Main Syllabus Staff Piazza

CS 168

Introduction to the Internet: Architecture and Protocols, Fall 2018

Instructor: Scott Shenker
Lecture: Tu/Th 6:30-8:00 pm, 1 Pimentel
OH: M 4:00-5:00 pm, 415 Soda

TOC

Description

This is an undergraduate level course covering the fundamental concepts of networking as embodied in the Internet. The course will cover a wide range of topics; see the lecture schedule for more details. While the class has a textbook, we will not follow its order of presentation but will instead use the have several projects that involve programming (mostly in Python).

Prerequisites

Required: CS 61A, 61B, Math 53 or 54.

You should know programming, data structures, and software engineering. In terms of mathematics, your algebra should be very solid, you need to know basic probability, and you should be comfortable with thinking abstractly. The TAs will spend very little time reviewing material that is not specific to networking. We assume that you either know the material covered in those courses, or are willing to learn the material as necessary. We won’t cover any of this material in lecture.

The projects will be programmed in Python, but we expect that it will be easy for students to learn enough Python on-the-fly to complete the assignments.

Textbooks

Recommended:

- Computer Networking: A Top-Down Approach, 7th edition by Jim Kurose and Keith Ross.

Computing Facilities

Students enrolled in the class will be given an instructional account, `cs168-???@cory` . To get an account, register yourself here. Most of the Unix systems have cross-mounted file systems, so you can generally work on other EECS Unix systems. Your final run for each assignment must be done under that account, and must run on x86 Ubuntu machines.

Homework Assignments / Self-Tests

There is no homework for this class. Instead, we will be releasing online self-tests after each lecture. These self-tests are meant to gauge which topics we should focus on more in future discussions and lectures. They will be graded on timely completion (before the start of next lecture), but we will penalize anyone that doesn’t put in the effort on these self-tests (e.g., answering tests without being penalized.

Projects

Projects will be graded by a script which will run the submitted code under a variety of conditions. Grades will be determined automatically from the results from the script. Projects are to be completed individually. **You are not allowed to work with a partner.**

You may not share code with any classmates. You may discuss the assignment requirements or general programming decisions (e.g., what data structures were used to store routing tables) - away from a computer and without sharing code - but you should not discuss the detailed nature of your solution (e.g., what algorithm was used to compute the routing table). Also, it is your responsibility to **make sure your code is not publically accessible to others in any way**. Your grade will suffer if you violate this requirement.

For example, if you are using Github, or other such services, ensure that your repository access is set to private (as a student, you have unlimited private repositories on GitHub!). Likewise, ensure that you do not put your code on public clipboard sites such as pastebin. If we find that someone has copied your code (even without your knowledge), you will be deemed equally responsible and will face the same penalty as the person who copied it.

Exams

There will be one midterm and one final. If you have a conflict with any of the exams, let us know as soon as possible to attempt to schedule a makeup.

- The midterm exam will be closed book with a single 8.5” by 11” (both sides) crib sheet.
- The final exam will also be closed book, with two 8.5” by 11” (both sides) crib sheets. So you can keep your midterm sheet to reuse it for the final.
- **The crib sheets must be handwritten.[1]** E.g., copy&paste of lecture slides is forbidden.

The exams will cover material from lecture, sections, and projects.

[1] Crib sheets written on a tablet and printed out are allowed, but you must be the writer of such sheets.

The course will be graded on a curve that is consistent with new guidelines, but on the lower end. A+ is reserved for the very few best students in the class. Graduate students and re-entry students are not included in establishing the curve, but they will receive grades based on where they would fall on the curve.

Your grade is based on a combination of projects, and exams, in the following proportions.

Category	Percentage
Project 1	20%
Project 2	20%
Self-Tests	5%
Midterm Exam	25%
Final Exam	30%
Total	100%

Participation

In order to pass this class, you must participate at least once during the semester. Participation can either be asking a question in lecture or coming to Prof. Shenker's office hours. You must record your participation by filling out this form, so we can give you credit. You should only fill out the form once. **Students who have not participated by the end of the semester will fail the course.**

Late Policy

The policy is simple: **no slip dates**. If assignments are late, they are penalized as follows: < 24 hours late, you lose 10% (0.9 * your raw score); < 48 hours, 20% (0.8 * your raw score); < 72 hours, 40% (0.6 * your raw score). More than three days late, 50% (0.5 * your raw score); you may only submit **ONCE** if you are more than three days late. If you receive less than 50% on any project, you may choose to redo it **ONCE** for up to 50% of the maximum score. Any project turned in after December 7, 11:59 PM will not receive any credit.

Regrades

Any requests for grade changes or regrading must be made before the specified deadline. For assignments returned on Gradescope, you should submit regrade requests directly through Gradescope. Please submit a separate request for each question you want regraded. For assignments graded in a different way than on Gradescope, we will post a regrade request Google form on Piazza.

Even if you ask for only one problem to be regraded, your entire work may be regraded, so your score could decrease, stay the same, or increase.

For Projects

Each project would have its own regrade policy defined by the lead GSI for that project. This will be explained when the project is released. Regrade policies across projects may not be consistent.

Cheating

It's OK to ask someone about the concepts, algorithms, or general approaches needed to do the project assignments. We encourage you to do so; both giving and taking advice helps you to learn. However, what you turn in **must be your own**; copying other people's code, solution sets, Google safaris, or from any other sources is **strictly prohibited**. If you are unsure, then **ask**.

One technique we will use to detect cheating is an automated system that performs comparisons across documents. It flags suspicious similarities, which we then inspect manually.

If elements of two assignments are determined to be clearly very similar (i.e., we believe that they were done together or one was copied from the other), both the copy-ers and the copy-ees will receive a 0 for that assignment. At the end of the semester, the **course grade** for all students involved in the incident will furthermore be **reduced by one letter grade** for the first offense, and **to an F** for the second offense. The reduction in grade may be taken without discussion or warning; the first notice you receive may 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 exams) will engender accordingly more severe penalties.

Again, note that you are responsible for not leaving copies of your assignments lying around and for protecting your files from unauthorized reading.

Last built: 2018-10-28 02:32 UTC