# CS64: Computer Organization and Design Logic
# Winter 2016

**Instructor:** Kyle Dewey (kyledewey@cs.ucsb.edu)
**Teaching Assistants:**

- Ravi Kumar (dropmailtoravi@gmail.com)
- Metehan Ozten (mozten@cs.ucsb.edu)

**Website:** http://cs.ucsb.edu/~kyledewey/cs64
**Lecture:** Tuesday / Thursday 3:30 PM - 4:45 PM in Phelps 3526
**Lab:** Wednesday 4:00 PM - 4:50 PM; 5:00 PM - 5:50 PM in Phelps 3525

**Textbook:**

- Computer Organization and Design: The Hardware/Software Interface, 5th edition (by David A. Patterson and John L. Hennessy)

**Course Description:**
(In my own words) An introduction to computer hardware, starting from the programmer's low-level interface and ending with basic circuitry.

(From the course catalog) Assembly language programming and advanced computer organization; Digital logic design topics including gates, combinational circuits, flip-flops, and the design and analysis of sequential circuits.

**Prerequisites:**

- CS16, with a grade of 'C' or better
- Math 3C or 4A with a grade of 'C' or better

**Graded Components:**

- 5% - Lab Attendance, 10 sessions @ 0.5% per session
- 30% - Lab Assignments, first 7 assignments @ 3% apiece; last 2 assignments @ 4.5% apiece
- 12.5% - Exam #1, closed book, one hand-written page of notes (double sided), along with MIPS reference card
- 12.5% - Exam #2, closed book, one hand-written page of notes (double sided), along with MIPS reference card
- 40% - Final Exam - closed book, two hand-written pages of notes (each double sided), along with MIPS reference card

**Course Outline:** (subject to change)

| Week | Day | Topic | Reading |
|---|---|---|---|
| 1 | 1/5 | Introduction: bases, binary, data representation, bitwise operations | Ch. 2.4, 2.6 |
| 1 | 1/7 | Bitwise operations, addition, multiplication | Ch. 2.6, 3.2, 3.3 |
| 2 | 1/12 | Assembly: MIPS introduction with `spim` | Ch. 2.2, 2.3, 2.6 |
| 2 | 1/14 | Assembly: arithmetic, branches | Ch. 2.2, 2.3, 2.6 |
| 3 | 1/19 | Assembly: MIPS memory | Ch. 2.8 |
| 3 | 1/21 | Assembly: MIPS memory<br>Data representation: floating-point, big/little endian | Ch. 2.8 |
| 4 | 1/26 | Assembly: procedures, exam review | Ch. 2.8, A.6 |
| 4 | 1/28 | **Exam 1** | |
| 5 | 2/2 | Assembly: calling convention rules | Ch. 2.8, A.6 |
| 5 | 2/4 | Assembly: calling convention rules | Ch. 2.8, A.6 |
| 6 | 2/9 | Assembly: calling convention rules | Ch. 2.8, A.6 |
| 6 | 2/11 | Digital design (DigDesign) introduction | B.1 - B.3 |
| 7 | 2/16 | DigDesign: boolean algebra | B.1 - B.3 |
| 7 | 2/18 | DigDesign: combinatorial logic | B.3, B.5 - B.7 |
| 8 | 2/23 | DigDesign: MUXs, ALUs, exam review | B.3, B.5 - B.7 |
| 8 | 2/25 | **Exam 2** | |
| 9 | 3/1 | DigDesign: clocks, latches, flip-flops, memory | B.8, B.9 |
| 9 | 3/3 | DigDesign: Drawing FSMs | B.10 |
| 10 | 3/8 | DigDesign: Drawing FSMs | B.10 |
| 10 | 3/10 | Final exam review | |
| 11 | 3/17 | **Final exam** | |

**Final Grade Assignment:**

The table below describes how final letter grades are assigned in the course. The left column shows the minimal score necessary to receive the grade in the right column. The highest letter grade possible given the score is chosen; e.g., if you receive an 88.2, you'd receive a 'B+' for the course, which corresponds to being >= 86.5.

| If your score is >=... | ...you will receive... |
|---|---|
| 96.5 | A+ |
| 92.5 | A |
| 89.5 | A- |
| 86.5 | B+ |
| 82.5 | B |
| 79.5 | B- |
| 76.5 | C+ |
| 72.5 | C |
| 69.5 | C- |
| 66.5 | D+ |
| 62.5 | D |
| 59.5 | D- |
| 0 | F |

The above cutoffs are strictly enforced; e.g., if you had a 79.4999999, this would be considered a 'C+' as opposed to a 'B-', as the cutoff for a 'B-' is 79.5. The reasons for this are twofold:

1. Ultimately, any grading system imparts a cutoff somewhere. Any relaxation introduces inconsistency, which is unfair.
2. These cutoffs are slightly lower than the typical cutoffs; e.g., the cutoff for an 'A+' is typically 97, not 96.5. As such, the cutoffs used for this class effectively have built-in rounding.

