CMPE 012 Syllabus - Summer 2018

Computer Systems and Assembly Language

Overview

This class exposes students to an overview of how computers work. We will discuss the building blocks of computer hardware, how data is stored in memory, and the MIPS instruction set.

Instructor

Rebecca J. Rashkin   rrashkin+ce12@ucsc.edu

Personal website   www.rebeccajr.com

Office Hours   E2-301, Wednesday 12:00 PM - 1:00 PM or by appointment

Salutations! I am thrilled to be teaching CMPE 012 this summer as a graduate student instructor. Please address me as *Rebecca, Ms. Rashkin, Miss Rashkin* or *Miss Rebecca*.

Teaching Assistant

Mengjie Li   mli55+ce12@ucsc.edu

General

Dates: June 26, 2018 - August 30, 2018

Lecture: Physical Sciences Rm. 110, Tue/Thu 1 - 2:45 PM

Labs: Social Sciences 1 Rm. 135, Wed 2 - 4 PM, Thu 4 - 6 PM

Open lab hours: Mon - Thu 9 AM - 8 PM, Fri 9 AM - 5 PM, Sat/Sun closed

Websites: www.canvas.ucsc.edu (must have a UCSC login ID)

     https://piazza.com/ucsc/summer2018/cmpe012/home

Textbooks

Required

Yale N. Patt and Sanjay J. Patel, Introduction to Computing Systems: From Bits and Gates to C and Beyond (Reader), McGraw-Hill Education, First Edition, ISBN 978-1307117530, 2017. (available at the Bay Tree Bookstore)

Charles W. Kann, Introduction to MIPS Assembly Language Programming, Gettysburg College Open Educational Resources, 2015. http://cupola.gettysburg.edu/oer/2/ (available online for free)

Optional

David A. Patterson and John L. Hennessy, Computer Organization and Design MIPS Edition: The Hardware/Software Interface, Fifth Edition, ISBN 978-0124077263, 2013. (available at the Bay Tree Bookstore)

Larry Gonick, The Cartoon Guide to Computer Science, Harper & Row Publishers, Inc., ISBN 0-06-460417-9, 1983. Can be purchased for $2 on amazon: https://www.amazon.com/gp/product/0064604179/ref=oh_aui_search_detailpage?ie=UTF8&psc=1

Prerequisites

Prior programming experience is highly recommended (e.g. CMPE 13, CMPE 5J/P) to succeed in this class. Specifically, it is imperative to understand functions, conditional statements, and loops. If you need a refresher on coding, check out:

http://codingbat.com/

Key Dates

Class

Holiday	Wed Jul 4, 2018 (no lab section)
Midterm	Tue Jul 24, 2018 (subject to change)
Final	Thu Aug 30, 2018

Administrative

Add	Thu Jul 5, 2018
Drop	Mon Jul 9, 2018 (tuition refund)
Change grade option	Fri Jul 20, 2018
Withdraw deadline	Fri Jul 27, 2018 (no tuition refund)

Attendance

Lab

Attending 1 lab section per week is mandatory. You may attend either lab section. However, if the computer lab is full, you may be asked to leave if you are not enrolled in that section. You are permitted to miss up to 4 lab sections with no penalty. If you miss more than 4 lab sections, you will not pass the class.

July 4th falls on one of the lab section days. For this reason, attendance will not be required in week 2. This will not count towards one of your 4 excused absences.

Signing in or being signed in by another student counts as academic dishonesty and disciplinary action will be taken.

Lecture

Attending lecture is not mandatory, however it is highly suggested because there will be in-class quizzes nearly every week.

Evaluation

40%   Labs (must average greater than 50% to pass)
20%   Quizzes (0 - 2 per week, 10 will be used in your grade calculation*)

20%   Midterm exam
20%   Final exam

Labs

Labs will be weighted as such:

2%   Lab 0
5%   Lab 1
10%   Lab 2
18%   Lab 3
15%   Lab 4
25%   Lab 5
25%   Lab 6

*Quizzes

There will be short quizzes administered 0 - 2 times per week. Your quiz average will be calculated by the best 5 quiz scores before the midterm and the best 5 quiz scores after the midterm. Quizzes may be administered at any point during the lecture period. Quizzes will be hand written, so please bring a writing utensil and paper to class. There are no makeups for quizzes.

