

General Information

Staff

Instructor:

Professor Alan Jay Smith, 735 Soda Hall, 642-5290, smith@eecs

Teaching Assistant:

Jingtao Wang, cs162-tb@imail.eecs.berkeley.edu

Readers:

To Be Announced

Office hours:

To Be Announced; see the office.hours file for times and places.

Classrooms, Times

Lecture:

Monday/Wednesday, 4-5:30pm, 247 Cory

Discussion Sections (*these may change!*):

101. Tu, 1-2pm, 4 Evans

102. Tu, 4-5pm, 285 Cory

NOTES:

- (a) Discussion times and rooms may change. Changes will be posted in `~cs162/Students/rooms_times`, and will be announced in lecture and discussion sections.
- (b) The section meetings are interchangeable; you may attend one or more at your convenience.

Exams

The first two midterms will be given in the evenings, 7:10-8:30pm, in a room to be announced. *Tentative* dates are Wednesday, March 4, 2009, and Wednesday, April 15, 2009. The third midterm (rather than a final exam) will be given Monday, May 11, 2009, at a time to be announced. (Either in class, starting at 4pm, or in the evening.) Exams will be *closed book*.

There will be review sessions prior to each midterm, beginning at 7pm, in a room to be announced. *Tentative* dates are March 2, April 13 and May 7.

There will be a Nachos tutorial lecture, starting at 7pm, tentatively on February 10, 2009, room to be announced.

Unless otherwise announced, all exams will be *closed book*. This means no textbooks, lecture notes, laptop computers, cellphones or programmable calculators. You can bring simple calculators (i.e. less than 8 bytes of memory), rulers, pens, pencils, (blank) scratch paper, and foreign language dictionaries (if you need one, and if English is not your native language). Please plan ahead - if you bring a programmable calculator, you won't be allowed to use it. Coverage on all exams will be inclusive and cumulative, from the start of the semester.

Exam dates may be changed - any change will be announced and posted.

If you have *any* conflict with any of the times for exams (as currently scheduled), you must tell the instructor as soon as you know about the conflict. Since the above exam times are being announced in the first lecture, they will and must take priority over exams for any other class unless you notify the instructor of the problem during the first week of classes (in which case, we will see what can be done, if anything).

Disabilities, Religious Conflicts

If we are required to give you an accommodation for a disability, you *must* notify the instructor within a week of the start of class, and provide documentation, so that we can make appropriate arrangements.

If you cannot attend a scheduled exam for religious reasons, you *must* notify the instructor within a week of the start of class, so that we can verify the conflict and make appropriate arrangements.

Grades

Your course grade will be based approximately 40% on the homework and 60% on the midterms. *There will be no makeup exams.* If you miss an exam, you should have an *extremely* good reason. Please note that in order to get a good grade in the course, you will need to do well on *both* the exams *and* the homework.

If you do have to miss an exam or homework deadline, and if you *do* have an *extremely good* reason, you must tell the instructor as soon as you are aware of the problem. Last minute requests will not be viewed favorably (except in the documented case of severe and sudden illness).

There will be no extra credit assignments or make-up work.

Grades will be assigned solely on the results of your exams and homework - i.e. everyone with a point total of between X and Y gets a grade of Z, etc. You will *not* be given a lower grade at your request (i.e. a D or F), so that you can retake the course. If you want a D or F, you'll have to earn it. Generally, skipping the last exam or not doing any homework is an easy way to be sure of an F. The grade distribution will be discussed in class after the first exam; the class average will usually be within the department grading guidelines.

Grades for homework will reflect peer evaluation (i.e. how much your group members said that you contributed). If your fellow group members say that you didn't do any work, your homework grade will be adjusted accordingly.

Any requests for grade changes or regrading *must must must* be made within two weeks of when the exam was returned, or when the grade for the assignment was first posted or May 7, 2009, whichever is earlier.

Class Attendance

You are responsible for any material covered in class (whether or not you were there) and for knowing about any announcements that were made (most of which are made at the *start of class*). The material in the exams comes mostly from what was covered in lecture, so I strongly recommend that you attend. The instructor is *not* teaching out of the book. The instructor will not repeat material from a previous lecture, and will not give you a private tutorial during office hours to cover material that you missed.

