15-213

“The Class That Gives CMU Its Zip!”

Introduction to Computer Systems

David O’Hallaron
August 26, 2003

Topics:

- Staff, text, and policies
- Lecture topics and assignments
- Lab rationale and infrastructure

Teaching staff

- Instructors
  - Prof. David O’Hallaron
  - Prof. Andreas Nowatzyk

- TA’s
  - Quarup Barreirinhaus
  - Stavros Harizopoulos
  - Matt Rosencrantz
  - Nitin Sharma
  - Joseph Trdinich
  - Timothy Wong

- Course Admin
  - Barb Grandillo (WeH 4218)

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

Textbooks

Randal E. Bryant and David R. O’Hallaron,

- “Computer Systems: A Programmer’s Perspective”, Prentice Hall 2003.
- csapp.cs.cmu.edu

Brian Kernighan and Dennis Ritchie,

- “The C Programming Language, Second Edition”, Prentice Hall, 1988

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

Web

- www.cs.cmu.edu/afs/cs/academic/class/15213-f02/www
- Copies of lectures, assignments, exams, solutions
- Clarifications to assignments

Newsgroup

- cmu.cs.class.cs213
- Clarifications to assignments, general discussion

Personal help

- Professors: door open means come on in (no appt necessary)
- TAs: please mail or zephyr first.

Policies: Assignments

Work groups

- You must work alone on all labs

Handins

- Assignments due at 11:59pm on specified due date.
- Typically 11:59pm Thursday evening (to avoid 212 conflicts)
- Electronic handins only.

Makeup exams and assignments

- OK, but must make PRIOR arrangements with either Prof. O'Hallaron or Nowatzyk.

Appealing grades

- Within 7 days of due date or exam date.
- Assignments: Talk to the lead person on the assignment
- Exams: Talk to either Prof. O'Hallaron or Nowatzyk.

Cheating

What is cheating?

- Sharing code: either by copying, retyping, looking at, or supplying a copy of a file.

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.

Policies: Grading

Exams (40%)

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

Labs (60%)

- 7 labs (8-12% 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

Assignments will use the Intel Computer Systems Cluster (aka “the fish machines”)

- 25 Pentium III Xeon servers donated by Intel for CS 213
- 550 MHz with 256 MB memory.
- Rack mounted in the 3rd floor Wean Hall machine room.
- We’ll be setting up your accounts this week.

Getting help with the cluster machines:

- See course Web page for info
- Please direct questions to your TAs

Programs and Data (8)

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 (bomblab): 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 (perlab): 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

- Data Lab: number representations, logic, bit manipulation.
- Bomb Lab: assembly, using debugger, understanding stack
- Perf Lab: profiling, measurement, performance debugging.
- Shell Lab: understanding Unix process control and signals
- Malloc Lab: understanding pointers and nasty memory bugs.
- Proxy Lab: network programming, server design

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

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

Autolab Web Service

Labs are provided by the Autolab system

- Developed in summer 2003 by Dave O'Hallaron
- Apache Web server + Perl CGI programs
- Beta testing Fall 2003

With Autolab you can use your Web browser to:

- Review lab notes
- Download the lab materials
- Stream autoresults to a class status Web page as you work.
- Upload (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!