

CMSC 23300/33300  
Networks and Distributed Systems

Department of Computer Science  
University of Chicago

*Winter 2012 Quarter*

**Dates:** January 3 – March 6, 2011

**Lectures:** TuTh 10:30-11:50 in Ryerson 251

**Website:** <http://www.classes.cs.uchicago.edu/archive/2012/winter/23300-1/>

**Lecturer:** Borja Sotomayor

**E-mail:** borja@cs.uchicago.edu

**Office:** Searle 209-A

**Office hours:** Open door policy (see page 5)

**TA:** Soner Balkr

**E-mail:** soner@cs.uchicago.edu

**Office:** Jones 209

**Office hours:** TBD

Contents of this Document

Course description . . . . .	2
Course organization . . . . .	2
Books . . . . .	3
Grading . . . . .	3
Policy on academic honesty . . . . .	5
Asking questions . . . . .	5
Acknowledgements . . . . .	6

---

## Course description

This course focuses on the principles and techniques used in the development of networked and distributed software. Topics include programming with sockets; concurrent programming; data link layer (Ethernet, packet switching, etc.); internet and routing protocols (IP, IPv6, ARP, etc.); end-to-end protocols (UDP, TCP); and other commonly used network protocols and techniques.

In this course, students will learn how to...

- ...implement multithreaded client/server applications using sockets.
- ...interpret existing specifications of network protocols, and translate them into code.
- ...design and combine network protocols that form the foundation of the Internet.
- ...develop software collaboratively through the use of version control tools, code reviews, and project management.

CMSC 15400 and a working knowledge of the C programming language are strict prerequisites of this course. Students who have not taken CMSC 15400 must speak with the instructor to verify that they meet the prerequisites for this course.

---

## Course organization

The development of three programming projects accounts for the majority of the grade in this course. To successfully complete these projects, students must understand fundamental concepts in networking. The class meets two times a week for lectures that provide this conceptual scaffolding, but will also cover material that is not directly applied in the projects. There will be a midterm and a final exam. The course calendar, including the contents of each lecture and project deadlines, is shown in Table 1.

## Projects

Throughout the quarter, students will develop three projects:

1. **IRC Server:** Implementing an Internet Relay Protocol (IRC) server (partially compliant with RFC 2810, 2811, 2812, and 2813) using POSIX sockets and pthreads.
2. **STCP:** Implementing a reliable transport protocol on top of an unreliable one.
3. **Routing:** Implementing an Internet router using Stanford's Virtual Network System.

Each project is independent from the others. Students will develop these projects in pairs. Project groups must be formed by Friday, January 6. Groups can be changed from one project to another, but you must inform the instructor that you intend to do so. If your partner drops out of the course or you feel he/she is not contributing to the group's effort, you should make the instructor aware of this.

## Graduate Project

Graduate students will also have to complete a research-oriented project divided into four stages. In this project, students will have to gather data and perform a series of experiments to empirically test a series of hypothesis (within the realm of Computer Networks). Students will peer-review these papers amongst themselves in a double-blind fashion, will have an opportunity to revise their papers based on the feedback they receive, and will finally present their results in a poster session.

## Exams

There will be a midterm on Tuesday, February 7. This midterm will take place in class, and will only occupy the first 50 minutes of the lecture. The final exam is tentatively scheduled to take place during Finals Week, on March 13 from 10:30 to 12:30.

Questions and exercises related to the projects will make up a substantial part of both exams. Students who have developed the projects on their own (which also requires understanding the material presented in class) should be able to answer these questions with relative ease. However, there will also be a question that are not related to the projects, but will be in line with the learning goals outlined at the beginning of this syllabus.

---

## Books

The *suggested* text for this course is *Computer Networks: A Systems Approach*, 5th edition, L. Peterson and B. Davie, Morgan Kaufmann 2012. Available for purchase from the Seminary Co-op Bookstore.

---

## Grading

For undergraduates, the final grade will be based on the projects (60%, each project worth 20%), midterm (15%), and final exam (25%).

For graduate students, the final grade will be based on the projects (40%, each project worth 13.3%), graduate project (20%), midterm (15%), and final exam (25%).

