CS 61BL Summer 2019

About Piazza Resources Staff

About

Welcome to CS 61BL

The CS 61 series is an introduction to computer science, with particular emphasis on software and machines from a programmer's point of view. In CS 61A, students are exposed to a diverse set of mental models for computational problems and solutions through programming paradigms like functional programming, object-oriented programming, and declarative programming. In CS 61BL, we refine those mental representations by focusing particularly on the efficiency of writing programs (design) and running programs (runtime).

Hours and Workload

This is a summer course, which means it is run at twice the speed of a course during the normal semester. In addition, CS 61B naturally has more involved programming than CS 61A, and CS 61BL has even more, so expect there to be a lot of work. **We strongly recommend not taking any other courses alongside CS 61BL.**

While we can't predict how many hours you will spend on the course, here is a brief breakdown of what you will have to do in this course.

- 12 hours per week in lab.

- One two-hour lecture per week.
- Three exams in the 8 weeks of the course.
- Four projects in the 8 weeks of the course.

Prerequisites

CS 61A is an important prerequisite. We expect to build heavily on data-oriented and object-oriented design approaches introduced in this course and on algorithms for recursive list and tree manipulation. Engineering 7, CS 88, and Data 8 students may find the beginning of the course to be a bit scarier, particularly when it comes to object-oriented programming and recursion. We assume you are coming in with zero Java experience, but we will move through basic Java syntax very quickly.

We recommend all students to complete the optional introductory assignment by the beginning of the semester to get comfortable with Java and practice some of the programming skills expected by this class.

Alternatives

This is a course about data structures and programming methods. For those who may have already had a data structures course and simply want to learn Java, self-study may be a better option. The self-paced center offers CS 9G: Java for Programmers, that will teach you more of what you want to know in less time. Students with “sufficient partial credit” in CS 61B should consider taking CS 47B to complete the CS 61B requirement without taking the full course. Both of these self-paced courses are only offered during the fall and spring semesters.

Course Format

Labs

Lab sections meet every day Monday through Friday, except Wednesday. You are required to attend the lab section you are enrolled in, enforced through physical worksheets. All lab work must be done with a partner.

If you would like to attend a lab section that you are not officially enrolled in, you must receive permission from your current lab TA and the lab TA of the section you would like to attend. Historically, the afternoon labs (such as the 2-5 pm labs) have the highest attendance, so it’s unlikely that you will find a seat there. **Attending a lab that you are not**

officially enrolled in or have not received permission to attend will result in a loss of credit for that day's worksheet or quiz.

Lectures

There is one 110-minute lecture 3-5 pm every Wednesday. Attendance is strongly recommended, but will not be enforced.

Instructor Office Hours

Instructor office hours are primarily for short questions and administrative problems, but we're happy to make appointments by email for longer periods of time. The instructors are teachers too: we like teaching, and we'd rather see you as soon as you don't understand something than right before the exam.

In a lab-based course like CS 61BL, we expect that most of the course-related questions will occur in lab or our online forum, Piazza. For administrative inquiries like enrollment and DSP accommodations, email the course's administrative email address, cs61bl@berkeley.edu.

Assignments

Lab Exercises

Each lab assignment is due 22 hours after the start of your enrolled lab section (or of your switched section if have received approval from both the TA and the instructors). For example, if your lab section is from 8-11 AM, then Monday's lab assignment will be due on Tuesday at 6 AM. Similarly, Friday's lab assignment will be due on Saturday at 6 AM. Lab exercises and self-assessments will be graded via online submission to Gradescope, and you may submit as many times as you like before the deadline. The last lab of each week will additionally require you to fill out a self-reflection form about the labs in the past week.

On Gradescope, you may still receive credit on the lab assignment if you turn it in a few hours after your deadline. However, note that all deadlines will be retroactively enforced according to the section that we have on record at the end of the semester when grades are calculated.