There is **one exception to the above rules**: if your exam average is below 60% and your overall average is significantly below the class average, you will receive an 'F' in the class. This is to protect against students who lean on their partners too much. This very rarely comes into play.

**Grading Errors:**
If you believe that there was a grading error made for any assignment, let us know **within one week** of receiving your grade. We can then double-check the result. If the student lost points due to an error in grading, then those points will be returned. This can only possibly increase the number of points awarded for an assignment; if we discover a grading error that works in the student's favor (i.e., the student should have been awarded fewer points), then this will be ignored.

**Regrading:**
All of the labs as well as exams 1 and 2 allow for **regrading**, wherein a student may get back up to half of the points lost on assignment. This is applicable only if an assignment was turned in and *some* attempt was made (e.g., turning in empty files or a blank exam is not acceptable). If a student is interested in regrading, then a request for a regrade must be received **within one week** of receiving the grade for the assignment. Regrade requests must be submitted **via email to the instructor.** Requests for regrading **must** contain the following information:

**1.** The question(s) you want regraded. For example, "Task 2 of assignment 3".
**2.** In your own words, what you did wrong **for each question**. For example, "I forgot to increment the loop counter".
**3.** (Optional) Why you got the question incorrect. This information is used both to improve the course (as when students consistently had the same problems), and to help identify if some extra help would be beneficial. Some possible answers:
    - Careless error
    - I ran out of time
    - I panicked
    - I didn't understand the question
    - I didn't understand some concept the question assumed I understood
    - I didn't study enough
    - I don't have a strong grasp on the material
**4.** Your revised answer **for each question**, which should be correct. For example, assembly code with the updated loop counter.

You might not be able to provide all the above information. For example, if you're struggling to understand the material or you're otherwise confused about the course content, you might not be able to provide a correct answer, or even start to provide a correct answer. If this is the case, try your best to provide a correct answer, and say that you want some extra help.

In all cases, for each question you want regraded, you'll be given a similar problem which you'll need to solve. This problem will be given either via email or in-person. We will attempt to honor all explicit requests to meet, and we may recommend that a meeting occur even if one isn't requested. This similar problem will then be graded, and based on the grade for this similar problem, you'll be assigned up to half of the points you lost.

While requests for regrading must be submitted within one week, it may take significantly longer on our end for the regrade to happen.

**Background Behind Regrading:**

Regrading exists to encourage students to revisit assignments that went poorly, in order to ensure that material is properly learned. This is good advice in general, but it's especially important in this class with respect to the final exam. The final exam is cumulative, and nearly every single point of the final exam is derived from an assignment which is eligible for regrading. Additionally, the final exam is **not** eligible for regrading. It is because of these two reasons that the final exam is worth so much - it's designed to test only material you've already seen, and you have had at least two opportunities to be accessed favorably on this material (i.e., the first time you performed the assignment, and once for regrading).

It may seem odd that you must solve the original problem in addition to a problem given, especially considering that it's from the new problem where your updated score is derived from. We follow this process ultimately because if you understand the topic at hand, then redoing the original problem should not be difficult. If it is difficult, then you almost assuredly won't be able to successfully complete a similar problem. Not only does this improve your chances of success on a similar problem, it will make it immediately clear whether or not additional help is needed from us.

**On Using Partners:**

All the labs allow for the use of a partner. Under this setup, students are allowed to form **groups of two**, and may discuss assignment solution details within the group. Each member of the group is still accountable for his or her own work, though free communication is permitted within the group. This is more privileged than a group member communicating with a non-group-member (who cannot exchange assignment solution details), but less privileged than a team (who submit assignments as a single unit, as opposed to individuals). The intention with this setup is that if someone is stuck, they can ask their partner for help without needing to contact the TA or the instructor. Students may change groups between assignments, though not within an assignment, which avoids situations where information could be exchanged between more than two students on a single assignment. Forming groups is optional; students may work alone if they want.

