

CS 61B: Data Structures and Programming Methodology

Instructor: David Sun,
360 Hearst Memorial Mining Building
davidsun@cs.berkeley.edu

Introduction

Welcome to CS 61B Summer session. CS 61B is the second course in the CS 61 series offered at EECS, UC Berkeley. This course will expose you to a range of programming techniques including data structures, interfaces, and algorithms for sorting and searching. Just as important, you will learn to think about the correctness of the programs you write as well as the tradeoffs in time and memory that arise from a variety of methods for structuring data. Finally, you will gain some appreciation for “software engineering” by designing and programming solutions for moderately complex problems.

The summer session runs for a total of 8 weeks, which is much shorter compared to a Spring or Fall semester. During this time, you will complete three projects, two midterms, and a final. CS 61B is a very demanding course even for a full length semester so expect there to be a lot of work.

This overview contains information about textbooks, labs and discussions, grading, and various policies that will be administered in this course. Please read it carefully.

Staff

The instructor is David Sun, davidsun@cs.berkeley.edu. The teaching assistants are Ben Blum, Adam Kirk, and George Wang. There will also be lab assistants working with us during the lab sections, and readers to grade your homework and projects. The instructor and teaching assistants will hold weekly office hours. Please consult the course website for contact details and our office hours. If the listed office hours are not convenient, you may make an appointment with any of us by email.

The course newsgroup at ucb.class.cs61b is a good place to ask general questions about the class. The teaching staff will monitor and respond to questions and concerns raised on the newsgroup. We will post announcements, clarifications, hints, and other information. It’s your responsibility to stay in touch with the information there.

If you would like to ask questions privately (for instance, if the question exposes your solution to a project or homework), you may send email to cs61b@cory.eecs.berkeley.edu. The email will be forwarded to all the TAs. Please prefix your email subject with the text “[CS 61B]” (e.g.

“[CS 61B] Midterm question”) to make it easier for us to pick out your message from the masses of emails that we receive each day.

Books

There are three required texts for the course, available at the campus bookstores:

- *Head First Java*, second edition, by Kathy Sierra and Bert Bates (OReilly, 2005);
- *Pragmatic Unit Testing in Java with JUnit*, by Andrew Hunt and David Thomas (The Pragmatic Bookshelf, 2004)
- *Objects, Abstraction, Data Structures and Design using Java 5.0*, by Elliot B. Koffman and Paul A.T. Wolfgang (Wiley, 2005)

Head First Java has been used in CS 61B for several semesters, so used copies should be readily available. *Objects, Abstraction, Data Structures and Design* also comes in a Desktop Edition. For 60% of the price of the hard-cover version, you can get a personalizable downloadable version. For more information, consult the textbooks Web site, <http://he-cda.wiley.com/WileyCDA/HigherEdTitle/productCd-0471692646, courseCd-CX1500.html>. There is no class reader.

Class Activities

Lectures

Lectures meet on Monday, Tuesday, Wednesday, and Thursday from 11:00 am to 12:30 pm in 306 Soda Hall. The lectures will not be webcasted for the summer session. Here is a tentative schedule of topics to be covered, project due dates, and exams. All the dates are subject to change. It is your responsibility to check the course website regularly for changes and updates:

Date	Event	Topics
6/23		Introduction, flow-of-control
6/24		Variables and objects
6/25		Classes
6/26		Arrays, JUnit
6/30		Loops and invariants
7/1		Lists
7/2		Inheritance and polymorphism
7/3		Interfaces and abstract classes
7/7		Exceptions
7/8		Java packages
7/9	Midterm I	
7/10		Big-O
7/14		Collections
7/15	Project 1 due	Stacks, queues, and trees
7/16		Trees and searching
7/17		Priority queues and heaps
7/21		Binary and balanced search trees
7/22		Binary and balanced search trees
7/23		Sorting algorithms
7/24	Project 2 Due	Sorting Algorithms
7/28		Maps and hashing
7/29		More hashing
7/30	Midterm II	
7/31		Generic programming
8/4		Threads
8/5		Game trees
8/6		Graphs
8/7		Graphs algorithms
8/11	Project 3 due	Dynamic programming
8/12		Garbage collection
8/13		Review
8/14	Final	

Labs and Discussion

Each week, you should attend two discussions and two lab sessions. Lab sections meet on Mondays and Wednesdays, and the discussion sections meet on Tuesdays and Thursdays, all after the morning lectures. Because of the limited number of workstations, you will be allowed entrance to a lab section only if Telebears lists you as enrolled. You may attend a section to which you are not enrolled only with the permission of both the relevant TAs. You may attend any discussion section, though you are encouraged to attend the section for which you are registered. If you cannot attend your registered discussion, try to at least be consistent in attending the same discussion – it will make it easier for your TA to know who you are and to help you.

Attendance to labs and discussion sections are **not mandatory**. However, we **strongly** encour-

age attendance – students who make effort to attend labs and discussions generally do better in the class. The sections are your opportunity to ask your TA about concepts covered in class and clarify homework/project questions. Homework assignments are typically built on top of lab exercises, so it is in your best interest to attend and keep up with lab sections. The second and third project will be done in partnership, so lab sections provide the opportunity for you to find potential partners.