Worksheets

Most labs will also have a physical worksheet, which should be completed during your lab section and submitted to your lab TA by the end of your section. Turning in the worksheet to your TA by the end of your section will mark your attendance for the day. Worksheets will be graded on correctness.

Quizzes

On Fridays, there will be a short written quiz administered in your lab section, with the intent of helping students and staff identify areas for improvement.

You can only take the quiz in your own lab section. If you need to take the quiz in another section, you must receive TA and instructor approval before doing so. There are no makeup quizzes.

Projects

There are four projects in this course. Projects are larger and more challenging assignments than you would typically find in a lab. All projects except for the first are to be done with your partner.

Exams

This course has two midterm exams and a third, final exam.

- Midterm 1 will be Monday, July 8, 6-8pm.
- Midterm 2 will be Monday, July 29, 6-8pm.
- The Final Exam will be Thursday, August 15, 6-9pm.

There will be no alternate exams. If you miss a midterm exam, your score will be reweighted with your performance on the final exam according to the exam supersession policy. Students with DSP accommodations that require alternate exam timing will be accounted for, so long as you can make a time that overlaps the official time. If you have a disability that prevents your ability to make such a time, we will discuss alternate arrangements with you directly.

If you have an exact time conflict with two or more midterm exams, or if you are unable to make the final exam, we suggest you seek accommodations with the conflicting party, or take CS 61B next semester. If, for whatever reason, you still need to take this course this semester, email cs61bl@berkeley.edu by the first week of the course explaining your needs. We are willing to accommodate, but expect that most students who miss exams will need to take an incomplete grade and complete the exams the next semester as there will be no alternate exams.

Exam Supersession

For those of you who miss an exam, have a bad night, or make major improvements over the semester, the exam supersession policy gives you a chance to replace one of your midterm exams. Specifically, if it helps your score, we will replace one of your midterm scores by its “final statistical equivalent” (FSE). We compute the FSE of an exam as follows:

Let F be the number of standard deviations above the mean that you score on the final. For example, if you are 0.3 standard deviations below the mean, $F = -0.3$. Let $\overline{M}$ be the class-wide mean (not including zeros) on a midterm, M. Let σM be the class-wide standard deviations (not including zeros) on the same midterm M. Your FSE for that exam is $\sigma M F + \overline{M}$.

If the FSE of midterm M is better than your original midterm M score, we will replace the original score with the FSE instead. If the FSE of all two midterm exams is better than each original score, then we will replace the one exam that gives you the greatest overall benefit. If all of the FSE are worse, nothing happens, so doing badly on the final won’t hurt your earlier exam scores.

Grading

Your course grade is computed using a point system with a total of 300 points.

Category	Percentage	Points
Labs	19%	57
Quizzes	5%	15
Project 0	3%	9
Project 1	6%	18
Project 2	12%	36
Project 3	10%	30
Midterm 1	10%	30
Midterm 2	15%	45
Final Exam	20%	60

Category	Percentage	Points
Total	100%	300

Grade breakdown

Each letter grade for the course corresponds to a range of scores:

A+	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F
295	280	265	245	225	205	195	185	175	170	165	160	0

Grade bins

There is no curve. Your grade will depend only on how well you do, and not on how well everyone else does. These bins were designed based on past semester student performance with the intention of complying with departmental grading guidelines.

Incomplete grades will be granted only for dire medical or personal emergencies that cause you to miss the final, and only if your work up to that point has been satisfactory. You must complete all coursework before the drop deadline to be considered for an incomplete grade.

There are a total of 22 graded lab exercises, each worth 3 points. For labs with a worksheet, the lab itself will be worth 2 points and the worksheet will be worth 1 point. For labs without a worksheet, the lab will be worth 3 points. Thus, while the total point value for labs is 66 points, lab grades are capped at 57 points. This means that you can miss up to 9 points on labs without penalty. In other words, getting more than 57 points on lab is equivalent to getting 57 points for your lab score. A similar idea will be applied to quizzes, which are each worth 3 points.

