15-213
"The Class That Gives CMU Its Zip!"

Introduction to Computer Systems

David O'Hallaron
August 27, 2002

Topics:

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

Teaching staff

- **Instructors**
  - **Prof. Randy Bryant (Wed 10:00-11:00, WeH 4220)**
  - **Prof. David O'Hallaron (Tue 10:30-11:30, WeH 8125)**

- **TA's**
  - **Rajesh Balan (TBD, WeH 8205)**
  - **Shimin Chen (TBD, WeH 8019)**
  - **Andrew Faulring (TBD, NSH 2504)**
  - **Anubhav Gupta (TBD, WeH 8218)**
  - **Annie Luo (TBD, WeH 8402)**

- **Course Admin**
  - **Rosemary Battenfelder (WeH 4218)**

These are the nominal office hours. 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**
  - (except for Lab 7: Web Proxy, with work groups of two)

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. Bryant or O'Hallaron.**

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. Bryant or O'Hallaron.**

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 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 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 of architecture and compilers**

Assignments

- **L1: Manipulating bits**
- **L2: Defusing a binary bomb**
- **L3: Hacking a buffer bomb**

Performance (3)

Topics

- High level processor models, code optimization (control and data), measuring time on a computer
- Includes aspects of architecture, compilers, and OS

Assignments

- L4: Optimizing Code Performance

The Memory Hierarchy (2)

Topics

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

Assignments

- L4: 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: 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: 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: Writing your own Web proxy**

Lab Rationale

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

- Defusing a binary bomb.
- Winning a performance contest.

Doing a lab should result in new skills and concepts

- Data Lab: computer arithmetic, digital logic.
- Bomb Labs: assembly language, using a debugger, understanding the 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!

Good Luck!