFS, DFS, A*), minimum spanning tree algorithms
- Week 9: Inheritance and polymorphism in C++
- Week 10: Special topics, quarter wrap-up

Textbook

Roberts, Eric. *Programming Abstractions in C++*. ISBN 978-0133454840. Required. There is also a PDF version available for free: http://cs.stanford.edu/people/eroberts//CS106BX-Reader.pdf

Class Environment

CS106B is fast-paced and challenging, by design. Students coming from CS106A will notice a significant step up in complexity of the programming assignments and amount of individual design and planning required. These challenges exist because we want to effect the greatest possible increase in your skills and performance as a programmer in the short 10 weeks that we have together. To make that happen for *everyone*, I strive to create an inclusive and equitable classroom. I further depend on you to help each other obtain excellence rather than mistaking Stanford or this class for a zero-sum game. **Please do your part by seeking to promote the success of others, and by treating each other in ways that respect and celebrate the diversity of talent that is drawn to our exciting field of Computer Science.**

Here are a few specific things you should know about my policies on creating an inclusive and equitable class:

- **Preparation:** Perhaps more than many other subjects, students come to computer science with greatly varying previous exposure to programming and computer science. I want to assure students whose first real experience with programming was in CS106A that you are well prepared to succeed in CS106B. The same was true of the students who are now your section leaders!

- **Classroom environment:** For some reason, it seems unusually common in computer science classes that some students ask questions that are not really questions so much as opportunities to demonstrate knowledge of jargon or facts that are beyond the scope of the topic at hand. This can have a discouraging effect on other students who are not familiar with those terms and worry that this indicates that they are less prepared to do well in the class (note: this is rarely the case—knowing terms outside the scope of the course is not a good predictor of success). If you find yourself wanting to make such a question or comment in lecture, I encourage you to consider whether office hours would be a better venue for exploring that topic. I LOVE discussing things that are tangentially related to our class in office hours.
- **Office of Accessible Education accommodations:** If you have an OAE letter, please present it to me at your earliest convenience, so I can ensure that the course materials and staff support comply with your needs. I am always willing to do what it takes to support you, but I ask that you have your exam scheduling requests submitted no later than 1 week prior to the exam, because I must respect our room scheduling staff person's time to make those arrangements.
- **Preferred name and preferred pronouns:** I want you to be you in our class. You are always welcome to write your preferred name on all class assignments and exams. If you have a name and/or pronoun preference that doesn't match what our class roster gets from the registrar's office, please let me know and I will ensure we use that in our class.
- **Class expenses (textbook, device for class participation, etc.):** If obtaining any material for use in our class presents a financial hardship for you, please let me know and I will be happy to provide gift or loan items for you as needed.
- **Feedback:** Please do not hesitate to reach out to me, anonymously if you prefer, if any aspect of our course or class community can be improved.

Grades

Your overall course grade will be determined by **<u>your choice</u>** of the following two percentage breakdowns:

With lecture participation credit		*Without lecture participation credit*	
5%	section participation	5%	section participation
5%	**lecture participation**		
25%	midterm exam	25%	midterm exam
25%	final exam	**30%**	**final exam**
40%	homework assignments	40%	homework assignments

You are not required to formally notify me of your grading criteria choice; my spreadsheet that calculates final grades will simply apply the formula most favorable to you. Grades will be assigned *no stricter than* the usual scale of 90% = A- or better, 80% = B- or better, etc. A more generous curve may be adopted at the instructor's discretion, depending on exam difficulty or other factors.

Section Participation

You must sign up for a weekly 50-minute section, held on various times/places on Wed-Fri. **The section signup process will begin at the end of the first week of the quarter on the section leading website (http://cs198.stanford.edu).** After the signup period, a matching process is undertaken by the course staff, and your section assignments will be e-mailed out to you. Sections begin the **second** week of classes. Although Axess lists sections for this course, **<u>we don't look at Axess when assigning sections</u>**. If you miss the section signup period, there will be a second round chance for you to enroll in a section, but you will be limited to sections that are not full; if this happens to you, just wait and watch the section leading website (http://cs198.stanford.edu) for the second round to open.