Textbooks and Readings

The primary (recommended) textbook for this course is:

Operating System Concepts, by Silberschatz, Galvin and Gagne, Wiley.

Note - the course will be based on the material that is covered in lectures. The textbook is primarily for background reading. If there is any conflict between what the textbook says and what the instructor says, you should assume that the instructor is correct (and in any case, the instructor writes the exams). If you do find differences between the textbook and what is said in class, however, do please *tell* the instructor, so that he is aware of it. You should be able to get by with an old or different edition of the textbook. The bookstore should have the 8th edition, but you can use an earlier one. The instructor will suggest sections of the textbook that you can read, but you can use older editions of the text (or even a different textbook, although Silberschatz is recommended and preferred).

Another recommended book for this course, which is *not* required, is: Leffler, McKusick, Karels, Quarterman, *The Design and Implementation of the 4.4BSD Unix Operating System*, Addison-Wesley. No readings will be assigned from this book, but you may wish to look up how Unix does things.

There will also be some readings (technical papers) for the course. The instructor will announce in class *when and where* the readings will be available; do not attempt to buy them before they are printed.

Printed copies of the code for the Nachos project will also be sold. Do *not* print your own copy.

Homework: Grading, Rules, etc.

The homework for this course will consist of (a) written problem sets, (b) programming assignments, and (c) taking notes on one of the lectures for the course, as described below.

Below are some general comments about logistics for the programming assignments. Corrections, updates and changes to these procedures will be posted and announced by the TA. There *will* be changes.

All homework for this class (including the "written" problem sets) must be done on the computer. You will be told by the TA how to turn in your assignments; that information will also be placed online. We may wish to examine your code, and/or run it, so we will need to be able to find it. In your class account cs162-xx, for each assignment, you should create a directory called HWn, where n=1...8. For the written problem sets, the text should be in a file called "text". For the programming assignments, your code should be in that directory. You should also have a file called "readme", which contains any necessary English language text to explain your results, etc.

Note - for the two written problem sets, the reader(s) may request that hard copy also be handed in. We'll announce the policy in time.

Every student (see below) will be given a class account. For the projects (see discussion below), you will be working in groups. *For two of the assignments (the two written problem sets), you will be doing individual work.* That means that for those assignments, you'll need to reset the password on any shared account(s), so that you have confidentiality for that work. To minimize the time when the 'group' account is inaccessible, you may wish to do those assignments on your personal (named) account, and copy it to your class account just before it is due. (You can get a named account if you are a CS major.)

All files containing your programming assignments (code, comments, output) should be retained until the end of the semester.

Note that there will be a disk quota - you only have a limited amount of disk space. You may need to delete executables and browser caches.

For all programming assignments, you should write the code in Java.

All homework should be clearly identified with your name(s) and electronic address(es) in the first comments in "main" and at the top of "text" or "readme" file.

The grading of the Nachos assignment code will be done by an autograder, under the control of the reader(s). Information on how to prepare your code for use by the autograder will be posted online and given out by the TA. The reader(s) will grade the other homework and any written design reviews. Please see the reader(s) *first* if you have a question about your grade, but not outside of his/her office hour(s) - s/he doesn't have many hours to get the grading done. If there is still a problem, see your TA next; if the problem is still not resolved, then see the instructor. Please note that if there is a general problem with the grading (e.g. randomness, grades all too low, or all too high), the instructor won't know if you don't tell him - please don't suffer in silence.

Homeworks are due at varying times of the day - please check both the due date *and* time. The policy on late homework is: (a) You have a total of 5 days of permitted "lateness," total for the semester. (All late amounts will be computed in integer days, rounded up.) If your homework is late, you can say in the readme file for that assignment or in the comments at the top of "main" and "text" or "readme", that you are using "*N*" days of the grace period. (Obviously, $N \leq 5$.) (b) After the *N* days are used up, your credit for each late assignment declines 10% per day, linearly from the original amount. No homework is accepted more than 5 days late. (c) The two written assignments will not be accepted (at all) later than 5pm on the day that they are due. (d) Nothing, except the lecture notes assignment, will be accepted after May 5, 2009. (All lecture notes (assignment 8) must be handed in no later than May 7 - see below.) Please use your own judgement as to whether to hand in a partially correct assignment on time or a fully correct assignment late.

