

# CS106B Syllabus

Summer Quarter 2024. Lecture MTWTh 1:30 - 2:45 PM in Skilling Auditorium.

## Course Overview and Welcome

Hi there 🙋, and **welcome to CS106B: Programming Abstractions!** This is the second course in our introductory programming sequence. The prerequisite, CS106A, establishes a solid foundation in programming methodology and problem-solving in Python. With that under your belt, CS106B will acquaint you with the C++ programming language and introduce advanced programming techniques such as recursion, algorithm analysis, and data abstraction, explore classic data structures and algorithms, and give you practice applying these tools to solving complex problems. .

We're excited to share this great material with you and have a superb team of section leaders that will support you through the challenges to come. We hope you will find the time worth your investment and that you enjoy your growing mastery of the art of programming!

## Teaching Team



Instructor: Sean



Head TA: Julián

We have an incredible group of undergraduate section leaders (SLs) who lead weekly sections and help students 1-on-1 in LaIR hours. Read more about the [teaching team and office/LaIR hours](#).



Andrew



Anjali



Ben



Celeste



Connor



Devanshu



Emma



Eric



Felicity



Iona



Jasmyin



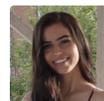
Katherine



Katie



Linda



Lucia



Nico



Rosman



Suzannah



Suze



Tina

## Course Topics

### Overview of Main Topics

Our planned curriculum will cover the following topics in this approximate order:

1. C++ basics
2. Data abstraction, classic ADTs
3. Recursion and backtracking
4. Classes and object-oriented programming
5. Pointers and dynamic memory
6. Linked data structures
7. Advanced algorithms

## Learning Goals

After completing CS106B, we hope you will have achieved the following learning goals:

- I am excited to use programming to solve real-world problems I encounter outside class.
- I recognize and understand common abstractions in computer science.
- I can identify programmatic concepts present in everyday technologies because I understand how computers process and organize information.
- I can break down complex problems into smaller subproblems by applying my algorithmic reasoning and recursive problem-solving skills.
- I can evaluate design tradeoffs when creating data structures and algorithms or utilizing them to implement technological solutions.

We'll also be giving you tools to tackle the following questions (note that these don't have single right or wrong answers!):

- What is possible with technology and code? What isn't possible?
- How can I use programming to solve problems that I otherwise would not be able to?
- What makes for a "good" algorithm or data structure? Why?
- Which problems should I solve with algorithms and data structures? What does a responsible programmer do when using data about real people?

## Prerequisites

The prerequisite for CS106B is completion of CS106A and readiness to move on to advanced programming topics. A comparable introductory programming course or experience (including high school AP courses) is often a reasonable substitute for Stanford's CS106A. If you are unsure if this course is the right for you, read [more about course placement](#).

## Course Structure

### Lectures

Lecture meets MTWTh 1:30 PM - 2:45 PM in Skilling Auditorium. Lecture recordings will be posted on Canvas for future review, but we ask that you treat regular lecture attendance as a professional obligation as an enrollee of this course. All students are expected to attend lectures in person if at all feasible (i.e., unless you are sick or an SCPD remote student), in order to fully participate in class discussions and other synchronous activities, as well as to stay on top of course material and get the most out of . We understand that this isn't possible for every student every time, so if you are unable to attend we ask that you at least watch the video of class before the next class, so you are up to speed on all course topics and important announcements. Read [more about lectures](#).

*Note about recording consent:* Video cameras are located in the back of the lecture room to capture the instructor presentation. These recordings might be reused in other Stanford courses, viewed by other Stanford students, faculty, or staff, or used for other education and research purposes. While the cameras are positioned with the intention of recording only the instructor, occasionally a part of your image or voice might be incidentally captured. If you have questions, please contact a member of the teaching team.

### Sections

Each student is assigned to a weekly small group discussion section, led by an undergraduate section leader. Your section leader is your mentor, grader, and personal connection to the greater CS106B course staff.

Sections begin the first week of classes, and attendance and participation are mandatory for all students. Your section leader will evaluate your section participation; this contributes to your course grade.

Read [more about section](#).

### Assignments

There will be regular assignments, about one per week. An assignment may include written problems, hands-on exercises with the tools, coding tasks and/or a larger complete program. Assignments are to be completed individually.

Programs are graded on "functionality" (is the program's behavior correct?) and "style" (is the code well-written and designed cleanly?). We use a bucket grading scale to focus attention on the qualitative rather than quantitative feedback. Read [more about assignments, grading, and late policy](#).

## Exams

We have mid-quarter and end-quarter exams. The mid-quarter is a check-in to assess your understanding of core topics covered in the first half of the course and help you chart a path forward. The final is a comprehensive assessment of your mastery of the course learning goals.

Mark these dates in your calendar now!

- **Midterm:** Friday, July 19, 1:30 - 3:30 PM
- **Final:** Friday, August 16, 3:30 - 6:30 PM

Read [more about exams](#).