Passing Requirements

You must earn at least a **50%** average on the **exams** to pass the class. Likewise, your average **lab scores** must be no less than **50%** to pass the class. You must also be present for lab section at least 5 weeks. There is no minimum grade requirement for the quizzes.

It is not possible to pass this course with a 0 on any lab assignment. You can receive a 0 if the lab assignment was not submitted, files are missing, files are in the wrong format (including having the wrong file extension or naming convention), you put in little to no effort, or are caught cheating.

Regrades

Re-grading of lab assignments will only be done if we have made a clerical error (i.e., we added points wrong) or we somehow missed your work. Note that **this does not include you forgetting to submit** or if you made a mistake in the submission process.

Re-grade requests must be submitted as a private message to all instructors on Piazza within **one week** of when grades are released.

Note

You will receive the **same grade** for CMPE 12 and CMPE 12L. Assume that CMPE 12 and 12L are one 7-unit course. You must take both the lab and lecture to complete this course.

Late Submission

Students will have 72 hours total of "grace period" that can be used on lab assignments to allow for unexpected life events that interfere with turning in assignments by the deadline. The grace period hours are consumed in 12 hour (half day) increments that is kept track on Canvas.

We do NOT need to know why you are using your grace period. If there is some extenuating circumstance that is not covered by the grace period, we need to be informed of this ASAP (not after the fact).

Piazza

Piazza is the official class online forum for delivering announcements and connecting with other students. It is your responsibility to check the forum regularly for updates on assignments and logistics.

The website for our class is: https://piazza.com/ucsc/summer2018/cmpe012/home

Guidelines
1. If you have a question, help yourself first. Look at the notes, book, other Piazza posts using the search feature.
2. If you can't figure out the answer on your own, post a question with a useful subject and describe your problem.
3. Piazza is a professional forum so be polite and respectful. Your post is never anonymous to instructors. Inappropriate or rude posts will be removed.

Software

The following software is used for the lab assignments.

Git
https://git-scm.com

Multimedia Logic
http://www.softronix.com/logic.html

MARS: MIPS Assembler and Runtime Simulator
https://courses.missouristate.edu/KenVollmar/MARS/

Incompletes

Students requesting an incomplete must be in good standing (have a passing grade up until that point). An incomplete will only be granted as a result of a medical or family emergency. You must request an incomplete before the last day of instruction (August 28, 2018).

Note
If you ask for an incomplete after not turning in labs or homework assignments, you are not in good standing.

Disability Accommodations

Please request academic accommodations due to a disability through:

Disabled Resource Center
146 Hahn Student Services
(831) 459-2089 (voice)
(831) 459-4806 (TDD/TTY)

These accommodations must be discussed with the instructor in the first 2 weeks of the session. In addition, please confirm with SOE teaching services at least one week before each exam to secure testing accommodations.

Academic Integrity

Academic honesty is a requirement for the course. All assignments must be your own independent work.

What is cheating? It is presenting work that is not yours as your own. You can - and are encouraged to - discuss and strategize with your colleagues on the material and labs, but your work should be your own. Copying is NEVER acceptable.

On the labs, cheating is sharing code unless explicitly told that it is permitted. If a student is caught cheating in either the class or the lab, this will result in an immediate failure in the class and the lab. It will be reported to your college and your department. DO NOT CHEAT; it is not worth it.

Collaboration vs Plagiarism

Collaboration is a key aspect of education, and of professional engineering. As such, we want to encourage you to help each other understand the labs and the material. However, this is NOT license to copy others' work. Credit for collaboration should be explicitly noted in each assignment's README. Failure to give credit on collaboration is considered a form of cheating and will be dealt with accordingly.