Late Policy

Labs

For lab assignments, you will receive a total of 4 slip days, with one slip day released to you every two weeks. If you use a slip day, your lab deadline is extended by 24 hours. You can use at most one slip day per lab assignment. You may not use a fraction of a slip day; for example, turning a lab 12 hours late will use 1 slip day, not 0.5 slip days. Lab assignments turned in late without using a slip day will receive no credit.

You should only be using a slip day on a lab assignment if you have made significant progress on the lab prior to the original deadline for the assignment. The intention of a slip day is to give you enough time to put the last finishing touches on the lab assignment before turning it in (for example, if you have been facing a bug for awhile, you should use a slip day, which buys you time to get some sleep before asking your TA for the last bit of help you need the next day). We recommend that slip days are not be used if significant progress wasn't made on the lab before the deadline, such as if you couldn't make a lab section and are trying to complete the entire lab the day after. In this situation, it may be better for you to consider skipping the assignment and proceeding with the more current ones to keep pace with the course.

To request to use a lab slip day, email your TA before the deadline of the assignment. If you have a slip day to use, then your TA will grant it for you.

Projects

For projects, you will not receive any slip days. However, you may turn in your project late and receive a penalty depending on when it is submitted:

- Within 24 hours of the deadline: 10% penalty
- Within 48 hours of the deadline: 50% penalty
- More than 48 hours after the deadline: 100% penalty (no credit)

For example, a project that scored 10 points on the autograder that was turned in 8 hours after the deadline will receive a 10% penalty; the final score will be 9 points. Similarly, the same project turned in 25 hours after the deadline will receive a 50% deduction, and the final score will be 5 points.

DSP

Students who normally receive additional time for assignments under their DSP accommodations should email cs61bl@berkeley.edu within the first week. **Any extensions requested for DSP reasons must be requested before the deadline of the assignment.**

Resources

Online Forum

Our discussion forum this semester will be Piazza. For most questions about the course, Piazza is the right place to ask them. The course staff read it regularly, so you will get a quick answer. Furthermore, by posting online as opposed to emailing us directly, other students benefit by seeing the question and the answer.

Staff Email

The email address cs61bl@berkeley.edu will send a message to the instructors. You can use it for correspondence that you don't want to send to Piazza. **Please do not email the instructors directly** since your message may be misplaced that way.

Reading

You are expected to read each lab's contents. There are also related textbooks that belong to CS 61B, which you may read if you please. The first text we use is Josh Hug's free, online course notes. If you find these notes insufficient, you might consider consulting Paul Hilfinger's (free) Java Reference or Head First Java, 2nd Edition by Sierra and Bates (O'Reilly, 2005). The optional textbook for the latter half of the course is Algorithms, 4th Edition by Wayne and Sedgewick.

Collaboration and Cheating

Plagiarism on any lab or project will result in a negative score on that assignment. A second instance of plagiarism on a lab or project will result in an F in the course. All incidents of plagiarism will be referred to the Office of Student Conduct, including carelessly leaving code up on GitHub.

Deadlines can be stressful, and we know that under extreme pressure, it becomes tempting to start rationalizing actions that you would otherwise would consider inappropriate. If you find yourself in this situation, please talk to a staff member immediately! See the late policy above.

During the Spring 2017 semester, we compiled a series of incident reports written by students who were caught plagiarizing. If you find yourself tempted to cheat, consider turning to the words of others who have made the wrong choice for guidance.

For labs and projects, if you commit some act that is unreasonable but bring it to the attention of the course staff within 72 hours, we may impose local sanctions such as assignment deductions, but the course will not refer the matter for further disciplinary action except in cases of repeated acts.