Computer Accounts

All the labs for CS 61B are scheduled to be held in 275 Soda. Outside of class, you can use any computer that is not being used by another scheduled lab. To access the CS 61B Course Portal, goto <http://summer08.ucwise.org>. You should use a recent version of the Firefox browser, with a plugin that supports Java 1.5. Neither earlier versions of Mozilla nor any version of Internet Explorer will work well. There are a variety of free Java programming environments available, e.g. JBuilder, DrJava, NetBeans, and Eclipse. We will use Eclipse in lab. You may download it for your own computer from www.eclipse.org/downloads.

You will get an account on the EECS instructional computers in your first lab section. If you miss the first lab session, it's important that you contact your TA as soon as possible. You should change your password after receiving your account. We use your login information to determine when your work is submitted.

The Soda labs are open from 7:30am to 6pm Monday through Friday. Outside of these hours the doors to the building are locked. You will need to obtain card key access to use the labs by visiting 387 Soda.

Homework, Projects, and Exams

Homework assignments are small programming exercises that are to be done individually. Homeworks are typically assigned at the end of each lab section and build off the material from that section. Solutions are due before your next lab section. You will submit programming solutions online. All homeworks are equally weighted. Your worst two homework grades will be dropped.

Projects are an integral component of CS 61B. You will use what you learnt in class and sections to solve moderately complex problems. There are three projects: the first project will be an individual project, and the last two projects will be done in partnership. The projects are considerable work, so please start early!

There will be two midterms plus a final for the class. The midterms are held on: July 9 and July 30. The final is to be held on the last day of class on Aug 14. All exams are to be held during class. Midterms are two hours, from 11:00am - 1:00pm. Graded midterms will be returned in discussion sections. The final is three hours long, starting at 11:00am and finishing at 2:00pm.

Grading

There are a total of 200 points that you can earn towards your final grade. The various course activities will contribute points to your final grades as follows:

Activity	Points	% of Total Grade
All Projects	70	35
Homework	30	15
Midterm exam I	25	12.5
Midterm exam II	25	12.5
Final exam	50	25

Your letter grade is determined by total course points, as shown in the table below:

Points	Grade
185 - 200	A+
175 - 185	A
165 - 175	A-
155 - 165	B+
145 - 165	B
135 - 145	B-
125 - 135	C+
115 - 125	C
105 - 115	C-
95 - 105	D
85 - 95	D-
0 - 85	F

There is no curve for this class. Your grade will depend only on how you do, and not on how well everyone else does.

In complete grades will be granted only for dire medical or personal emergencies that cause you to miss the final exam, and only if your work up to that point is satisfactory.

Policies

Collaboration and Cheating

The Department and University takes cheating cases very seriously. Copying and presenting another persons work as your own constitutes cheating. Consistent with departmental policy, cheating on homeworks will automatically result in the maximum negative grade on that assignment, e.g., if

you cheated on a 20-point homework, you will get -20 points; cheating on projects or exams may result in a failing grade in the course. All incidents of cheating will be reported to the Office of Student Conduct where records of academic misconduct are kept throughout your undergraduate career.

Discussing class material with your peers and helping each other learning in the class are highly encouraged. Sometimes it can be hard to draw the line between cheating and appropriate collaboration. Obviously wrong is getting a homework service to write your program for you, or finding a solution on the Internet and submitting it as your own. On the other hand, providing suggestions to a classmate about the meaning of a question and offering advice about the likely meaning of a compiler error message are examples of interaction that we encourage. If you are not sure whether a particular interaction is appropriate, talk to the instructor or your TA before you submit the solution. Some useful rules of thumb, however, are the following:

- There is no reason that you should ever examine or be in possession of another student's solution or partial solution, either electronically or in hardcopy form. (We will call this the "no code rule".)
- You should never offer to give a solution or partial solution to another student, even with the explicit understanding that it will not be copied.
- You are not allowed to develop a single solution (with someone other than your partner for the project) and submitting copies (or modified versions) of that solution under multiple names.
- If you receive a significant idea from someone else, clearly acknowledge that student in your solution. Not only is this a good scholarly conduct, it also protects you from accusation of theft of your colleagues ideas.
- Programming assignments will be checked using advanced cheating-detection software. If you share code with another student, both of you will be caught, even if you take steps to hide your cheating.

Fortunately, we have found in the past that close interaction between students and the teaching staff serves to support student learning and reducing the need to cheat. Sometimes students cheat because they fall behind in the class and then panic when homeworks and projects are due. Remember that we are here to assist you in your learning, so don't be afraid to talk to one of the teaching staff, even if you are convinced that you are the last person in the class not understanding the material.

Lateness

Late work will not be accepted. Please do not ask for extensions for homework – each of these assignments is worth very few points.

Projects can be turned in late, but with a penalty. If the project is N hours late, your score will be reduced by $N/2$ percent. Please do not abuse this policy – it is there provide some leeway for you to handle emergencies. A project that is one day late will lose 12% of the total mark. A project

that is five days late will not earn you a passing grade, even if it's a perfect solution. During the course of the semester, however, we will give you a total of three late days (72 hours) for free. The free slip hours apply to project work only. For the second and third team project, the number of slip hours is the average of the remaining slip hours for you and your partner.

Regrades

If you believe your assignment or project submission was misgraded, email the reader who graded your work with a clear explanation of the problem. If you believe we misgraded questions on a midterm, return the paper to me (or your TA) with a written note of the problem on a separate piece of paper. Upon receiving a regrade request, the entire exam will be regraded, so be sure to check the solution to confirm that you will not lose more points, which has happened in the past. Requests for regrades must be made within two weeks after you receive the graded exam. By University policy, final exams may *not* be regraded.