Unfortunately, in most quarters, we find evidence of plagiarism in a significant fraction of the CMPE 12 student body. In many instances, students are (or claim to be) unaware that what they are doing is plagiarism. To clarify, here are examples of acceptable collaboration:

- Discussing the labs, or lab requirements with peers.
- Planning, diagramming, or writing pseudocode for a lab with peers.
- Helping a peer understand an error or assist in debugging.

In the above cases, this collaboration must be described in the README that you submit.

Here are examples of unacceptable collaboration (ie, cheating):

- Reading someone else's code for inspiration before you start your own code.
- Reading someone else's code as you write your own.
- Checking with a friend every few minutes to be sure you've written similar things.
- Using more than a line or two from Stack Overflow or a similar online resource.
- Using code from a friend or online resource, but altering it slightly.
- Reading someone else's code to find a solution to a flaw in your own code.
- Typing out code that friend dictates to you as you type.

A good practice for collaborating while avoiding cheating is to use a whiteboard or a piece of paper to communicate ideas, and look at each other's monitors as little as possible.

Title IX

The university cherishes the free and open exchange of ideas and enlargement of knowledge. Maintaining this freedom and openness requires objectivity, mutual trust, and confidence; it requires the absence of coercion, intimidation, or exploitation. The principal responsibility for maintaining these conditions must rest upon those members of the university community who exercise most authority and leadership: faculty, managers, and supervisors.

The university has therefore instituted a number of measures designed to protect its community from sex discrimination, sexual harassment, sexual violence, and other related prohibited conduct. Information about the Title IX Office, the online reporting link, applicable campus resources, reporting responsibilities, the UC Policy on Sexual Violence and Sexual Harassment and the UC Santa Cruz Procedures for Reporting and Responding to Reports of Sexual Violence and Sexual Harassment can be found at:

http://titleix.ucsc.edu

The Title IX/Sexual Harassment Office is located at 105 Kerr Hall. In addition to the online reporting option, you can contact the Title IX Office by calling 831-459-2462.

Topics

- History and Introduction
- Numbering Systems
- Binary Numbers
- Data Representation
- Transistors
- Digital Logic: Combinational, Sequential
- ALU Computations
- Computing Overview
- MIPS ISA
- MIPS address space
- Intro MIPS Programming
- Subroutines, the stack
- MIPS Instruction Decoding
- MIPS Architecture / data path
- IO and Exceptions / Traps / Interrupts
- Number systems revisited
- Fractional Binary
- Floating Point

Reading List		COMPUTER ORG 5TH ED		INTRODUCTION TO COMPUTING SYSTEMS	
WEEK-DAY	**TOPIC**	**SECTION**	**PG**	**SECTION**	**PG**
1-1	history, layers of abstraction			1	1
1-2	binary representation		2.4		
1-2	binary-decimal conversion			2.4	27
2-1	transistors			3.1	51
2-1	inverter / not	B.2	B-4	3.2.1	53

2-1, 2-2	or, nor, and, nand	B.2	B-4	3.2.2-3.2.5	54
2-2	boolean algebra, identities, DeMorgans Law	B.2	B-4	3.2.4	
3-1	sum of products, product of sums, PLAs, decoder, mux, full adder, logical completeness	B.3	B-9	3.3	59
3-2	bitwise (ignore the bits about C)	2.6	87	12.3.5	
3-2	shifts	2.6	87		
3-2	data representation, unsigned, signed, 2's complement	2.4	73	2.1-2.4	21
4-1	ascii, unicode	2.9	106	2.7.3	
4-1	floating point representation	3.5	196	2.7.2	37
4-2	clock	B.7	B-48		
4-2	memory elements: latches, flip flops, registers	B.8	B-50		

Labs

There will be 6 labs on the following topics:

Lab 1:   Basic programming

Labs 2-3: Logic design using Multimedia Logic

Labs 4-6: Assembly using MARS