



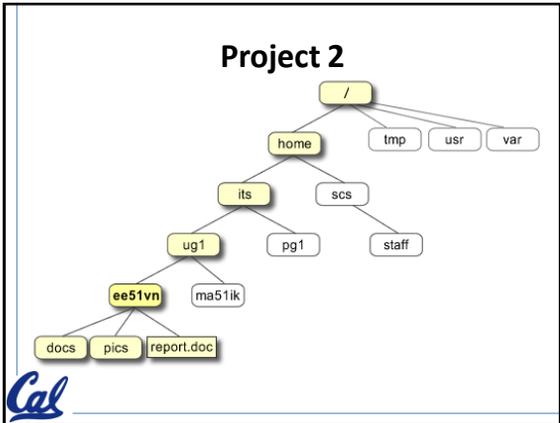
What's this course all about?

- <http://inst.eecs.berkeley.edu/~cs61b/su09/>

A screenshot of a music player window titled 'CPM: My Song'. It shows a musical staff with a treble clef and a simple melody consisting of several quarter notes.

Project 1

A 2x3 grid of six images showing a person wearing a mask that resembles a baby's face with large eyes and a wide mouth. The person is wearing a blue shirt and a hat.



Project 3

A photograph of a smartphone displaying a Tetris game. The screen shows a grid with colored blocks (red, blue, yellow) and a piece being placed.

Staff

 Instructor: Colleen Lewis colleen@berkeley.edu	 TA: Kashik Iyer cs61b-ib Section: 102	 TA: Jonathan Kotler cs61b-ja Section: 101
 Reader: Angela Juang cs61b-aj	 TA: George Wang cs61b-gw Section: 104	 TA: David Zeng cs61b-dz Section: 103

Lab Assistants (more to come!)



Stephanie Chou



Jason Ku



Dylan Scott



	Monday	Tuesday	Wednesday	Thursday	Friday
8:00-9:00					
9:00-10:00	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	
10:00-11:00					
11:00-12:00					
12:00-1:00	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	
1:00-2:00					
2:00-3:00					
3:00-4:00	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	
4:00-5:00					
5:00-6:00		Lecture 10 Evans - Collins	Office Hours 275 Soda - Collins	Lecture 10 Evans - Collins	
6:00-7:00					
7:00-8:00	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	
8:00-9:00					

Lectures

	Monday	Tuesday	Wednesday	Thursday	Friday
8:00-9:00					
9:00-10:00	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	
10:00-11:00					
11:00-12:00					
12:00-1:00	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	
1:00-2:00					
2:00-3:00					
3:00-4:00	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	
4:00-5:00					
5:00-6:00		Lecture 10 Evans - Collins	Office Hours 275 Soda - Collins	Lecture 10 Evans - Collins	
6:00-7:00					
7:00-8:00	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	
8:00-9:00					

Tuesday/Thursday 5:10-6:00
10 Evans



Labs

	Monday	Tuesday	Wednesday	Thursday	Friday
8:00-9:00					
9:00-10:00	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	101 LAB 275 Soda - Jonathan	
10:00-11:00					
11:00-12:00					
12:00-1:00	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	102 LAB 275 Soda - Kanishk	
1:00-2:00					
2:00-3:00					
3:00-4:00	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	103 LAB 275 Soda - David	
4:00-5:00					
5:00-6:00		Lecture 10 Evans - Collins	Office Hours 275 Soda - Collins	Lecture 10 Evans - Collins	
6:00-7:00					
7:00-8:00	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	104 LAB 275 Soda - George	
8:00-9:00					

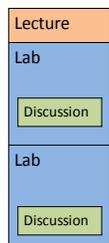
- Online curriculum: ucwise.org
- During lab
 - Take the quiz online (capped at 70%)
 - Do online lab activities
 - Work with a partner
 - Participate in discussion section
 - Get help from other students, lab assistants & TAs



What's the difference between CS61B and CS61BL?