Lab Collaboration

The entire point of labs is to learn, so we emphasize active learning and pair programming. For labs, feel free to collaborate with others however you choose, though keep in mind that greater independence between pairs is likely to give you a better learning experience, as long as you aren't totally stuck. Even though we will allow close inter-pair collaboration on labs, **the code you share with your partner should still be your own work**!

Project Collaboration

By contrast, the projects were designed not just for learning (particularly how to be self-reliant in the context of large unfamiliar systems), but also for the dual purpose of evaluating your mastery of the course material. As such, they are intended to be completed primarily independently of any other pair (or any other student if it is a solo project), particularly when it comes to writing actual code. However, we encourage you to discuss high-level strategies, specific syntax issues, and solutions to bugs.

Exam Cheating

For exams, we will be absolutely unforgiving. Any incident will result in a failing grade for the course, though Berkeley will let you retake CS 61B next semester. All incidents of exam cheating will be referred to the Office of Student Conduct.

Academic Honesty

The golden rule of academic dishonesty is that you should not claim to be responsible for work that is not yours. To help (but not entirely define) the bounds of acceptable behavior, we have three important rules for projects:

- **By You (and Your Partner) Alone**: All project code that you submit (other than skeleton code) should be written by you and your partner alone, except for small snippets that solve tiny subproblems (examples in the Permitted section below).
- **Do Not Possess or Share Code**: Before you've submitted your final work for a project, you should never be in possession of solution code that you (or your partner) did not write. You will be equally culpable if you distribute such code to other students or future students of CS 61BL (within reason). **DO NOT GIVE ANYONE YOUR CODE! EVEN IF THEY ARE DESPERATELY ASKING. DO NOT POST SOLUTIONS TO PROJECTS ONLINE (on GitHub or anywhere else)!** If you're not sure what you're doing is OK, please ask.
- **Cite Your Sources**: When you receive significant assistance on a project from someone else, you should cite that assistance somewhere in your source code with the

`@source` tag. We leave it to you to decide what constitutes “significant”.

Permitted

- Discussion of approaches for solving a problem. Giving away or receiving significant conceptual ideas towards a problem solution. Such help should be cited as comments in your code. For the sake of others’ learning experience, we ask that you try not to give away anything juicy, and instead try to lead people to such solutions.
- Discussion of specific syntax issues and bugs in your code.
- Using small snippets of code that you find online for solving tiny problems such as code for finding the distance between two points. Such usages should be cited as comments in your code.

Permitted with Extreme Caution

- Looking at someone else’s project code to assist them with debugging. Typing or dictating code into someone else’s computer is a violation of the “By You Alone” rule.

- Working on a project alongside another person or group of people. Your code should not substantially resemble anyone else’s!

Absolutely Forbidden

- Looking at someone else’s project code to understand a particular idea or part of a project. This is strongly discouraged due to the danger of plagiarism. We are very serious about the “By You Alone” rule!
- Possessing project solution code that you did not write yourself or another student’s project code in any form before a final deadline, be it electronic or on paper. This includes the situation where you’re trying to help someone debug. Distributing such code is equally forbidden.
- Posting solution code to any assignment in a public place (e.g. a public git repository, mediafire, etched into stones above the Mediterranean, etc). This applies even after the semester is over.
- Working in lock-step with other students. Your workflow should not involve a group of people identifying, tackling, and effectively identically solving a sequence of subproblems.

You can obey the letter of this entire policy while completely violating its spirit. However, this policy is not a game to be defeated, and such circumventions will be seen as plagiarism.

A Parting Thought

Grades and penalties aren’t the purpose of this course; we really just want you to learn and be successful in the course.

The entire staff is very excited to be teaching CS 61BL this semester and we’re looking forward to meeting such a large and enthusiastic group of students. Welcome to CS 61BL!

Acknowledgements

Some course information ideas derived from Paul Hilfinger’s CS 61B handout, Josh Hug’s CS 61B syllabus, Brian Harvey’s CS 61A handout, and David Malan’s CS50.