CS61c Summer 2014 Syllabus

1 Course Staff and Schedule

Instructor:

Alan Christopher
alanchristopher@berkeley.edu
OH: F 10-12; 651 Soda

GSIs:

David Adams
dadams@berkeley.edu
OH: M 2-3, W 2-3; 611 Soda

Fred Hong
fredhong@berkeley.edu
OH: T 2-3, Th 2-3; 611 Soda

Hokeun Kim
hokeunkim@berkeley.edu
OH: W 4-5, Th 5-6; 611 Soda

Kevin Liston
kmliston@berkeley.edu
OH: M 12-2, W 12-2; 611 Soda

Andrew Luo
andrewluo_cs61c@outlook.com
OH: F 7-9; 411 Soda

Lectures: MTuWTh from 9:30-11:00am in 100 Lewis.

Discussions: MW in 3 Evans, 9 Evans, and 310 Hearst Mining. Feel free to attend any discussion sections you like (you can even attend multiple discussions).

Labs: TuTh in 330 Soda, 200 SDH. It is important that you only attend your own lab section – the number of computers in the lab is limited and we need to ensure that there are always enough computers available.

2 Course Description

CS61C teaches the internal organization and operation of digital computers with a new emphasis on advances in parallelism. The subjects covered include C and assembly language programming, memory management, how higher level programs are translated into machine language, computer organization, caches, performance measurement, parallelism, CPU design, warehouse scale computing, and related topics.

Prerequisites: CS61a and CS61b.

3 Course Resources

Webpage:	http://inst.eecs.berkeley.edu/~cs61c/su14
Forum:	http://piazza.com/class#summer2014/cs61c
Textbooks:	*Computer Organization and Design: The Hardware/Software Interface* (5th Edition) by Patterson and Hennessy, ISBN 0124077269 *The C Programming Language* (2nd Edition) by Kernighan and Ritchie, ISBN 0131103628 *The Datacenter as a Computer* (1st Edition) by Barroso and Holzle. Available for free online.

4 Assignments and Grading

This course will be graded on a curve, with a planned mean GPA of 2.7 as recommended by EECS department policy. If we are impressed with the performance of the class as a whole we may push the average GPA as high as 2.9, and if we are disappointed we may push the average as low as 2.5. Your total class grade will be the summation of all of the assignments from each of the following categories.

Labs (6%): Labs are intended to give you an opportunity to apply concepts covered in class before you encounter them in projects or on exams. To ensure that everybody fully understands the contents of a lab we require that students get checked off for a grade. Because the goal of lab is not to assess your competence, but rather to guarantee your understanding, you may try to get checked off multiple times without any penalty. Checkoffs are to be completed in pairs, and both students in a pair are expected to have a full and complete understanding of the material – you should **NOT** be splitting up the exercises between you and your partner. We do not strictly forbid solo checkoffs, but your GSI may or may not provide them, at his discretion.

All labs are designed to be completed within the 2 hours allotted to them, but sometimes a student will require more time. If that happens you may get checked off in the following lab section without penalty. You will accrue a 1 point penalty (labs are marked out of 4 points) every lab section after that, until you reach 0 points for the lab. Take advantage of this late policy at your own risk – 61c moves at a ludicrous pace during full length semesters and we're covering the same amount of material in roughly half the time. If you fall behind you may well find it impossible to catch up.

Last, labs are to be checked off in your lab section, not office hours, discussion section, in another lab section, or anywhere else. If you attempt to get checked off at any time other than your assigned section you will be entirely at the mercy of the GSI who would be checking you off.

Homeworks (5%): Homeworks, similar to labs, are intended as an opportunity for you to get some practice with the material before being assessed on it. Unfortunately we don't have the time to walk through every student's solution to every homework, so we'll mark homeworks based on effort/completion instead of correctness and release solutions after the due date. That means that even if you get every question on the homework wrong you can still get full credit as long as it's clear that you put some time into completing the it (e.g. written answers contain reasonable justifications, code is at least partially functinoal and appears to be on the right track).

Projects (24%): There will be three projects throughout the class, each carrying the same weight. They will be largely autograded, but that autograder will not be made available to you until after the project is due, if at all. We will usually provide a minimal test suite to help you check your sanity, but that is all. That means that it is absolutely crucial that your test your code, and test your code thoroughly. Autograders don't much care if you "almost had it", and neither will we. You have been warned.

Midterm (30%): The midterm exam will test you on anything and everything contained within the sum of all human knowledge, but we'll place an emphasis on the material covered in the course prior to the exam. Each student is allowed to bring one double-sided 8.5"x11" sheet of handwritten notes to the exam (note that this means no Möbius strips). A copy of the MIPS green sheet will be provided. The midterm will take place over three hours on Monday 07/21. The exact time and location will be determined later and posted on piazza and the course webpage.

Final (30%): The final, like the midterm, will test you on anything and everything with a focus on the material we've covered in the course. Each student is allowed to bring two double-sided 8.5"x11" sheets of handwritten notes to the exam. A copy of the MIPS green sheet will be again be provided. It will take place over three hours on 08/15. The exact time and location will be determined later and posted on piazza and the course webpage.