1 week
in
CS61B



1 week
in
CS61BL

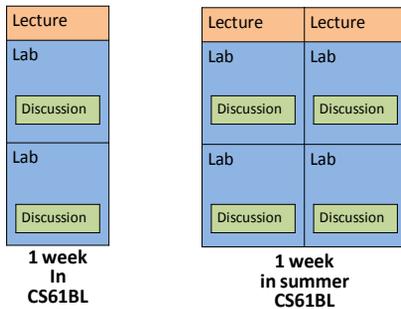


Why are we doing lab based?

- Community
 - Learn by doing
 - Interviewing practice
 - Extra pair of eyes (or ears)
 - Frequent feedback/support
- Lecture Notes:
 - <http://www.cs.berkeley.edu/~jrs/61b/>
 - Lecture Videos:
 - http://webcast.berkeley.edu/course_details.php?seriesid=1906978343



How is summer session different?



Do I have to come to lab?

- Yes - to get credit for the quiz!



Books

- **Head First Java**, second edition, by Kathy Sierra and Bert Bates (O'Reilly, 2005);
- **Pragmatic Unit Testing in Java with JUnit**, by Andrew Hunt and David Thomas (The Pragmatic Bookshelf, 2004);
- **Data Structures and Abstractions with Java**, second edition, by Frank M. Carrano (Prentice Hall, 2007).



Reading Assignments

Calendar (may change)

- HFJ: Head First Java
- PUT: Pragmatic Unit Testing
- DSA: Data Structures and Abstractions with Java

Wk	Date	Topic	Homework Due (before your lab begins)	Reading Due
1	6-22 M	Welcome and intro to Java		
	6-23 Tu	Loops and Conditionals LECTURE		HFJ ch 1 & ch 2 DSA 10.1-10.14 Scheme to Java General info about CS61BL
	6-24 W	Java Objects	Homework 01	HFJ ch 3, ch 4 & ch 9 except pages 250, 255 PUT ch 1 & ch 2 DSA 10.15-10.19 Boxes and Arrays
	6-25 Th	Objects and Arrays, Testing LECTURE	Homework 02	HFJ ch 5
	6-29 M	Arrays and Collection Classes	Homework 03	HFJ pages 273-287 and 292-293 PUT 3.1-3.3 & ch 4 & ch 5
	6-30 Tu	Loop Data Invariants and Iterators	Homework 04	HFJ ch 6 & ch 10 & appendix B: sec. 2 & 7



Exams

- **Midterm 1:** Tuesday July 7th,
–5-6pm in 10 Evans
- **Midterm 2:** Tuesday July 28th,
–5-6pm in 10 Evans
- **Final:** Thursday August 13th,
–5-8pm in 10 Evans



Review Sessions

- **Review 1:** Sunday July 5th,
–1-4pm in 306 Soda
- **Review 2:** Saturday July 25th,
–1-4pm in 306 Soda
- **Review 3:** Sunday August 9th,
–1-4pm in 306 Soda



Projects

- **Project 1** (individual):
–due Monday July 13th, 10pm
- **Project 2** (with 0 or 1 partner):
–due Wednesday July 22nd, 10pm
- **Project 3**(with 1 or 2 partners):
–due Tuesday August 11th, 10pm



inst.eecs.berkeley.edu/~cs61b/su09/



Homework

Assigned:	Due:
Monday during lab	Wednesday before your lab
Tuesday during lab	Thursday before your lab
Wednesday during lab	Monday before your lab
Thursday during lab	Tuesday before your lab



Grading

Assignment category	# points	percent
All projects	36	18%
All other homework scaled to:	24	12%
All quizzes scaled to:	20	10%
Midterm 1	24	12%
Midterm 2	36	18%
Final	60	30%



Grades:

There is no curve!

Points	Grade
190-200	A+
180-190	A
170-180	A-
160-170	B+
150-160	B
140-150	B-
130-140	C+
120-130	C
110-120	C-
100-110	D+
90-100	D
80-90	D-
<80	F



- On every homework assignment and project we will be running cheating detection software to compare your solution with the solution of your classmates and with all previous submissions.
- If you're falling behind at all – please talk to Colleen or your TA!



Cheating versus Collaboration

- **Obvious rules:** You can't get a solution from the internet by any method. Don't copy a current or previous student's solution.