In grading, we will expect all of the usual things: your programs give the correct answers, are well structured and readable, are thoroughly (!) documented, and don't crash when fed bad inputs. The text of any written problem sets should be in clear, correct, standard English. Figures should be drawn as best you can using the characters available on the keyboard.

Homework Schedule

The *tentative* schedule for homework is as follows. This schedule *may* change - *please check*.

February 9 (3:45pm) - assignment 1, CPU scheduling simulation.

February 24 (11:59p) - Nachos1

March 2 (3:45pm) - Problem Set 1

March 16 (3:45pm) - Nachos2

April 8 (3:45pm) - Nachos3

April 13 - (3:45pm) Problem Set 2

April 30 - (11:59p) Nachos4

date variable - (11:59p) assignment 8, lecture notes.

Design Reviews

There will be design review meetings between each group and the teaching assistant for some of the Nachos assignments. Information on how to arrange your meetings will be posted online.

Individual Work, Group Work and Cheating

For the two problem sets, you must do individual work only. For the programming assignments, you may work in groups of up to 5 if you wish. You may also work alone or in a smaller group. The members of your group should be identified in the *readme* file of your assignment and/or in the comments to the code. Each group will hand in one solution for each assignment. The members of the group may change between assignments, but changes in group makeup are *not* recommended, since assignments will build upon previous assignments and it is difficult to work with unfamiliar code.

You're allowed to work in smaller groups or even individually, but you won't receive any extra credit for doing more work because your group is small. Learning to work in groups is an important skill, and we recommend that you do so.

The TA will serve as a matchmaker if you are unable to find a group to join.

Members of each programming group will be *required* to evaluate the contributions of each of the other members of the group, and *the grades of each individual will be adjusted accordingly*. A procedure for this evaluation will be announced and posted. If you don't turn in the required evaluations of your group mates, your own grades will be lowered.

You (or your group) should work out and write your own answers to the questions, and/or write your own code. If we find two assignments that are more similar than could be reasonably expected from individual work, it will be considered cheating. This means that you shouldn't copy from each other or from other sources, such as textbooks or previous years' assignments. If we find two substantially identical assignments, you will be considered to have cheated, even if you both copied the answers from some other publically available source (a textbook) or from old homework in your fraternity, sorority, dorm or HKN files.

Note that I encourage you to talk with your friends about the work you are doing for this class; both giving and taking advice will help you to learn. However, what you turn in must be your own, or your group's own work: it's OK to talk about algorithms and approaches, but you must write your own programs and problem set solutions.

Exception - if individuals change groups, there may be instances of two or more groups starting with the same base code for a given assignment. If that occurs, please document that fact clearly in the comments to your code.

We have an automated system for detecting cheating: we have a program which will pairwise compare all homework assignments with all others, and report cases of suspicious similarity. (See the article in the class readings.) Any such similarity will be checked by the TA and/or instructor. If two assignments are determined to be obviously very similar (i.e. we believe that they were done together or one was copied from the other), then the course grade for *all* students involved in the incident will be reduced by one full grade for the first offense, and to an F in the course for the second offense. ("All" means both the copy-er and the copy-ee.) (The grade for that assignment will also be reduced to 0.) *The reduction in grade will be taken without discussion or warning; the first notice*

you receive will probably be a letter indicating the penalty. In addition, for every instance, a letter to the Office of Student Conduct will be attached to your permanent record, and a copy will be placed in the CS division office and the EECS department office. To facilitate this automatic checking, you must use the standard names for your assignments, as described above. If we can't find your assignment on-line, your grade for it will be reduced to zero whether or not you handed it in. You must retain your assignment on-line until the end of the semester.