## Course Grades

Final grades for the course will be determined using the following weights:

- **38%** Programming assignments
- **20%** Mid-quarter exam
- **35%** End-quarter final exam
- **5%** Section participation
- **2%** Quiz 0 on canvas

In order to receive a passing grade in the course, you must earn a passing grade on the programming assignments as well as (collectively) the exams (i.e., failing an exam will not disqualify you from passing the course, as long as your *average* across both exams is a passing grade). So, someone who has a passing exam average but a failing average on the programming assignments, or vice versa, will not receive a passing grade in the course.

## Units

If you are an undergraduate, you **must enroll in CS106B for 5 units** (this is by department and university policy, no exceptions). If you are a graduate student, you may enroll in CS106B for 3 or 4 units to reduce your units for administrative reasons. Taking the course for reduced units has no change on the course workload.

## Policies

### Incompletes

The university “I” grade (“incomplete”) is sometimes appropriate for circumstances of significant personal or family emergency disruption that occur late in the quarter and prevent a student from finishing course requirements on schedule. In order to be eligible for an Incomplete, University policy stipulates that a student must have completed a “substantial” part of the course work in “satisfactory” fashion. There must also be extenuating circumstances that warrant an extension of time beyond the end of the quarter to complete the remaining work. Approval for an incomplete is at the instructors’ discretion. Incompletes are not be considered for reasons such as poor performance in the course or over-commitment. Withdrawal from the course is more appropriate in those cases.

### Time Is of the Essence

We require timely notification if any circumstances arise that impact your ability to meet the requirements, deadlines, and expectations of the course in a way that our late assignment submission policy does not already accommodate – generally in advance of any impacted deadlines (including exam dates).

Generally speaking, we ask that you treat this course as you would any professional obligation. Think of project deadlines and exam dates the same way you would think of big project deadlines at work: if it became clear you weren't going to be able to meet such a deadline, or if you weren't going to have a presentation ready in time for a big quarterly staff meeting, your manager would want to know about that well in advance.

## Honor Code

As a student taking a Stanford course, you agree to abide by the Stanford Honor Code, and we expect you to read over and follow our specific [CS106B Honor Code policy](#). The work you submit for grading must be your own original, independent effort and must not be based on, guided by, or jointly developed with the work of others.

The CS department employs powerful automated plagiarism detection tools that compare assignment submissions with other submissions from the current and previous quarters, as well as related online resources. The tools also analyze your intermediate work, and we will run the tools on every assignment you submit.

The vast majority of you are here to learn and will do honest work for an honest grade. We celebrate and honor your commitment. Because it's important that all cases of academic dishonesty are identified for the sake of those playing by the rules, we will refer all cases of concern to the Office of Community Standards.

## Regret Clause

When people are under pressure, they sometimes do things they later regret. My impression from talking with students over the years is that this is often the case with students who violate the Honor Code in my classes: they're good people who make bad decisions they almost immediately regret. Accordingly, if you end up violating the Honor Code while working on an assignment in this course, you can mitigate the consequences by informing me of your infraction within 72 hours after the late deadline for the assignment. In response, the only penalty you will face for that infraction is a zero on the assignment. I will not lower your grade further as a result of the infraction, and I will not refer the matter to the Office of Community Standards.

To take advantage of this "regret clause," you must fill out the retraction form we will make available after each assignment deadline. We do not anticipate allowing retractions outside the 72-hour window for each assignment, and we will not do a last call for retractions at the end of the quarter.

This policy is modeled after the regret clause in [Harvard's CS50 class](#).

## Course Resources

### Textbook

Roberts, Eric. *Programming Abstractions in C++*. ISBN 978-0133454840.

You can find different options to [access the textbook here](#). Recommended readings for each lecture will be posted on our lecture schedule.

### Software

The official CS106 programming environment is **Qt Creator**, which is an editor bundled with C++ compiler and libraries. The software runs on Windows, Mac, and Linux and is free for personal/student use. The [Qt Installation Guide](#) has instructions for installing the tools onto your computer.

### Getting Help

We want to enable everyone to succeed in this course and offer different paths to help.

The instructors and Head TA will hold weekly office hours. The section leaders staff [LaIR](#) helper hours. The [CS106B Ed Discussion forum](#) allows public Q&A and discussion with your peers. Here is the [Quick Start Guide to using Ed](#).

### Accommodations

Students who need an academic accommodation based on the impact of a disability should initiate a request with the Office of Accessible Education. Professional staff will evaluate the request with required documentation, recommend reasonable accommodations, and prepare an Accommodation Letter dated in the current quarter. Students should contact the OAE as soon as possible since timely notice is needed to coordinate accommodations. The OAE has contact information on their web page: <http://oae.stanford.edu>. Once you obtain your OAE letter, please [send it to the head TA](#).

## Course Tools

The central place for CS106B resources is the course website right here at <https://cs106b.stanford.edu>. The website is your go-to for course materials (lectures, assignments, sections, exams) and course policies and information.

We also make use of these additional tools:

- [Canvas](#) to publish lecture videos and lecture quizzes.
  - [Ed Discussion forum](#) for online community, Q&A, and posted announcements.
  - [Paperless](#), our custom site for submitting assignments and viewing grading feedback.
  - [LaIR](#), our custom tool for managing the queue in helper hours.
  - [Gradescope](#), for viewing graded exams.
-