Teaching staff

- **Instructors**
  - Prof. Randal E. Bryant
  - Prof. David R. O'Hallaron

- **TA's**
  - Ashwin Bharambe
  - Michael Brotzman
  - Tudor Dumitras
  - Donnie Kim
  - Amit Manjhi

- **Course Admins**
  - Cindy Chemsak (NSH 4303)
  - Barb Grandillo (WeH 8018)

Come talk to us anytime! (Or phone or send email)

Textbooks

**Randal E. Bryant and David R. O'Hallaron,**
- [http://csapp.cs.cmu.edu](http://csapp.cs.cmu.edu)

**Brian Kernighan and Dennis Ritchie,**

Course Components

**Lectures**
- Higher level concepts

**Recitations**
- Applied concepts, important tools and skills for labs, clarification of lectures, exam coverage

**Labs**
- The heart of the course
- 1 or 2 weeks
- Provide in-depth understanding of an aspect of systems
- Programming and measurement
Getting Help

Class Web Page
- http://www.cs.cmu.edu/~213
- Copies of lectures, assignments, exams, solutions
- Clarifications to assignments

Message Board
- http://autolab.cs.cmu.edu
- Clarifications to assignments, general discussion
- The only board your instructors will be monitoring (No blackboard or Andrew)

Personal help
- Professors:
  - R. Bryant, office hour or appt.
  - D. O'Hallaron, office hour, appt, or when door is open.
- TAs: office hour, email, or appt.

Policies: Assignments

Work groups
- You must work alone on all labs

Handins
- Assignments due at 11:59pm on Wed or Fri evening
- Electronic handins using Autolab (no exceptions!).

Makeup exams and assignments
- OK, but must make PRIOR arrangements with Prof. O'Hallaron.

Appealing grades
- Within 7 days of due date or exam date.
- Labs: Talk to the lead person on the assignment
- Exams: Talk to Prof O'Hallaron.

Cheating

What is cheating?
- Sharing code: either by copying, retyping, looking at, or supplying a copy of a file.
- Coaching: helping your friend to write a lab, line by line.

What is NOT cheating?
- Helping others use systems or tools.
- Helping others with high-level design issues.
- Helping others debug their code.

Penalty for cheating:
- Removal from course with failing grade.

Detection of cheating:
- We do check and our tools for doing this are much better than you think!

Policies: Grading

Exams (40%)
- Two in class exams (10% each)
- Final (20%)
- All exams are open book / open notes.

Labs (60%)
- 7 labs (6-12% each)

Grading Characteristics
- Lab scores tend to be high
  - Serious handicap if you don’t hand a lab in
  - We offer generous redemption programs
- Tests typically have a wider range of scores
Facilities

- Labs will use the Intel Computer Systems Cluster (aka “the fish machines”)
  - 15 Pentium Xeon servers donated by Intel for CS 213
  - Dual 3.2 Ghz 64-bit (EM64T) Nocona Xeon processors
  - 2 GB, 400 MHz DDR2 SDRAM memory
  - Rack mounted in the 3rd floor Wean Hall machine room.
- Your accounts are ready.

Getting help with the cluster machines:
- See course Web page for login directions
- Please direct questions to your TA’s first

Logging into Fish Machines

- Read description on the course web-page carefully
- Run checkin script (once only) to setup Kerberos credentials
  - `% /afs/cs/academic/class/15213-f06/bin/checkin`
- Login using your Andrew ID and password:
  - `% ssh –x –l bovik@ANDREW.CMU.EDU tuna.ics.cs.cmu.edu`
- Keep your code in your “213hw” directory on your Andrew account

Programs and Data (6)

- Topics
  - Bits operations, arithmetic, assembly language programs, representation of C control and data structures
  - Includes aspects of architecture and compilers
- Assignments
  - L1 (datalab): Manipulating bits
  - L2 (bomlab): Defusing a binary bomb
  - L3 (buflab): Hacking a buffer bomb

Performance (2)

- Topics
  - High level processor models, code optimization (control and data), measuring time on a computer
  - Includes aspects of architecture, compilers, and OS
- Assignments
  - L4 (perflab): Optimizing code performance
The Memory Hierarchy (2)

**Topics**
- Memory technology, memory hierarchy, caches, disks, locality
- Includes aspects of architecture and OS.

**Assignments**
- L4 (perflab): Optimizing code performance

Linking and Exceptional Control Flow (3)

**Topics**
- Object files, static and dynamic linking, libraries, loading
- Hardware exceptions, processes, process control, Unix signals, nonlocal jumps
- Includes aspects of compilers, OS, and architecture

**Assignments**
- L5 (tshlab): Writing your own shell with job control

Virtual Memory (4)

**Topics**
- Virtual memory, address translation, dynamic storage allocation
- Includes aspects of architecture and OS

**Assignments**
- L6 (malloclab): Writing your own malloc package

I/O, Networking, and Concurrency (6)

**Topics**
- High level and low-level I/O, network programming, Internet services, Web servers
- concurrency, concurrent server design, threads, I/O multiplexing with select.
- Includes aspects of networking, OS, and architecture.

**Assignments**
- L7 (proxylab): Writing your own Web proxy
Lab Rationale

Each lab should have a well-defined goal such as solving a puzzle or winning a contest.

Doing a lab should result in new skills and concepts.

We try to use competition in a fun and healthy way.

- Set a reasonable threshold for full credit.
- Post intermediate results (anonymized) on Web page for glory!

Autolab Web Service

Labs are provided by the Autolab system

- Autograding handin system developed in 2003 by Dave O’Hallaron
- Apache Web server + Perl CGI programs
- Beta tested Fall 2003, very stable by now

With Autolab you can use your Web browser to:

- Review lab notes, clarifications
- Download the lab materials
- Stream autoresults to a class status Web page as you work.
- Handin your code for autograding by the Autolab server.
- View the complete history of your code handins, autoresult submissions, autograding reports, and instructor evaluations.
- View the class status page

Good Luck!