In order to guarantee that everyone has a shot at a good grade up until the final we'll be implementing a "midterm clobber" policy where a portion of the final will cover only material from before the midterm. If you do better on this than you did on the final we'll replace your midterm score with your score on the midterm portion of the final, renormalized according to the relative difficulties of the exams. More precisely, if we let μ_1, σ_1 be the mean and standard deviation of the midterm, μ_2, σ_2 be the mean and standard deviation of the midterm portion of the final, M be your score on the midterm, and F be your score on the midterm portion of the final then we define your clobbered midterm score M' as follows

$$M' := \max \left[M, \frac{F - \mu_2}{\sigma_2} \cdot \sigma_1 + \mu_1 \right]$$

EPA (5%): EPA (Effort, Participation, Altruism) is not actually directly included in the course grade. Rather, it allows course staff to bump the grades of borderline students that have put in a lot of effort, been unusually active in class, or been particularly helpful to other students. If you want to make sure you have an EPA boost available to you then you should help other students in your lab section after you finish (we do notice that), attend a lot of office hours, answer a lot of questions on piazza, and participate in discussion.

4.1 Slip Days and Late Penalties

Homeworks and projects which are late acquire a 33% penalty for each day they are late. This penalty is taken from the assignment value, not your score on it, so if you were one day late and scored 10/15 on a project then you'd receive 5 points after the late penalty. We do understand that sometimes life gets in the way and you can't complete an assignment in time. To deal with these circumstances we provide every student with three slip days at the start of the semester, each of which negates one day's worth of late penalty. Please don't request an extension on an assignment from the staff if you still have slip days – you already received the extension in the form of the slip days. Slip days will automatically be applied to late submissions unless you send an email to your reader within three days of the assignment due date.

5 Piazza

Piazza is an indispensible tool for the course staff to encourage learning, but if students use it incorrectly its usefulness rapidly detiororates. Here are a couple of the worst behaviors we've seen, the effects they have on the forum, and how you're expected to avoid them:

- Redundant Questions – If you ask a question that's already been asked then you're drowning out other useful conversation and making it harder for people to find the posts they're looking for, in addition to wasting the time of the instruction staff. Please be sure to spend at least five minutes searching for similar questions before asking your own (and don't forget about other resources, like google).

- Half-baked Posts – From time to time a post will crop up, and it's clear that absolutely no thought went into its construction. Maybe the English resembles poorly formed text-speak, or maybe the question asks about something which was clarified in the first line of a project specification. In any case, this question isn't going to help anybody else, and it's taking up just as much space as any other post. All you have to do to avoid this one is to spend a little bit of time forming a precise question before asking it. Another, related issue is when a student fails to properly title his or her question. Something along the lines of "hw1 q2" is unacceptably vague; a better version would be "hw1 q2: Clarification on how to handle escape characters".

- Using the Student Answer Section to Ask Questions – I've seen a number of students who will chime in with things like "me too!" inside the student answer section. Doing so will mark the question as resolved, and will significantly reduce the likelihood of the course staff seeing the question. Just don't do this, ever.

- Asking for Debugging Help – If you need help debugging a piece of code, then you should go to office hours, where we can help you to learn how to debug for yourself. Helping over piazza almost always devolves into us pointing to a line of code and saying "there's your problem", which is not at all instructive. As a result we will not under any circumstances answer these sorts of questions, other than to tell you that you're wasting your time.

6 Academic Integrity

Unless explicitly stated otherwise, all the work you turn in in this class is to be yours and yours alone. You are encouraged to discuss the assignments with other students, but we expect that what you hand in is yours. It is **NOT** acceptable to copy solutions from other students or to copy (or start your) solutions from the Web. We have tools and methods, developed over many years, for detectings this. You **WILL** be caught, and the penalties **WILL** be severe. At minimum:

1. NEGATIVE 100% for BOTH the cheater and the enabler on the assignment.
2. A letter to the university record documenting the incident.
3. Consideration for an automatic F in the course, at the discretion of the course staff. This is by far the most likely option if we tell you that we found you cheating before you tell us that you cheated.

Do not think that because you aren't accused early that you've gotten away with it. We may leave all of the cheating cases to the end, or we may prosecute a random subset before the end of the semester, or we may prosecute all of them as we find them. The process will be completely opaque.

There has been some confusion in the past about how much discussion is allowed before you've started cheating. As a rule of thumb, if what you're discussing is much more detailed than a basic flowchart of your approach then you're overstepping your bounds. You should definitely never share implementation details with any other students (unless they're your partner for a partner assignment). If you're unsure of whether or not something is over the line, then *ask*.

Last, a word of warning. You are responsible for the security of your code. If you leave your computer logged in with your code visible, or keep your code on a public repository, or on pastebin, or anywhere where an unscrupulous student could find and steal it without too much difficulty then you are liable as an enabler of cheating.