Other types of cheating include such as copying someone else's work without their knowledge, cheating on exams, etc. These types of cheating are considered more serious, and (for a first offense) will typically result in the person(s) cheating receiving an F, and having a letter placed in their permanent files in the Office of Student Conduct, in the CS division office and in the EECS Department office. Note that you are responsible for not leaving copies of your assignments lying around and for protecting your files. (You must set up your files so that they are protected from anyone other than members of your group reading them.) If it appears that the copy-ee cooperated with the copy-er, both will be considered to have cheated. If the copy-ee failed to safeguard his or her work, that person's grade will be reduced.

Our current plan (subject to change) is to post your final grades (see below) in a readable file in the cs162/Students directory, indexed by code number. We may also post other grades that way, or we may use glookup. Please be sure to check your grades regularly! If the posted grade is wrong or missing (once the grading for that assignment is complete), you have only 2 weeks to notify the TA and/or instructor.

Code Number

University policy is that grades and other information should not be posted by either name or student ID. Therefore, you will need to create a 7 digit alphanumeric code, which we will use for posting grades. Your code number should be a 7-character alphanumeric string consisting of only the characters A...Z, 0...9. You **must must must** put your code number on the computer account form that we will require that you fill out to get your computer account. If your code number is accidentally the same as that chosen by someone else, we will change it and notify you of the new code number. Your code number should not be the leading 7 digits of your student ID number. (It can be the trailing 7 digits if you wish.) *You should write down and remember your code number!*

Prerequisites

The prerequisites for this course include 61B and 61C, and Math 55 or CS 70. This means, in particular, that you know Java, data structures (especially linked lists, storage allocation, garbage collection, trees), have done some assembly language programming, know about linkers and loaders, and that you know about series and products, proofs by induction, logarithms, matrices, advanced algebra, some calculus, basic probability (means, standard deviations, cumulative and density distributions, variance, etc.), and basic CPU architecture as covered in CS61C (e.g. cache memories, TLBs, I/O, interrupts, task switching, etc.). (We may spend a *small* amount of time reviewing the necessary data structures, computer architecture, and probability and statistics.) We will assume that you either know the material that is supposed to be covered in those courses or are willing to learn it when necessary. *We may briefly review some of this material in class, but you are responsible for the necessary background, whether or not it is reviewed in lecture or discussion sections.* In particular, you are expected to *remember* the material from those courses as needed.

Computing Facilities

All students in the class will be given an instructional account cs162-xx on the instructional computers. (Note: the URLs given below are believed to be correct. If they're out of date, please tell us and we'll get you the correct ones.) You should see <http://inst.eecs.berkeley.edu> and <http://inst.eecs.berkeley.edu/connecting.html> for information. You can see inst.eecs.berkeley.edu/~iesg/iesglabs.html for a list of systems you can use. *Your final run for the assignment must be done under that account. The input, output and code for your assignment must be saved in that account so that the TA and/or Readers can examine them, using the standard names for that assignment (see above). The assignment must be kept on-line until the end of the semester.* If you are doing some of the work on another machine, it is your responsibility (and your problem) to transfer your files to the class machine for the final run. Note that for some of the assignments, there will be input data or code available only on the machines on which

the class accounts are located. Your account files **must** be protected against reading or writing by others; your account will be set up this way. If we find that your files are readable, your account will be turned off and your course grade will be reduced.

Accounts for the machine will be given out *only* during discussion section meetings and/or by the TA during his office hours. You will need to fill out a special form, to be given to the TA, before you get your account. There is a writeup, "User's Guide to EECS Instructional Computing", at <http://inst.eecs.berkeley.edu/share/b/pub/html/Users.Guide/> which provides an introduction to the EECS instructional computers and to Unix use. See also the /usr/pub files, which have considerable "help" information. Accounts that we provide that have not been used will be canceled at some unpredictable time during the semester, so be sure to immediately log on to your account and reset the password. Accounts for those students who don't actually register for the course will also be canceled.

You *must* set your "finger" information with your correct, full name; no nicknames. In the ".plan" file, put the names and other relevant information for those using that account.

Java is installed in /usr/sw; reference is <http://inst.eecs.berkeley.edu/usr/pub/java.help>.

The WEB page is <http://inst.eecs.berkeley.edu/~cs162>.

