Overview for today

- Class overview
  - Administrativa (who, what, where)
  - Expected outcomes
  - Structure of the course
  - Policies and procedures

- A brief overview of Computer Networking
  - High-level concepts
  - An end-to-end example
Course instructors

- **Stefan Savage** – Lecturer & taskmaster
  - Web: [http://www.cs.ucsd.edu/~savage](http://www.cs.ucsd.edu/~savage)
  - E-mail: savage@cs.ucsd.edu
  - Office hours: Tu 4-5pm (or by appt) CSE 3106

- **Alex Rasmussen** – TA
  - E-mail: arasmuss@cs.ucsd.edu
  - Office hours: TBA

- **Nima Nikzad** – TA
  - E-mail: nnikzad@cs.ucsd.edu
  - Office hours: TBA
About me

- I work at the intersection of networking, operating systems and computer security

- Research
  - Large-scale network measurement projects
    » Routing behavior, WiFi performance, measurement tools
  - Large-scale Internet attacks (worms/virus, bots, spam) and e-crime economics

- Policy
  - National Science Foundation (CISE)
  - NRC’s Computer Science & Telecommunications Board
  - ISAT advisory group for DARPA

- Industry
  - Asta Networks (defunct anti-DDoS company)
  - Netsift (UCSD-originated net company) -> Cisco
  - Lots of consulting
Course info

- Discussion section: M 1-1:50  Center 109
- There will be a discussion board (TBA)
- Course Web page
  - http://www.cse.ucsd.edu/classes/fa11/cse123-a/
Alert!

- No discussion section this Monday
- Tuesdays class is cancelled

- **We next meet on Thursday the 29th**
Expected Outcomes

- This course *will* teach you the *fundamentals* of computer networks:
  - Layering, signaling, framing, MAC, switching, routing, naming, Internetworking, congestion control, router design, etc.

- I *will not* teach you much about signals and coding
  - Take an EE course to learn about modulation, encoding, etc. on different hardware technologies

- Similarly, we will not cover Internet apps/services
  - CSE124 covers application layer protocols, Web, etc.
  - You can also pick up much of this on your own
Prereqs

- CSE120
  - I will approve enrollment for students who have not taken it,
  - But, you will be solely responsible for concepts and experience from the class (e.g., concurrency)

- Programming experience
  - We will be assigning programming projects in C/C++
  - This course will not teach you C. The TAs will help, but you need to learn it on your own if you don’t already know it.
CSE 123 Class Overview

- Course material taught through class lectures, textbook readings, and discussion sections
- Course assignments are
  - Homework questions (based on lecture and reading)
  - 3 programming projects (two significant)
- Discussion sections are a forum for asking questions
  - Lecture material and homework
  - Additional networking topics
- Discussion board (TBA)
  - The place to ask questions about lecture, hw, projects, etc.
Rules

- Written assignments are due at the *beginning* of class
- Regrades should be the *exception*
  - Addition errors (happy), significant errors in grading (fine), nit picking/grade mongering (death to you)
  - We reserve the right to *completely* regrade your assignments
  - All regrades go to TAs first

- **No Cheating**

- Cheating means not doing the assignment yourself
  - Ok to *talk* with other students about assignments outside of class
  - No copying, no Google, etc. If you’re unsure, then ask

- Don’t mess with the professor. He’s a mean man.
Labs

- We will use the uAPE (B230) lab in the basement of the CSE/EBU3B building
  - Linux running on Intel machines

- You can also use your home machine
  - But…. we will test on uAPE machines
  - Be sure to test your projects there as well
Exams

- **Midterm**
  - Tentatively Tuesday, November 1st
  - Covers first half of class

- **Final**
  - Friday Dec 9\textsuperscript{th}, 11:30-2:30
  - Covers second half of class + selected material from first part
    - I will be explicit about the material covered

- **No makeup exams**
  - Unless dire circumstances

- **All tests closed book with crib sheet**
  - You can bring one double-sided 8.5x11” page of notes to each exam to assist you in answering the questions
  - Not a substitute for thinking
Grading

- Homeworks: 20%
  - Think of these collectively as a take-home midterm

- Midterm: 20%

- Final: 30%

- Projects: 30%
How *Not* To Pass CSE 123

- **Do not come to lecture**
  - It’s nice out, class is early, the slides are online, and the material is in the book anyway
  - Lecture material is the basis for exams and directly relates to the projects
  - I guarantee you I’m more fun than the textbook

- **Do not do the homework**
  - It’s only 20% of the grade
  - Excellent practice for the exams, and some homework problems are exercises for helping with the project
  - 20% is actually a significant fraction of your grade (difference between an A and a C)
How *Not* To Pass (2)

- Do not ask questions in lecture, office hours, or email
  - Professor is scary, I don’t want to embarrass myself
  - Asking questions is the best way to clarify lecture material at the time it is being presented
  - Office hours and email will help with homeworks, projects

- Wait until the last couple of days to start a project
  - We’ll have to do the crunch anyways, why do it early?
  - The projects cannot be done in the last couple of days
Class Web Page

http://www.cs.ucsd.edu/classes/fall/cse123-a/

Will serves many roles…

- Course syllabus and schedule (updated as quarter progresses)
  » Lecture slides
- Announcements
- Homework handouts
- Project information
Questions

- Before we start the material, any questions about the class structure, contents, etc.?
A “Simple” Task

- Send information from one computer to another

- Endpoints are called **hosts**
  - Could be computer, iPod, cell phone, etc.

