

CS61B Summer 2006

Instructor: Erin Korber

1 Introduction

CS61B is the second course in the CS61 “Introduction to Computer Science” series. You will use the Java programming language to study programming techniques, data structures, and algorithms, and begin to investigate software engineering - the design and implementation of large programs.

This course will have a heavy emphasis on learning by doing - the best way to learn how to design and then write programs is, well, to design and write programs!

The course webpage is <http://inst.eecs.berkeley.edu/~cs61b/>. Up-to-date course information, assignments, and contact information for the course staff can always be found there.

General questions about the course may be posted in the `ucb.class.cs61b` newsgroup. The instructor and TAs check it regularly, and other students can help you too if you post there, as well as benefitting from the answers you receive. Should you wish to keep your question private, please email the course staff personally. Do not use the `cs61a@inst.eecs` account - it will not be checked regularly due to spam overload. You are encouraged to ask questions in office hours and discussion - frequently it is easier to explain something in a face-to-face conversation than over email. Should you be unable to attend office hours, please feel free to make (by email) an appointment at another time.

2 Prerequisites

CS61A is a listed prerequisite for this course. Although the registration system does not enforce prerequisites in the summer, it will still be assumed that you are acquainted with the material in 61A. If you have not taken 61A, you are still welcome to attempt this course, but you should be sure to have available a copy of the 61A textbook and other materials (an old

reader would probably be helpful) to refer to if necessary. Prior experience programming in Java is not needed or even expected.

You will need some basic familiarity with the Unix operating system and the emacs editor (e.g. that learned in 61A). Should you not be acquainted with them, some documentation will be made available on the course web-page. Please do also ask your fellow students and your TAs for help getting started, should you need it.

3 Administrivia

On the first day of class, in lab, you will be given a class account that you will use to complete and submit all of your work this semester - in lab that first day, you will make sure you can successfully log in, etc.

Access to Soda Hall and the lab (275) that you will be using is restricted after hours. In theory your registration for this class should enable your keycard (your Cal ID) within the first week of class; however, in practice, this rarely works. You may need to go to the CS Department front desk (387 Soda) and fill out a form to obtain keycard access.

If you are anything other than a Berkeley undergraduate, you will likely need some kind of special form signed by me in order to take this class. Please see me sometime in the first week after lecture about this.

4 Labs and Discussion Sections

Each section (there are 2) has 2 2-hour labs and 2 1-hour discussion sections per week. With the 6 hours of lecture, this means there are 12 hours of class time per week! This is, obviously, a lot of time. There is no official attendance policy, but the pace and workload of the course are set with the assumption that you will attend virtually all of the lectures, labs, and discussions. You are certainly free to choose not to do so, but no accommodations will be made for you, should you later decide that this was a bad idea.

Each day of lab, there will be an assignment for you to complete. These tasks are expected to take approximately 2 hours, though some (in particular the Wednesday labs) will be longer. Your lab work will be “checked off” by your TA each Monday. Completion of all lab work is strongly recommended if you wish to be able to successfully complete the projects.

Discussion sections will be devoted to answering any questions that have arisen about material from lecture or lab and working through examples

with your fellow students, with guidance from your TA.

Please attend the lab and discussion section in which you are officially enrolled, at least for the first week. Although we don't actually care about your attendance matching up with what Telebears says you are in, the near-fullness of the course at this time means that it is unlikely that there will be enough actual space for you in the other section.

5 Textbook and Lecture Notes

The only official textbook for this course is *Head First Java* by Sierra and Bates. This book (the first 11 chapters, to be specific) will be useful to you in the first 2 weeks of the course. This book is (as it states) not a reference book - online documentation will likely be more helpful for that purpose.

There is no official textbook for the remaining 3/4 of the course, during which we will be discussing data structures and algorithms. Should you wish to look at a book on this topic, I recommend Weiss' *Data Structures and Problem Solving Using Java*. At no time will I refer to any specific book on this topic - I mention this one only in case you wish to have something to read on the topic besides my notes.

I will be preparing detailed lecture notes for each day, paper copies of which will be distributed in class. They will be made available on the website after class, should you not be present. Extra copies of notes from previous days will also be available at lectures, in an effort to discourage printing more copies of the notes than is necessary. These notes will be a complete reference for the course content for the last 6 weeks, but should be considered as complementary to the textbook for the first 2 weeks.

6 Coursework and Grading

All work will be submitted electronically, and is due at midnight on the due date. Grades can be checked using the `glookup` program.

