CS 1652: Data Communication and Computer Networks

Syllabus

Course Website

`http://www.cs.pitt.edu/~jacklange/teaching/cs1652-f20`

Instructor

Jack Lange, Associate Professor
Office: Sennott Square, Office #5407
Phone: 412-648-0168
Email: jacklange@cs.pitt.edu
Office Hours: Tues. 2:30-4:30PM

Teaching Assistant

TBA
Office:
Email:
Office Hours: TBA

Location and Time

MW., 1:15 - 2:30PM
Alumni Hall, Room 121
Zoom: Via Canvas

Prerequisites

Required	CS1550
Required	Knowledge of C and C++
Highly Recommended	Unix development experience (gcc, gdb, make, etc)
Recommended	Unix systems programming experience

Textbooks

James Kurose and Keith Ross
Computer Networking: A Top-Down Approach
7th Edition, Pearson, 2016
This will be the primary textbook for the course. Required readings and homework assignments will come from this book.

Richard Stevens
TCP/IP Illustrated, Volume I: The Protocols
Addison Wesley, 1994
This book will be used as a secondary reference source. You will find it extremely useful during the projects.

Recommended books

Larry L. Peterson and Bruce S. Davie
Computer Networks - A Systems Approach
Morgan Kaufmann, 2003
Good supplementary material to the required textbooks

Richard Stevens
Unix Network Programming - Volume 1 (2003) and **Volume 2 (1999)**
Prentice Hall
Describes the details of socket programming and IPC on Unix.

Richard Stevens
Advanced Programming in the Unix Environment
Addison-Wesley, 1992
A basic book for anyone writing programs that run under Unix

Bjarne Strousup
The C++ Programming Language, Special Edition
Addison-Wesley, 2000

Overview

This course introduces the underlying concepts behind networking using the Internet and its protocols as examples. There are three goals: (1) to give you an understanding of how networks, especially the Internet, work, (2) to give you experience with large scale systems, and (3) to teach you network programming.

We will cover the first five chapters of Kurose in detail, working our way down the network stack from the application layer to the data-link layer. Concurrent with the lectures, you (in groups of two) will be building a functional TCP/IP stack and a small web server that will run on it. What you build will be 'real' your code will interoperate with other TCP/IP stacks and you/ll be able to talk to your web server using any browser on any TCP/IP stack.

This class places an equal emphasis on practical experience as well as theoretical foundations. You will interact directly with parts of our Internet infrastructure and implement several core components. It will be a lot of work, but it will also be a lot of fun, provided you enjoy this sort of thing. We will assume that you do and that you will make a good faith effort. After finishing the course, you will be able to do the following.

- Understand the Internet protocols
- Build implementations of the Internet protocols
- Generalize this knowledge to other networking protocols.
- Be a competent network and systems programmer.
- Think like a networking practitioner
- Read and judge articles on networking in trade magazines
- Begin to read and judge research and technical articles on networking
- Create simplicity and reliability out of complexity and unreliability
- Structure and design software systems to achieve that simplicity and reliability

Interaction

The plan for this lectures in this course is for synchronous delivery at the assigned timeslot, with lecture recordsings available to be viewed asynchronously. When/if it becomes appropriate the course will meet in person, otherwise it will operate remotely. Remote participation will be supported via the Zoom web conferencing service.

Office hours will be handled via Zoom as well during the posted times, as well as by appointment. However we will be evaluating the effectiveness of this as the course progresses, and may make changes based to better address the course needs.

It is important to note that while we intend to maintain consistency, everyone should be ready to adapt to a new arrangement if the needs arise. Any changes to lecture delivery will be announced and will be made in accordance with university policies.

Communication

- **Website** - Announcements will be made both in class and via Canvas Please check it regularly for clarifications and corrections. Homework and project materials will also be published there. We are also not able to host some material on Canvas due to limitations, and so a mirror of course material will also be hosted on the course website.

  Web address: `http://www.cs.pitt.edu/~jacklange/teaching/cs1652-f20`

- **Slack Team** - A slack team is setup for the course at "cs1652-f20.slack.com", and an invitation link will be sent to all students enrolled in the class. Slack channels can be used to request help and discuss the course projects with other students in the class. The channel will be loosely monitored by the instructor and TA, urgent questions for the instructor should be sent over email or asked during office hours.

Projects

Over the course of the quarter, you will implement a user-level TCP/IP stack and a small web server that runs on top of it. Your code will not implement the full functionality of HTTP or TCP/IP, but it will implement enough of it to be able to interoperate with other, complete implementations. In keeping with the top-down approach of Kurose and Ross, you will build this from the web server down instead of from the network card up. I will initially provide you with the whole stack (as object code) and you will implement the web server. Next, we will move lower down the stack, leaving you to implement your own version of TCP. Each layer will have well-defined interfaces that you will fill out.

35% Web server (Sockets)
65% TCP

To evaluate your project, we will spot-check your source code, compile it, and run randomized testcases on it. When appropriate, we will supply you with examples of such testcases.

Homework

Each lecture covers material from the textbook which you will be responsible for reading before you come to class.

There will also be four homework sets that will be periodically assigned to help you improve your understanding of the material.

Late Policy

Unless otherwise indicated, homework and projects are due by midnight on their due date. If you submit an assignment late, a 10% penalty will be accrued for each day it is late.

Exams

There will be a midterm exam and a final exam. The final exam will not be cumulative. ¡

Grading

40% Project
25% Midterm
25% Final
10% Homework

Cheating

Since cheaters are mostly hurting themselves, we do not have the time or energy to hunt them down. We much prefer that you act collegially and help each other to learn the material and to solve development problems than to have you live in fear of our wrath and not talk to each other. Nonetheless, if we detect blatant cheating, we will deal with the cheaters as per university guidelines.

Schedule (Tentative)

Date	Lecture	Reading	HW/Proj
8/19	Class overview and introduction	1.1 - 1.3	HW 0 out
8/24	Introduction cont'd	1.4 - 1.7	
8/26	Web, HTTP and FTP	2.1 - 2.3	HW 0 in, HW 1 out
8/31	Email, DNS, and P2P	2.4 - 2.6	
9/2	Sockets	2.7 - 2.8	Proj 1 out
9/7	Transport intro, multi/demultiplexing, UDP	3.1 - 3.3	
9/9	Reliable transport principles	3.4	
9/14	Connection oriented transport: TCP	3.5	
9/16	Lab Session		HW 1 in, HW 2 out
9/21	Congestion Control	3.6 - 3.7	
9/23	TCP Alternatives		
9/28	Midterm review		Proj 1 in, Proj 2 out
9/30	Midterm		
10/5	Midterm grading review		
10/7	Network Layer introduction	4.1 - 4.3	HW 2 in, HW 3 out
10/12	Network Layer: IP and ICMP	4.4	
10/14	No Class (Labor Day)		
10/19	Lab session		
10/21	Routing algorithms	4.5	
10/26	Hierarchical routing, RIP, OSPF, BGP	4.6 - 4.7	
10/28	Data link layer, error detection and correction multiple access protocols	5.1 - 5.3	HW 3 in, HW 4 out
11/2	Link layer addressing, Ethernet, hubs and switches	5.4 - 5.6	
11/4	Lab session		
11/9	Wireless and mobile networks (wireless links, 802.11)	6.1 - 6.3	
11/11	IPv6 and related protocols	7.1 - 7.8	
11/16	Advanced Topics	8	Proj 2 in
11/18	Final Review		HW 4 in