Your section leader ("SL") will be your first point of contact for questions about homework or other course topics, grade your homework, provide individual guidance to you throughout the quarter. In section, your SL discuss sample problems in more detail than we can in lecture. Part of your course grade comes from attending and participating in your section and in code review meetings ("interactive grading" or "IGs") on a regular basis.

Lecture Participation

There will be frequent opportunities for participation during lecture: solving problems, responding with mobile app, and group discussion. **Lecture attendance is not required**, but **everyone present on a given day is expected to sit in groups of about three students and contribute to their group's discussions**.

This class operates more like a conversation than other college classes you may be used to.

Discussion questions with mobile app votes are used to help me gauge class understanding in real time (rather than waiting until the midterm) and tailor the class to your needs. To recognize the effort these votes require on your part, I offer **optional credit** for regular class participation. If you attend and participate actively in lecture (phone app *and* active group discussion), this can count to your final grade. In order to accommodate illnesses, vacation, sports competitions, family events, emergencies, etc., but still recognize your otherwise consistent efforts to participate, 6 days are automatically dropped from your score (there is no extra credit for <6 absences, absences in excess of 6 are scored as *days_attended*/(*total_days*-6)). **There is no need to email the instructor about the reason for your absence ☺**—it is automatically counted.

Exams

If you must miss an exam due to an unavoidable academic schedule conflict (an exam in another class at the same time), **you must contact the instructor immediately**. **There are no discretionary alternate exams. Family vacations, internships, job interviews, etc., are not allowable reasons to miss an exam.** Please note that University policy prohibits students from enrolling in courses with conflicting final exams, so we do **not give alternates for conflicting final exams (only midterms, because those are scheduled out of class time).**

106B exams are **open-book (the official course textbook), but closed-notes/handouts except one page of notes**. I know textbooks are expensive and that some students choose not to purchase one, so a few copies will be available for loan during the exam. A reference page will be included in the exam itself, containing commonly needed information (for example, documentation on the Stanford C++ library classes). The purpose of my open-book, closed-notes policy is to reduce the need for rote memorization of anything (this is a problem-solving course, not a memorization course), but still standardize the playing field in terms of precisely what information students have with them.

The final exam is comprehensive (covers topics from the entire quarter), if for no other reason than it would be impossible not to cover topics from the first half of the quarter because they are so fundamental to the subsequent material.

Homework Assignments

Homework consists of programming assignments done on an individual or pair basis. Pair programming is optional though encouraged on the assignments where it is allowed (some assignments, like the first assignment, are strictly individual). If you work on an assignment as a pair, you must submit as a pair, and you will be given the same grade.

Programs will be graded on both functionality and style. Rather than a direct point-based scoring system, a check-plus, check, check-minus grading system is used. Dividing the grades into these general categories means that your section leader can spend more of the interactive grading ("IG") time talking about what you need to learn from the assignment without being bogged down in accounting for minor point deductions. For each assignment, you must make an appointment for an IG session with your section leader. Your section leader will explain in section how to schedule these sessions and go over the grading process and standards in more detail.

From past experience, most grades will be ✓ or ✓+. The meanings of the grades are as follows, with *approximate* letter grade equivalents:

0 **Zero:** No credit. Typically this is only given if you do not submit any work on the assignment. *(F grade)*

– – **Minus-Minus:** Has extremely serious issues; shows almost no understanding *(D or F grade)*

– **Minus:** Has serious issues and does not demonstrate significant understanding. *(C or D grade)*

✓– **Check-Minus:** Has problems serious enough to fall short of the expectations/requirements for the assignment. *(B- grade)*

✓ **Check:** Meets the requirements for the assignment, with a few small problems or areas of improvement. *(B or B+ grade)*

✓+ **Check-Plus:** Satisfies all the requirements for the assignment, showing very solid functionality as well as good style. It reflects a job well done, so this is generally the score to aim for. *(A- or A grade)*

+, ++ **Plus, Plus-Plus:** Excellent; significantly exceeds our expectations for the assignment. To receive these grades, a program often reflects additional work beyond the requirements and/or solves the problem in a particularly elegant way. A mark of + or ++ will be given out only on rare occasions and should not be considered a standard goal for each assignment you submit. It is more of a way of congratulating the rare submission that goes above and beyond. *(A+ grade)*

An assignment's style score may be capped at a given maximum if its functionality score is particularly low.

