

Syllabus

CS 226: OS and networking

Spring 2009

Time: MWRF 5th hour
Room: MWF: SMC D-205
Th: SMC A-215 (C&B Lab)
Website: <http://courses.knox.edu/cs226/>

Professor: Don Blaheta
Email: dblaheta@knox.edu
Office: SMC E-210
Office hours: Monday 6th; Tuesday 5th; Thursday 4th;
and by appointment or walk-in

Overview

In this class, we study principles and implementations of operating systems and networking. The operating system manages hardware resources and provides a simplified interface for programs to use these resources. Networking allows different computers to communicate and potentially act as a larger virtual system. These topics are closely related; networking is often managed by the operating system (and always requires use of the hardware it manages) and the operating system uses the network to provide services like the file system. To facilitate our study of these topics, we will learn the C language, which provides low-level access to the hardware and is often used in operating systems and networking.

Books

The two books I want you to buy are actually reference books:

- Peter Prinz & Ulla Kirch-Prinz, *C Pocket Reference*.
- Michael J. Donahoo & Kenneth L. Calvert, *TCP/IP sockets in C*.

The reason I'm having you buy them is to have them next to you when coding—far more convenient than having to use up valuable screen real

estate on reference. They're not the sort of thing you're likely to read and then set aside.

In contrast, the *textbooks* for the course I'm *not* requiring you to buy, because they'll be on reserve and you can read them there (and we're only using each one for a few weeks):

- Andrew S. Tanenbaum, *Modern operating systems*, 3e.
- Andrew S. Tanenbaum, *Computer networks*, 4e.

If you're the sort that likes to buy books, these are also the books we tend to use in CS 305 and CS 308 respectively (but no promises), so they might be worth acquiring. You can also get older editions for pretty cheap online, and those would probably be suitable for the purposes of this course.

Contacting me

Any time my door is open (which is most of the time) you should feel free to come in and ask questions or just to chat. If you can't catch me in my office, email is probably your best bet. If you don't get a quick response to email, I do respond well to (light) nagging—you might want to try sending another one.

Red Room

The Red Room is not just for 100-level classes! This term, two of the CS tutors (Ellie on Wednesdays and Matt on Thursdays) are eligible to assist CS 226 students. Don't make them sit all lonely waiting for a 142 student to drop by.

Graded work

I figure that I have about a third of your time over the course of the term, or about 12–15 hours a week including class. This time is spread over a number of different types of graded assignments. Note that this means I expect roughly 4–5 hours outside of class for each 5% of the grade—budget your time accordingly.

Homework. Homework will go out relatively frequently, and will often be short and due soon. They will typically be a mix of on-paper work (e.g. simulating an algorithm, describing concepts) and implementation problems. Feel free to discuss them amongst yourselves, but be certain to do all the writing yourself (see the collaboration policy; homeworks and projects are what I term “lightly collaborative”).

Projects. There will be two substantial implementation projects in the later portion of the course, which you will have two weeks to work on. As always in one of my courses, start *early* on these so you can come for help if you get stuck! You can also talk amongst yourselves as long as you aren’t writing each other’s code; again, see the collaboration policy.

Exams. There will be two exams, one at midterm and one for the final. Both will be take-home, and you will be given several days to work on them. These will be non-collaborative, and you are not permitted to discuss the exam with anyone other than me.

Labs and participation. A small portion of the grade is devoted to making sure you’re attentive in lecture and active in labs. Occasionally asking questions, or answering them, is a good way to lock in these points. With respect to labs, you don’t lose points for not finishing the lab in the allotted time, nor do you lose points for finishing early and leaving (after checking with me); mostly you just lose points if you’re goofing off (or absent). So don’t do that.

Breakdown

Homework	35%	
Projects	20%	(10 each)
Exams	40%	(20 each)
Participation	5%	

Schedule

Homeworks will go out intermittently and typically be due the following class period. All assignments are due at the beginning of class on the specified due date.

