

CS 4400: Computer Systems

Administrative Details and Syllabus

Fall 2012

Course Information

Description. The objective of this course is to help students bridge the gap between high-level programming and actual computer systems: processors, the memory hierarchy, operating systems, compilers, linkers, assemblers, networks, and more.

Our basic goal is to understand how a computer works, so that as programmers we can make it work efficiently. Thus, this course is an introduction to computer systems from a programmer's point of view.

The official prerequisite for this course is CS 3810 (Computer Organization). *It is strongly recommended that students complete CS 3505 (Software Practice II) before taking this class.*

Meetings. Mondays and Wednesdays 1:25-2:45p in 101 WEB.

Instructor. Matthew Flatt. *Email:* mflatt@cs.utah.edu. *Office:* 3458 MEB

Course Materials

Textbook. The required course text is *Computer Systems: A Programmer's Perspective* by Randal E. Bryant and David R. O'Hallaron, 2nd edition, Prentice-Hall, 2010 (ISBN: 0-13-610804-0). An e-version is available at <http://www.coursesmart.com/0132130661/?a=1773944>.

A highly recommended book is *The C Programming Language* by Brian W. Kernighan and Dennis M. Ritchie, 2nd edition, Prentice-Hall, 1988 (ISBN: 978-0131103627).

Website. The class website is <http://www.eng.utah.edu/~cs4400>. It will be updated throughout the semester with the lecture schedule, assignments, links to course handouts, and more.

Lecture notes. The instructor will often make use of slides, sample code, and other documents during lecture. These documents will be posted on the class website following the lecture; however, such posted documents may not represent completely the material covered in class. Students who must miss class are strongly encouraged to check with a classmate.

Student Evaluation

Lab Work. Because this is a 4-unit course, there is a significant amount of lab work in the form of programming assignments.

The lab work makes heavy use of C, Unix, and the Intel IA32 (x86) architecture. Students not currently fluent in any of these three topics should not panic, as this course will cover them in more detail throughout the semester. However, there is an assumption that students have some familiarity with C or C++. Students should be prepared to learn some of the C programming language on their own, for which the Kernighan and Ritchie reference text will be very useful.

To behave properly, all lab work must use an x86 processor (either AMD or Intel) that has a Unix OS. *Code must be in ANSI standard C—nothing else will work.* Unless explicitly noted otherwise, grading of assignments will be done using CADE Lab 1 machines. Students who choose to develop their code on any other machine are strongly encouraged to run their programs on a CADE Lab 1 machine before turning it in. *There will be no credit for programs that do not compile and run on a CADE Lab 1 machine,* even if they run somewhere else.

CADE Lab 1 machines are numbered 1-48 and have names `lab1-x.eng.utah.edu`, where `x` is the machine number. For more information on the CADE lab and how to remotely log into these machines, see <http://www.cade.utah.edu>.

Programming assignments are due by 11:59p on the due date via electronic submission. Late programming assignments are accepted according to the following rules.

- Assignments are not accepted more than 3 days after the due date.
- Assignments submitted any time X days after the due date (midnight to 11:59p) are penalized $X * 10\%$ of the assignment grade.

Exams. Two midterm exams will take place in class September 26 and November 7. The final exam is cumulative and will take place 1-3p December 14 in 101 WEB.

Problem Sets. At each class meeting, a number of problems from the textbook will be assigned. The solutions are submitted in weekly batches. That is, solutions to the problems assigned on the Monday and Wednesday of a week are due by 1:25p on the following Monday via **handin**. (If the following Monday is a holiday, solutions are due by 1:25p on the next class meeting day.) Unless explicitly noted otherwise, all submissions must be in PDF format. The solutions must be clear and easy to read, but are not required to be typed. *Problem sets are not accepted late.* Solutions to the problems will be discussed in class or emailed to the class mailing list. The first problem set will be graded, and about 1/3 of the other problem sets will be randomly selected for grading.

Course Grade. For students with an average, un-curved score on exams (Midterm 1, Midterm 2, Final Exam) of 65% or lower, the final course grade will be this average. Otherwise, the final course grade will be based on a number of evenly-weighted lab-work assignments (50% total), two evenly-weighted midterm examinations (25% total), a final examination (15%), and weekly problem sets (10%).