Working in Pairs

Some the assignments must be completed on an individual basis, but the majority of them allow you to optionally work in a pair with a partner. Working in pairs can improve student learning by giving you someone to talk to when you are stuck, or by letting you see a different way of approaching the same problem.

If you choose to work with a partner, you must pair with another student who is currently taking the course and in your section. **If you have a friend you want to work with, request the same section or request a section swap if necessary**. Students working in a pair must be taking the course for the same grading basis. That is, a student taking the course Credit / No Credit may not pair with one taking it for a letter grade. Students auditing or sitting in on the course may not work in a pair with a student who is taking the course. No person who is not enrolled in the course may be part of any pair.

If you submit as a pair, each of you are expected to make a significant contribution toward solving that assignment. You should not claim to be part of a pair submission if you did not contribute significantly to help solve that program. Regardless of pairs, every student is still responsible for learning all course material. Of course, all exams are completed individually.

Many details about working in pairs are answered in a FAQ linked to from the class web site assignments page. Please read that page and make sure that you follow its guidelines.

Assignment Submission and Lateness

Submit your assignments electronically through a link provided on the course web site. Each student begins the course with **3 "late days"** for use on homework assignments. **Late days are intended to be used for emergencies that would typically warrant exceptions granted by the instructor:** illness, accidents, family emergencies, and the like. Please use care in deciding when to use your late days, because students who have consumed late days for reasons other than dire emergency will **not** be granted additional extensions, **even in cases of dire emergency**.

To use a late day, simply turn in the assignment as usual (do not email anyone).

A late day allows you to submit a program **up to one lecture day later without penalty**. For example, if a program is due on Monday at 1pm, using one late day allows you to submit it on Wedesnday at noon without penalty. If a program is due on Friday at 1pm, using a late day allows you to submit it on Monday at noon without penalty. **You may use up to 2 late days on any given assignment; you may not use all 3 on the same program.** Any fraction of a late day counts as one day. For example, if a program is due on Monday at 1pm, turning in the assignment at 5pm will consume an entire late day. If you turn in an assignment late and do not have any more late days, you will lose one bucket grade per late day (*i.e.*, per lecture day).

In no case may any assignment be turned in more than 3 lecture days late. Graders have deadlines for completing their work and it is not fair to them to delay their work more than this, so the assignment will be given zero.

At most one late day may be used on the very last assignment; you may not use more even if you have more late days remaining, and you may not turn in an assignment more than one day past the deadline for a deduction.

Help Resources

The department operates a student drop-in CS106 help service on the second floor of Old Union. Section leaders will be available to help students with problems. Please refer to the cs198.stanford.edu website for the most current schedule information.

Students are also invited to post questions (but **no source code**) to the Piazza web discussion board for our class. Emailing your SL is another great idea.

SCPD

Our class is being recorded for SCPD this quarter. SCPD is a distance learning program for working professionals. Of course, on campus students are also welcome to use the videos as a resource. A link is on the lectures page of the course website. SCPD students have a few special policy arrangements that apply only to them, as follows:

- **Section:** SCPD students are placed in a special SCPD section that takes place by video. Like other students, SCPD students should go to the cs198.stanford.edu website to register for a section during the section enrollment period that begins at the end of Week 1, but **at the cs198 website be sure to select the link for SCPD section sign-up**. Because weekly attendance at section and in-person participation in IGs is not possible, SCPD students are exempt from the 5% of the course grade for section participation (grade on remaining portion will be projected onto that 5%).
- **Class participation:** SCPD students may either opt out of lecture participation (as all students can), or they may get credit for lecture participation by a separate mechanism that will be explained by email to SCPD students (watch for it by the end of the first week of class).
- **Exams:** SCPD students are required to have a designated proctor arrangement that has been vetted with the SCPD office. You should plan to take exams with your proctor during the same time (in Pacific Time) as our actual exam. Understanding that some SCPD students may be located in different time zones, special handling instructions for this circumstance will be emailed to you (watch for it by the end of the first week of class).
- **Homework partners:** As explained above, some assignments allow partner work. The same rules apply to SCPD students as all other students, that is, SCPD students may choose to work with a partner from the same section. Because all SCPD students are enrolled in an SCPD section, in practice this means your partner must be another SCPD student. To identify a partner, you may use the Piazza partner finding tool.