CMPS 221: Advanced Operating Systems

Computer Science Department
University of California, Santa Cruz

Winter 2016

Time: Tue & Thu 4:00–5:45 PM
Location: Kresge 323
Instructor: Professor Ethan Miller (office: Engineering 2 365, elm@ucsc)
Office hours: Tue 11:00–noon; Thu 2:00–3:00
Prerequisites: CMPS 111 or equivalent undergraduate operating systems class
Readings: List of papers available from course web site
Course web site: http://classes.soe.ucsc.edu/classes/cmps221/Winter16/

Course objectives

This course is a graduate level study of the issues in the design and implementation of computer operating systems. The readings are taken from the current research literature and articles of historical significance. The topics include, but are not limited to:

- Early systems & history
- Kernel structures
- Memory management
- Virtualization
- Scheduling and synchronization
- File systems
- Authentication, protection, and security
- Performance evaluation
- Reliability and fault tolerance
- Communication and distributed systems

The course proceeds by topic, beginning with systems of historical significance. The focus will be on current research in the area, so the time allocated to each topic will depend on the availability of recent papers. Additional topics may be covered, depending on the interests of the students and the professor.
Preparation

You are expected to have basic operating system knowledge, such that provided by an undergraduate course such as CMPS 111, offered here. Undergraduates will be admitted with the permission of the instructor.

Web pages

All of the information for this class will be distributed via the Internet. The class home page is at the URL listed above, and is the starting point for lots of information about the class including readings lists, assignments, and notes. We’ll be using Piazza (linked from the course web page) for discussions and class announcements. You’ll need to register for Piazza in order to be able to view and post messages. Part of your class participation grade will depend on active participation on Piazza.

Access to some of the class web pages is restricted to users on campus. In particular, many papers can only be downloaded from a computer in the it.ucsc.edu domain (this is done for copyright reasons). If you want to access restricted web pages or papers from off-campus, you can use the campus-wide VPN or SOE VPN. You may also get papers from other students who’ve downloaded them from the course web page. Please do not download papers from random online sites such as Google Scholar unless you’re certain that the paper is the authoritative version—many papers exist in multiple versions, and it’s important to ensure that you get the right one.

Course requirements

Because this is a seminar, the majority of the course work will involve the reading, class discussions, paper presentations, the final project, and the final exam.

Readings

A major component of this course will be the in-class discussion of papers on research in operating systems. Typically, you will need to read two papers per class; the reading list is available from the course web site, and all of the papers are available as links from the reading list (though only to computers in the it.ucsc.edu domain). These papers should be read carefully, and a short summary of each article and a few questions or insightful comments about the material (at least 3 per paper) prepared for the class meeting in which the paper will be discussed. The summary of each article consists of brief answers to the following five questions, and three comments or questions about the paper (the sixth item):

- What is the problem the authors are trying to solve?
- Why is it interesting, relevant, and/or important?
- What other approaches or solutions existed at the time that this work was done?
- What was wrong with the other approaches or solutions?
- What is the authors’ approach or solution, and how does it compare to earlier approaches or solutions?
- List Three or more comments/questions about the paper.

Paper summaries

Each student is expected to write a summary of each paper as described above. The summaries for the papers assigned for a course meeting must be turned in by 10 AM on the day the papers will be discussed. Summaries must be turned in using an online form available from the course Web site. The summary submission form will require your CruzID and Blue password.

Summaries will be graded on a 0–2 scale (0 = not turned in, 1 = minimal effort, 2 = satisfactory). Students may find it useful to bring printouts of their summaries to class discussions.
Paper presentations & discussions

Every student in the class will lead the class discussion of one or two papers during the quarter; the exact number depends on the total number of enrolled students. Needless to say, you need to be in class on the that “your” paper is being discussed. You may get materials from anywhere you like, including the original paper authors, as long as you cite your sources. However, **you must prepare your own slides**; part of the goal of giving the talk is to give you practice presenting technical material.

Students are all required to participate actively in discussing the papers we read. This means that you’re required to attend class; absences for illness and conferences are expected, but you should attend at least 80% of the classes during the quarter. We won’t take attendance, but your class participation grade will suffer greatly if you’re not in class and participating actively in discussions.