Cal

Cheating versus Collaboration

- **No code rule:** You should never have any part of a current or previous student's homework or project solution in your possession, either electronically or in hardcopy form.
 - **Lab collaboration clarification:** In lab you will write a lot of programs that you will never turn in for homework or a project. For a program that is not turned in, we encourage you to write code with a partner on one computer and then share that collaboratively written code. Always check the day's homework assignment to make sure you don't need to turn that code in. If you do need to turn it in, you can talk, but you can not share code.

Cal

Cheating versus Collaboration

- **Reusing code:** You may reuse code that you have written. The first time you are writing the code if it is for a homework or project you **MUST** write it yourself. If not, you may work together with a partner. If you would like to reuse some code that you and your partner wrote in a regular lab that was not part of a homework or project, you may reuse that code. Please make a note whenever this is the case.

Cal

Cheating versus Collaboration

- **Looking at code rule:** You can help a classmate with bugs in their homework or projects and in doing so you can look at their code. Don't copy their code that you see – all work must be your own and all work must be their own.
 - **Don't show your own:** If you're helping a classmate find a bug, it is **NOT** okay to show them your code – whether it is working or not. You should focus only the buggy code in question.

Cal

Cheating versus Collaboration

- **Debugging clarification:** You can help someone if they have a logical error in their code. You should:
 - Never touch their keyboard or mouse.
 - Ask them lots of questions! Examples:
 - If they have a run time error:
 - » What is happening on the line indicated by the run-time error? What are you trying to do?
 - » How could that error happen on that line? What would need to be true? How could those conditions be met?

Cal

Cheating versus Collaboration

- **Debugging clarification:** You can help someone if they have a logical error in their code. You should:
 - Never touch their keyboard or mouse.
 - Ask them lots of questions! Examples:
 - If the program doesn't do the right thing:
 - » What test case(s) demonstrates the error?
 - » Can you come up with a simpler test that demonstrates the error?
 - » Can you trace through by-hand a simple example that demonstrates the error?
 - » Have you run the debugger on the simple test to see when the error is introduced?
 - » Have you tested the sub-procedures to make sure that they work?

Cal

Cheating versus Collaboration

- **Compiling clarification:** If their code doesn't compile you can help! You should:
 - Never touch their keyboard or mouse.
 - Try to help them figure it out for themselves. You should just be *helping*.
 - Talk about each syntax error to make sure that it makes sense why the compiler was complaining.

 Cheating



Cheating versus Collaboration

- **When in doubt – ask rule:** If you are not sure whether a particular interaction is appropriate, talk to Colleen Lewis or your lab TA before you submit the solution.

 Cheating



Cheating versus Collaboration

- **Give credit rule:** 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.

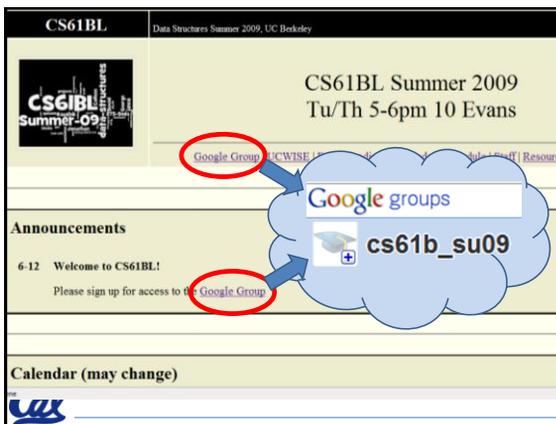
 Cheating



Waitlist

- To be enrolled for the course you need to enroll for a section that is not full.
 - This means the 8-11am or 6-9pm labs
 - Do this ASAP!
- If a spot opens up in a the 11-2 or 2-5 sections you can switch into that section.
- Please don't over-crowd the 11-2 or 2-5 sections!





CS61BL. Data Structures Summer 2009, UC Berkeley

CS61BL Summer 2009
Tu/Th 5-6pm 10 Evans

Google Group [PCWISE] [Resources]

Announcements

6-12 Welcome to CS61BL!
Please sign up for access to the Google Group

Calendar (may change)



CS Rocks!



Welcome to CS61BL!