- The plumbing is called a **link**
  - We don’t care what the physical technology is: Ethernet, wireless, cellular, etc.
What if hosts aren’t directly connected?

- **Routers/Switches**: moves bits between links
  - *Circuit switching*: guaranteed channel for a session (Telephone system)
  - *Packet switching*: statistical multiplexing of independent pieces of data (Internet)
Measures of success

- How fast?
  - Bandwidth measured in bits per second
  - Often talk about KBps or Mbps – Bytes vs bits

- How long was the wait?
  - Delay (one-way or round trip) measured in seconds (typically milliseconds)

- How efficiently?
  - Overhead measured in bits or seconds or cycles or…

- Any mistakes?
  - Error rate measured in terms of probability of a flipped bit (or corrupted packet)
How long to send a message?

- Transmit time $T = \frac{M}{R} + D$
  - $M =$ message size, $R =$ bandwidth, $D =$ delay
  - 10 Mbps Ethernet LAN ($M = 1$KB)
    » $\frac{M}{R} = 1$ms, $D \approx 5$us
  - 155 Mbps cross country ATM link ($M = 1$KB)
    » $\frac{M}{R} = 50$us, $D \approx 40-100$ms

- Where are the bits in the mean time?
  - In transit inside the network

- R*D is called the **bandwidth delay product**
  - How many bits can be “stored” be stored in transit
  - Colloquially, we say “fill the pipe”
But there’s more to networking than sending bits…

- Like what?
- Sending bits to a *particular* destination among many
- Sending a **long** message to a particular destination
- Detecting if there was an error
- Fixing the error
- Deciding how fast to send
- Making sure the message is kept private
- Etc, etc, etc…
Layering: A Modular Approach

- **Sub-divide the problem**
  - Break up functionality into distinct services (e.g., reliable message delivery)
  - Organized services into order series of **layers**
  - Each layer relies on services from layer below
  - Each layer exports services to layer above

- **Interface between layers defines interaction**
  - Hides implementation details
  - Layers can change without disturbing other layers

- **Interface among peers in a layer is a protocol**
  - If peers speak same protocol, they can interoperate
TCP/IP Protocol Stack

Application Layer

Transport Layer

Network Layer

Link Layer

Physical Layer (encoding on wires, light, radio, etc)
Encapsulation via packet headers

TCP

IP

Ethernet interface

Headers

Payload

HTTP

CSE 123 – Lecture 1: Course Introduction
Putting this all together

- **ROUGHLY**, what happens when I click on a Web page from UCSD?

My computer → ? → Internet → www.google.com
Web request (HTTP)

- Turn click into HTTP request

GET http://www.google.com/ HTTP/1.1
Host: www.google.com
Connection:keep-alive
...
Name resolution (DNS)

- Where is www.google.com?

My computer (132.239.9.64) → Local DNS server (132.239.51.18)

What’s the address for www.google.com

Oh, you can find it at 66.102.7.104
Data transport (TCP)

- Break message into packets (TCP segments)
- Should be delivered reliably & in-order

```
GET http://www.google.com HTTP/1.1
Host: www.google.com
Connection: keep-alive
...
```
Network Layer: Global Addressing

- Address each packet so it can traverse network and arrive at host

My computer (132.239.9.64)

www.google.com (66.102.7.104)
Network layer: packet routing

- Each router forwards packet towards destination
Link Layer (e.g. Ethernet)

- Break larger network message into individual frames
- Media Access Control (MAC)
  - Can I send now?  Can I send now?
- Send frame
Physical layer

2.4Ghz Radio
DS/FH Radio
(1-11Mbps)

802.11b Wireless Access Point

Cat5 Cable (4 wires)
100Base TX Ethernet
100Mbps

Ethernet switch/router

To campus backbone

62.5/125um 850nm MMF
1000BaseSX Ethernet
1000Mbps
For Next Class…

- Reminder
  - No discussion section this Monday
  - Tuesdays class is cancelled
  - *We next meet on Thursday the 29th*

- Read Chapter 1 (excepting 1.4)
- Go bookmark the Web page
- Drop now or plan to stick it out!

- Have a great weekend!