Final exam

There will be a take-home (open book) final exam, given the last week of classes and due on Wednesday, March 16th at 7 PM (the end of the university-assigned final exam slot).

Final project

Students in the class must complete a project in the general area of operating systems. Both a paper describing the project and a brief (5–6 minute) “work in progress” presentation will be required. The project may be a research project, in which case it may be a collaboration between several students in the class, or it may be a detailed survey paper on a focused research area related to operating systems, in which case it must be done as an individual project.

Research projects (but not survey papers) may be shared between classes with prior explicit permission from the instructors of both classes; if a project is shared between classes, expectations will be significantly higher for the content of the project.

While you’re encouraged to use resources available on the Web and elsewhere (see below on how to cite material), I expect **you** to put a “reasonable” amount of effort into your class project. A project that requires 5 hours of time to compile and run already-existing software isn’t much of a project, and will be graded accordingly. Your project should take approximately 60–80 hours over the course of the quarter, including time to read background material, build and run your experiments, and write up your results.

If you want to work with up to three students in the class on your research project, you may do so with **prior approval** (*i.e.*, please see the instructor before doing this). If you work in a group, the expectations for the scope of your research project will be adjusted accordingly.

There will be checkpoints about every two weeks during the quarter to keep you on schedule to complete your project. Please see the course web site for a list of checkpoints and dates. For each checkpoint, you should submit a few paragraphs describing your progress. For the bibliography checkpoint, please submit your bibliography. Obviously, the last two checkpoints require more work; they’ll require an actual paper or presentation.

More details and sample project topics will be available during the first week of class.

Attendance

Class attendance is mandatory. Because this is a graduate class, I expect students to participate actively in class, and that’s hard to do if you’re not actually there. I won’t take attendance at class (except as necessary to make the registrar happy), but you cannot pass if you miss too many classes. If you need to miss a class for a good reason, such as a conference or other research-oriented commitment, please see me in advance if possible.
Grades

Your grades will be determined as follows:

- Class participation & summaries: 40%
- Paper presentation: 10%
- Final project: 30%
- Final exam: 20%

You must turn in a final project (and present your project) and turn in a final exam to pass the class.

Getting help

You’re strongly encouraged to seek help if you need it. You can do this by going to office hours, reading the Piazza forums, or by email. Office hours are optional, but highly recommended if you’re having any difficulty understanding the material, preparing your presentation, or working on your project. More in-depth discussions of OS-related topics are also appropriate (and encouraged) during office hours. As a further incentive, I’ll have free espresso available in my office during office hours. You’re welcome to use the course forum and send email at any time, but please arrange any meetings outside of office hours in advance.

Email to the instructor or will be answered if possible, typically within one business day—if you want short turnaround time, go to office hours. The best kinds of questions to ask via email are those that require short answers but are irrelevant to others in the class. Questions like “why doesn’t my code work?” and “please explain this concept to me” are too difficult to answer via email, and are best asked and answered in person at office hours.

Academic Honesty

This is a graduate seminar—I encourage you to discuss the material with other students in the class and perhaps others outside the class. However, all paper summaries must be your own. If you turn in a summary substantially similar to someone else’s, you will receive a -2 for that summary. If this happens more than once, you will fail the class and formal cheating procedures will be begun. I’d rather see you miss a summary than copy someone else’s. If someone else’s ideas make it into your summary or final project, you must formally acknowledge their contributions when you turn in the material.

You’re encouraged to use any resources (code, traces, etc.) you can get for your projects, as long as you properly attribute them in your paper. Science is a collaborative enterprise; as Newton said, “If I have seen a little further it is by standing on the shoulders of Giants” [Wikipedia]. By making use of what others have already done, you can accomplish a great deal more in a quarter than you could otherwise. Science is built on properly crediting those whose work you use; failure to do this will not be tolerated. Improper use of others’ work in your project will result in a failing grade in the class, but it’s not as bad as improperly using others’ work in your own (independent) research, which can earn you (relatively) permanent disgrace. In short, attribute everything! If you’re not sure about the ethicality of something, feel free to talk to me before you do something.