**On Collaboration:**

In actual software development, very little work is done as an individual effort. Most modern software systems are simply too big to be undertaken in this style, so collaboration is key. This is one of the reasons why we allow partners for some of the labs - this represents the common case.

That said, collaboration is **not** simply taking the credit for someone else's work. When this happens, **everyone** loses. The obvious loser is whoever did all the work. The less immediate loser is the one who took half of the credit for the work, because that person did not learn whatever the assignment was trying to teach. This can (and I have seen it happen during my own undergrad!) snowball long-term. Eventually the student who took the credit without working is truly completely lost, often without realizing it, and is held up entirely by the work of other students. This is a worst-case scenario, but in my experience it is far too common.

The point is this: collaboration means to work together, not to blindly take someone else's work or to give your own work away.

**Attendance policy:**

Attendance will not be taken in lecture, and there will be no graded assignments given during lecture. However, any material covered in lecture is ultimately **your** responsibility, regardless of whether or not the lecture was attended. Attendance will be taken for lab, and being present at each lab counts towards a total of 0.5% of your grade.

**Due Dates / Late Policy:**

For items turned in late, each person has 24 hours worth of "grace" time in total. For example, if someone were to submit the first lab 4 hours late and the second lab 6 hours late, then a total of 10 "grace" hours have been used. Both submissions would be accepted without incident, and there would be 14 "grace" hours remaining. Except in extenuating circumstances, submissions for students who have gone beyond their grace time will **not** be accepted.

A little background on this policy - the grace time is intended to be used as a sort of last-minute "oops" relating to a poor time estimate of (what should be) final touches. This policy tries to reduce the number of submissions hastily done just to meet a deadline, and to prevent issues of submissions that missed the deadline by a relatively small amount of time. It is not intended to be used as a way to extend the deadline for one assignment for a day, although it certainly can be used that way. Be forewarned: once it's gone it's gone, so use it wisely!

**Extenuating Circumstances:**

"Extenuating circumstances", for the purpose of this class, is defined as anything beyond our immediate control. In these cases, at my discretion I can grant an extension. To be absolutely clear, there is **no** guarantee that I will do so, and I am not obligated to grant them. For the things we can predict (e.g., trips), I expect to be contacted at least a week in advance. For the things we cannot predict (e.g., illness), I need official documentation explaining the situation (e.g., a doctor's note).

**Communication Policy:**

I have two office hours per week, though I may increase this to 3-4 if questions abound. I'm also available by appointment.

With email, assume that I will take at least 24 hours to respond. Typically my response time is much, much faster than this, but I do occasionally take this long. Historically, this has only been an issue the last hours before a project deadline, and only for students who started far too late. The point being: start early!

**Academic Honesty:**

In as few words as possible, cheating and plagiarism will **not** be tolerated. I understand that the temptation may be high ("it's just this one assignment" or "I just need this class"), but this is **no** excuse. At the very least, this is unfair to all the students who did not resort to such unethical means, who instead took the time and struggled through. I will be following UCSB's Academic Conduct policy on this (from http://www.sa.ucsb.edu/Regulations/student_conduct.aspx, under "General Standards of Conduct"), quoted below for convenience:

> It is expected that students attending the University of California understand and subscribe to the ideal of academic integrity, and are willing to bear individual responsibility for their work. Any work (written or otherwise) submitted to fulfill an academic requirement must represent a student's original work. Any act of academic dishonesty, such as cheating or plagiarism, will subject a person to University disciplinary action. Cheating includes, but is not limited to, looking at another student's examination, referring to unauthorized notes during an exam, providing answers, having another person take an exam for you, etc. Representing the words, ideas, or concepts of another person without appropriate attribution is plagiarism. Whenever another person's written work is utilized, whether it be a single phrase or longer, quotation marks must be used and sources cited. Paraphrasing another's work, i.e., borrowing the ideas or concepts and putting them into one's "own" words, must also be acknowledged. Although a person's state of mind and intention will be considered in determining the University response to an act of academic dishonesty, this in no way lessens the responsibility of the student.

Note that collaboration with a non-lab partner also constitutes cheating. Any incident of cheating **will** be reported. While this may sound steep, real-life cases of cheating (i.e. taking someone else's code without permission) have led to job termination and lawsuits among other things, so this is not unrealistic.