The exams and projects are tentatively scheduled as follows:

	Out	Due
Midterm exam	Mon, 20 Apr	Thu, 23 Apr
Project 1	Fri, 24 Apr	Fri, 8 May
Project 2	Fri, 15 May	Fri, 29 May
Final exam	Fri, 29 May	as announced

The midterm and projects are due at the start of class on the due date (modulo the Flunk Day policy, see below). The final exam will be due at the start of the course's official exam period. You are, of course, permitted to hand it in early, either in person or under my office door.

Topics

Week 0

Administrivia, intro to C; general C syntax (CPR pp1–42, 49–52, 56–62); Intro to pointers.

Week 1

Pointer arithmetic; Structs, standard C libraries, OS calls (MOS §1.6); man pages, header files (CPR pp43–55); History of Oses, hardware basis (MOS §§1.1–1.4);

Week 2

Overview of Oses (MOS §1.5); Files and directories (MOS §§4.1–4.2); Accessing directories; Implementing files and directories (MOS §4.3).

Week 3

Filesystem issues (MOS §4.4); Memory management (MOS §§3.1–3.2); Using dynamically-allocated memory (CPR pp110–11); Virtual memory (MOS §3.3).

Week 4

Processes (MOS §2.1); Inter-process communication (MOS §2.3); Input redirection; IPC continued.

Week 5

The dining philosophers problem (MOS §2.5); Threads (MOS §2.2); Using threads; Deadlock (MOS §§6.1–6.4).

Week 6

Network layers and reference models (CN §§1.3–1.4, TSIC §1.1); The transport layer (CN §§6.1–6.2), TCP model (CN §6.5.2); Sockets and data transfer (TSIC Chh1–3); TCP continued (CN §6.5).

Week 7

Checksums and error detection (CN §3.2); UDP (TSIC Ch4); Managing multiple TCP connections (TSIC §5.4); DNS (CN §7.1, TSIC Ch7), HTTP (CN §7.3).

Week 8

Email: SMTP, POP, IMAP (CN §7.2); Base64, MIME (CN §7.2.3); Separate compilation, make, ifndef (MOS 1.8); The network layer (CN §5.1).

Week 9

IP, NAT, DHCP (CN §§5.6.1–5.6.3); ALOHA, MAC protocols (CN §4.2); Debugging with gdb; Ethernet (CN §4.3).

Policies**The “read the damn spec already” policy**

It is vitally important that you, as a budding computer scientist, learn to write programs according to spec. In addition, although I do give partial credit, my grading task becomes infeasible, or at least a lot harder, if I have to fix your code before I can even test it.

Therefore, I reserve the right to impose draconian penalties if your program doesn’t even compile, or doesn’t accept the right command line, or other basic spec violations. Please, just reread the problem before you hand in and make sure you’ve set it up right, ok?

Collaboration policy

Exams are noncollaborative. The other assignments (homeworks, projects) are “lightly collaborative”. What I mean by this can be found at <http://faculty.knox.edu/dblaheta/collab.html> .

Systems and environments

The supported systems in this class will be the department Macs in the Crash-and-Burn lab. You are welcome to use any systems you like for development, but the work you hand in has to be able to run on those machines. Also, I can't guarantee that I'll be able to help as much on other systems.

Flunk Day

Nobody knows when Flunk Day will be, of course, and class won't meet that day, obviously. If Flunk Day falls on the day something is due *or* the day before, then the deadline for that assignment is automatically extended to the next class period. Have fun!

Attendance and late policy

Attendance is required, and assignments must be turned in on time. That said, if you have a good reason to miss class or hand something in late, I tend to be fairly easygoing about it; but I need to hear about it in advance. (Medical and family emergencies excepted, of course. But get a note from a dean.) Extensions of a day or two will be granted fairly routinely, as long as the reason is something better than "I didn't start until today, and now I can't finish on time!" If in doubt, come talk to me. Frequent absence will result in a lowered participation grade; habitual absence may in extreme cases result in a failing grade for the class. Unexcused late assignments will normally be given a zero.

Early bird policy

My standard early bird policy applies: extra credit if you find an error in the problem definition (for homework, project, exam, whatever). See <http://faculty.knox.edu/dblaheta/policies.html#early> for details.