## Types of grades

Students may take this course for a quality grade (a “letter” grade) or a pass/fail grade. Students will declare on the final exam whether, depending on their final grade, they want to receive a letter grade, a pass/fail grade or withdraw from the course (a *W* grade). For example, students can declare “If my final grade is a C+ or lower, I will take a *P* (Pass) instead of a letter grade and, if my grade is an *F*, I wish to take a *W*”. By default, all students are assumed to be taking the course for a quality grade.

Note: *Students taking this course to meet general education requirements must take the course for a letter grade.*

Table 1: CMSC 23300/33300 Spring 2011 Calendar

Week	Date	Lecture	Book	Project Due	Graduate Project
1	Tu 3 January	Introduction	1		
	Th 5 January	Sockets, Concurrent Programming	—		
2	Tu 10 January	Sockets, Concurrent Programming	—	Project 1a	
	Th 12 January	Link Layer	2, 3		
3	Tu 17 January	Link Layer	2, 3		
	W 18 January				
	Th 19 January	IP	3		
4	M 23 January			Project 1b	
	Tu 24 January	IP	3		
	Th 26 January	TCP/UDP	5		
5	Tu 31 January	TCP/UDP	5		
	Th 2 February	TCP/UDP	5		
	F 3 February				Paper Submission Due
6	M 6 February			Project 1c	
	Tu 7 February	Midterm.	—		
	Th 9 February	Application Layer Protocols	9		
7	Tu 14 February	Routing Protocols, Congestion Control	4, 6		
	W 15 February			Project 2a	
	Thu 16 February	Routing Protocols, Congestion Control	4, 6		
8	M 20 February				Reviews Due
	Tu 21 February	Security	8		
	W 22 February			Project 2b	
	Th 23 February	Security	8		
9	Tu 28 February	Distributed Computing	—		
	Th 1 March	Distributed Computing	—		
10	M 5 March				Revised Paper Due
	Tu 6 March	Other Topics (TBD)	—		Poster Session
	W 7 March			Project 3	

## Late submissions

For the projects, the instructors will collect the latest revision each group commits to their GitHub repository before the deadline. Any work committed after the deadline is ignored and not collected. Each group is allowed four 24-hour extensions during the quarter. More than one extension can be applied to a single submission. i.e., a single 24-hour extension on four submissions, or a 96-hour extension on a single submission. No extensions will be allowed for the graduate project.

If extraordinary circumstances (illness, family emergency, etc.) prevent a student from meeting a deadline, the student must inform the instructor *before* the deadline.

---

## Policy on academic honesty

The University of Chicago has a formal policy on academic honesty that you are expected to adhere to:

<http://studentmanual.uchicago.edu/academic/index.shtml#honesty>

In brief, academic dishonesty (handing in someone else's work as your own, taking existing code and not citing its origin, etc.) will *not* be tolerated in this course. Depending on the severity of the offense, you risk getting a hefty point penalty or being dismissed altogether from the course. All occurrences of academic dishonesty will furthermore be referred to the Dean of Students office, which may impose further penalties, including suspension and expulsion.

Even so, collaboration between students is certainly allowed (and encouraged) *as long as you don't hand someone else's work as your own*. If you have discussed parts of an assignment with someone else, then make sure to say so. If you consulted other sources, please make sure you cite these sources.

If you have any questions regarding what would or would not be considered academic dishonesty in this course, please don't hesitate to ask the instructor.

---

## Asking questions

This course has an *open door policy* for asking questions. Instead of setting fixed office hours, you are welcome to consult with the instructor at any time. Nonetheless, you should try to give the instructor, whenever possible, some advance warning of your visit (by e-mail) to make sure that he will be in the office at that time.

The preferred form of support for this course is through *Piazza* (<http://www.piazza.com/>), an on-line discussion service which can be used to ask questions and share useful information with your classmates. Students will be enrolled in Piazza at the start of the quarter.

All questions regarding the projects or material covered in class must be sent to Piazza, and not directly to the instructor or TA, as this allows your classmates to join in the discussion and benefit from the replies to your question. This rule will be applied strictly: if you send a message directly to the instructor or the TAs, you will only get a reply telling you to send your question to Piazza. The only exception to this rule is if your question requires revealing part of your solution to a project; in that case, please send an e-mail to the following address:

`cmsc23300-instructors@cs.uchicago.edu`

This address reaches both the instructor and the TAs.

---

### **Acknowledgements**

This syllabus is based on previous CMSC 23300/33300 syllabi developed by Prof. Anne Rogers and Prof. Ian Foster from the University of Chicago.