

CPS 110: Operating Systems

Course Overview

Spring 1999

Instructor: Jeff Chase (chase@cs.duke.edu)

Class Meetings: Tuesday and Thursday, 9:10 - 10:25 (D106 LSRC)

Teaching Assistant: Rajiv Wickremesinghe (rajiv@cs.duke.edu)

Course Web site: <http://www.cs.duke.edu/~chase/cps110>

This is an introductory undergraduate course in operating systems, emphasizing the core operating system concepts of protected kernels, processes and threads, concurrency and synchronizations, file systems, and virtual memory. A major component of the course is a series of projects using the infamous Nachos instructional operating system.

Who should take this course. All undergraduate CS majors should take this course at some point. You should have completed the basic introductory sequence, and also CPS 104. Familiarity with C, C++, and basic Unix program development tools (e.g., *gmake*) is assumed.

Workload. This is a demanding course. There are seven required Nachos programming assignments, with one due every two weeks. To make life a bit easier, you will work through the Nachos projects in teams of three or four students. As in previous offerings of CPS 110, you will present your projects in demo sessions with me or with a TA. These projects will be fun and valuable if you form teams quickly, start early on the projects, and make your Nachos work a priority this semester. In addition to the Nachos assignments there will be two or three closed-book exams and optional problem sets.

Topics. The course covers four primary topic areas.

- *Concepts in operating systems.* Operating system services, operating system structure, processors and processes, protected kernels, traps, interrupts, processes, and threads.
- *Concurrency, synchronization, and communication.* Threads, mutual exclusion, events, scheduling, and deadlock. Implementing and using synchronization.
- *Virtual memory and file systems.* Address spaces, protection, virtual address translation, control of memory management hardware, and physical memory management. File and directory structures, file cache management, dealing with disks, failure and recovery, access control and security.
- *Introduction to networking and distributed systems.* Network communication, network file systems, and other networked operating system services. Problems and challenges in distributed computing.

Readings and resources. The readings are from the book *Operating Systems: A Modern Perspective*, by Gary Nutt. All course materials, lecture slides, and announcements are featured on the course web site. You will visit this web site often, as we will rarely hand out printed materials in class. Grades are also distributed through the web: please see the TA for a password.

There is also a course newsgroup *duke.cs.cps110*. We prefer that you post your questions to the newsgroup rather than e-mailing us. We read the newsgroup regularly, and so should you: it is embarrassing to ask a question or waste time struggling with a problem that has already been discussed in the newsgroup. If you know the answer to a posted question, please post a response.

Grading. The exams and Nachos projects are each worth half of 95% of your grade. The remaining 5% of your grade is based on “subjective factors”. The problem sets are ungraded, but many of the exam questions are drawn from or derived from the problem sets. There will be ample opportunity for optional extra credit in the Nachos projects.

I will try very hard to give you the grade you deserve. I always reward good work, but I reserve the top grades (A and A+) for students who have truly impressed me in some way. To assign grades, I sort the final averages and look for natural break points to determine the boundaries between grade levels. Next, I consider extra credit to raise some grades. Finally, I consider other factors to identify students whose numbers do not tell the whole story. These factors include: trends of improvement or decline, the nature of exam errors, Nachos group dynamics, willingness to help other students, and the relative strength of exams and project grades.

In assigning the grades I take into account that CPS 110 is an unusually difficult and time-consuming course. You should also take this into account when planning how to use your time. In short, you must work very hard to earn a good grade in this class. With that said, I am proud of the performance of the Duke undergraduates who have taken this course with me in the past, and I am confident that you will meet the same high standard this semester.

Policy on collaboration for CPS 110. We encourage any and all collaboration except during exams. The course will be more productive and more fun for everybody if you take the time to help each other out. However, you must take responsibility for doing your share of the work. In general, I expect every team member to be familiar with the approaches taken for each project, whether or not they wrote the code. To encourage everyone to participate fully, we make sure that every student is given an opportunity to explain and justify their group’s choices and approach. We may ask questions about who wrote the code, which you will answer honestly. If we do not ask, it is considered bad form to claim individual credit for your group’s work.