Scale for assigning letter grades:

100-93	A	89-87	B+	79-77	C+	69-67	D+	59-0	E
92-90	A-	86-83	B	76-73	C	66-63	D		
		82-80	B-	72-70	C-	62-60	D-		

Students who wish to appeal a grade on an assignment or an exam, must do so within one week of receiving the grade.

Getting Help

Instructor office hours. See <http://www.eng.utah.edu/~cs4400/help>.

Teaching assistants and consulting hours. See the class website for the consulting schedule of the course TAs. During consultation, use the TA Queue Signup (also on the class website) to alert the TA on duty that you have a question.

Communication. For questions outside of class and consulting hours, students are encouraged to use email.

The course staff will use the class mailing list (cs4400@eng.utah.edu) to send urgent messages to everyone in the class, such as corrections to assignments or changes in due dates. Students are not able to send mail to the class list. Students *must* subscribe to the class mailing list ASAP by visiting <https://sympa.eng.utah.edu/sympa/info/cs4400>.

Students who would like to ask a question of the course staff should use the staff mailing list (teach-cs4400@eng.utah.edu). The instructor or a teaching assistant will respond to each question directly.

Course Guidelines

Behavior in the classroom. All students are expected to maintain professional behavior in the classroom setting, according to the University of Utah Student Code, which is posted at <http://www.regulations.utah.edu/academics/6-400.html>. Students should read the Code carefully and know that they are responsible for the content. According to Faculty Rules and Regulations, it is the faculty responsibility to enforce responsible classroom behaviors, beginning with verbal warnings and progressing to dismissal from class and a failing grade. Students have the right to appeal such action to the Student Behavior Committee.

Working Together. Students are encouraged to discuss lab-work assignments and problem sets with fellow classmates, but each student is responsible for writing her own answer. *Cheating is:* sharing code or other electronic files either by copying, retyping, looking at, or supplying a copy of a file. *Cheating is not:* discussing concepts, answering questions about concepts or clarifying ambiguities, helping someone understand how to use the computer systems or basic tools (compiler, debugger, etc.), or helping with high-level design issues or general debugging.

Except when explicitly designated otherwise, each assignment is to be done individually. For all assignments, the solution submitted by each student will be checked against the solutions of other students (from this year's class, as well as, previous years) for anomalies. If an anomaly is found that cannot be explained satisfactorily, the students involved will fail the course.

Of course, there must be no collaboration during examinations. Please see the University of Utah Student Code for a detailed description of the university policy on cheating.

Any student found cheating will fail the entire course.

College of Engineering guidelines. For information on withdrawing from courses, appealing grades, and more, see <http://www.coe.utah.edu/SemesterGuidelines.pdf>.

Students with disabilities. The University of Utah seeks to provide equal access to its programs, services and activities for people with disabilities. If you will need accommodations in the class, reasonable prior notice needs to be given to the Center for Disability Services, 162 Union Building, 581-5020 (V/TDD). CDS will work with you and the instructor to make arrangements for accommodations.

Syllabus

The following are the key topics planned for study, the approximate number of lectures devoted to each, and the corresponding chapters in the course text.

Getting Started (1 lecture) Chapter 1

- Administrative details
- Overview of computer systems

Representing Information (8 lectures) Chapters 2-3

- Bits and bytes (information storage)
- Integers (representation, arithmetic)
- Floating point (representation)
- x86 machine-level code (accessing information, operations)
- Control flow (jumps, branches)
- Procedures (run-time stack, recursion)
- Data (arrays, pointers, structures, alignment)

Optimizing Code (2 lectures) Chapter 5

- Optimizing compilers
- Loops
- Branch prediction
- Memory performance

The Memory Hierarchy (2 lectures) Chapter 6

- Different kinds of memory
- The principle of locality
- Cache memory

Running Programs on a System (8 lectures) Chapters 7-10

- Linking
- Exceptions
- Processes
- Signals
- Virtual memory and address translation
- Dynamic memory allocation

Interaction Among Programs (3 lectures) Chapters 12-13

- Network programming
- Concurrent programming