There will be a total of 230 points that you can earn in this course. Grades will be determined by a set scale, *not* on a curve, so your final grade is based only on your individual performance. Your letter grade will be determined by your total points as follows:

212	195	183	171	159	147	135	123	111	99	87	80	< 80
A+	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F

6.1 Homework and Labs

The homework and lab work for each week will be released each Monday. Each week, there will be a small homework assignment for you to complete and submit electronically. Homework will be due on Monday or Tuesday - check the schedule for details each week. Each week's lab work will be checked off in the following Monday's lab (so you have the weekend to finish it).

Homework will be graded on a 2-point scale: 0 points for failing to turn in the HW or submitting nonsense, 1 point for attempting but being incorrect, and 2 points for a correct solution. Each lab checkoff will also be graded on a 2-point scale. There will be 6 or 7 homeworks and 8 labs, which totals 28 or 30 possible points - however, the maximum possible total HW + lab score will be 25 points. This effectively allows you to have a late or missed lab/HW or 2 without penalty over the course of the term.

6.2 Projects

There will be 2 programming projects this term - a smaller (but still significant) first project and a more substantial second one. The single most important piece of advice on doing the projects is to figure out *what* to write *before* you write it. It is easy to waste time by just sitting down to write code before actually knowing how to solve the problem. It is impossible to write code to solve a problem that you do not in principle know how to solve yourself.

The first project will be worth 30 points and the second, 50 points.

6.3 Exams

There will be 3 exams over the course of the term, all of different sizes. The first will be a smaller exam covering just the OOP/Java material from the first 2 weeks of the course. The second will be a more standard "midterm exam" occurring around 2/3 of the way in to the term. The last day of lecture will be devoted to a final exam, which will be cumulative, but emphasizing material from later in the class.

The first exam will be worth 25 points, the second, 40 points, and the final, 60 points.

Should you believe that there is an error in grading your exam, write down a *brief* explanation of what you think the problem is, staple it to the front of your exam, and give it to the instructor or your TA within 3 class

days of when the exam was originally returned to you. The entire exam will be regraded.

6.4 Late Policy

Homework and labs submitted late will not receive any credit - the amount of points involved is sufficiently small that a missed HW/lab or two will not affect your grade in any serious way (especially since you can earn the full 25 points without having every single one in), and their value is diminished by not remaining in pace with the lectures.

Projects may be submitted late for a reduced grade - 5% of the total point value of the project will be subtracted for each hour (or fraction thereof) late. So if a project is worth 60 points, and you submit a solution worth 55 points 2 hours and 20 (or any number of) minutes late, your score is $55 - (60 * .15) = 46$ points.

If you believe that you have a good reason (e.g. a family or health-related emergency) that you need to submit a project late, please talk to the course staff early (i.e. WELL before the work is due), if the nature of your problem makes this possible. Being excessively busy (e.g. with other coursework) will not be considered a good reason. The condensed, fast-paced nature of summer courses makes good time management more essential than ever. You should expect this course to be very time-consuming and plan accordingly.

7 Collaboration and Cheating

Collaboration on homework and lab assignments is allowed and encouraged. Homework and labs are intended to help you learn the new material as it is presented in lecture, not as a serious evaluation tool - they will be graded only casually and count for a small fraction of your grade. Therefore, directly copying another student's work on these assignments without thinking about it yourself is pointless and will likely have a negative impact on your project and exam grades. You are certainly welcome to talk to each other about how to solve the problems and then do your own final writeups (carefully writing up a solution is a good way to make sure you understand the whole thing).

Both of the projects in this class should be only your own work - no group projects. You may discuss the project questions with each other before working out solutions, but we expect your solutions to be substantially different from your classmates'. At no point should anyone be looking at another student's project solution code.

It is ok to ask each other for help while working on a project *only* on issues unrelated to the project content (e.g. “what does error message x mean?” or “what’s the formula for the area of a circle again?”).

The first incident of cheating on a project or an exam will result in your earned grade on that item negated (so if 38 points would have been added to your grade otherwise, (-38) will be added instead). A second incident will result in (at best) failing the course.

Almost no one begins the term intending to cheat; rather, they resort to it out of desperation after getting behind. Should you feel that you are getting behind, please see the instructor or TA early - we want you to do as well as you can. Particularly in a condensed summer version of the course, it is important to take care of issues and confusions as they arise - the sooner you do it, the easier it is to handle.