Lab access

All Cory and Soda Hall UNIX labs will be unlocked from 7am-6:30pm on weekdays. Cardkeys will be required after that. Certain rooms (e.g. PC labs) always need cardkeys. You should be able to get a cardkey just by having your name on the official enrollment list; go to 253 Cory (8:30am-12, 1pm-4:30pm) to sign up for a new cardkey. To validate an existing card key, you can go to 253 Cory. For students who add the course, and for extension students, the instructor will forward your name to authorize your cardkey.

There are various Unix workstations in various rooms on the second floor of Soda and 199 Cory.

Electronic Mail and Class News

Information will be sent by the instructor, teaching assistants and readers to the students in this course via class newsgroup or electronic mail to their class computer accounts and/or will be posted on the class web site. We will assume that you read that mail and/or the website at least once a day, and we will hold you responsible for such messages. Such messages will include changes in dates for exams and homework, announcements about review sessions and meetings, and changes and corrections to homework assignments. You can forward mail to another account by using a ".forward" file. We'll also put information in special files - see below.

Missed Classes

You should get the names and phone numbers of several other people in the class, so that you can get the notes and handouts for any classes that you miss.

Cancelled Classes

This course normally has 28 lecture periods, of which two are used for midterms - i.e. there are about 26 lectures when Prof. Smith teaches this course. This semester two or three of the midterms are being given in the evening, and there are actually 29 lecture periods available, so there will be two or three surplus lecture periods. The current plan is to therefore cancel two lectures, dates to be determined. (E.g. when the instructor is out of town.)

Lecture Notes Assignment

Because some students will miss some lectures (for various reasons), assignment number 8 requires that you take notes on one class lecture; those lecture notes will be posted.

At the first class, I will pass out a sign-up sheet. You *must* sign up for one lecture on which you will take notes. For that lecture, you must take *careful* notes, and type them into the computer, as cs162-xx/HW8/text. (Your notes should include what is on the board and also what I say. If necessary, you should look at the text and readings and include any additional information necessary to make the notes clear. Be sure to include any announcements,

such as due dates, lists of handouts, etc. If all you do is copy the lecture notes from the screen, you will receive only minimal credit. You should write up something that someone who didn't attend the lecture will understand well enough to study from.) There should be 1-2 students signed up for each lecture. You can each do your own set of notes, or you can collaborate; if there are 2 students signed up for a given lecture, from one to two sets of notes should be turned in, depending on how you chose to organize the note taking. (Obviously, a set of notes done by two people should be a better than a set done by one person.) The lecture notes can be plain text or pdf (no .doc files).

The lecture notes assignment is due by 11:59p one week after the lecture for which you are to take notes. (e.g. if you are taking notes for July 5, your notes must be handed in by 11:59p, July 12.) Your lecture notes should be emailed to cs162-tb. These notes will be graded for quality and correctness, and will be posted online in ~cs162/Students/Lecture_Notes, with the grade for the notes and the name(s) of the notetakers showing in the file. Please note that everyone will be able to see the quality of your lecture notes, so please do a good job.

(If you don't yet have a class account, your writeup won't be due until one week after you receive your account. If you sign up for the last week of classes, your lecture notes will be due at 11:59p May 7, 2009. If you sign up for a class that is cancelled, we will arrange for you to take notes for another, later, lecture.)

Note that since lecture notes are not due until a week after a lecture, and may not be posted for a few days after that, you can't count on having all of the lecture notes available before an exam.

Information, Special Files and Web Site

We will put information you need in various special files in the directory ~cs162/Students. (Note that the ~cs162 account is not readable. You have to do a "cd ~cs162/Students"). This material/files will also be available on the class website:

<http://inst.eecs.berkeley.edu/~cs162>.

The following are (tentative) names for some of the files and directories that should be in cs162/Students. There will be other files here as well.

Handouts - a directory with copies of class handouts (e.g. this one).

Homework - a directory with copies of the homework assignments. We'll post them here as soon as they are ready. (This directory should always have the latest copy, in case changes have been made.)

info - an append only file with things that we want to tell you.

Info_nachos - a directory with assorted Nachos information.

grades - a file with grades (indexed by student code number).

office_hours - a file containing office hours for the instructor, TA and Readers.

Lecture_Notes_09.s - lecture notes for spring, 2009 for CS 162.