CS 213
Introduction to Computer Systems

Course Organization

David O'Hallaron

August 25, 1998

Topics:

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

Teaching staff

Instructors

- Prof. Randy Bryant (Tue 10:30-11:30, WeH 7128)
- Prof. David O'Hallaron (Tue 10:30-11:30, WeH 8125)

TA's

- Chris Colohan (Wed 3-4, WeH 5101)
- Larry Greenfield (Wed 12-1, WeH 3130)
- Kip Walker (Wed 2-3, WeH 8218)

Course secretary

- Joan Maddamma (WeH 7121 Wean)

These are the nominal office hours. Come talk to us anytime!
(Or send email)

Textbook

Brian W. Kernighan and Dennis M. Ritchie,

- ***The C Programming Language,***
- ***Second Edition***
- **Prentice Hall, 1988**

Classic K&R book.

Partial coverage of course material.

Remainder will be provided in notes and handouts.

Course Components

Lectures

- Higher level concepts

Recitations

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

Labs

- Multi-week (2 or 3)
- Provide in-depth understanding of an aspect of systems
- Programming and measurement

Homeworks

- 1 week
- Solving a series of smaller problems
- Some programming

Getting Help

Web

- www.cs.cmu.edu/afs/cs/academic/class/15213-f98/www
- Copies of lectures, assignments, exams, solutions
- Clarifications to assignments
- Summaries of performance on exams and 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 may do all labs and homeworks in groups of up to 2 .

Handins

- Basically something due every Thursday
- Assignments due at 12:01am on specified due date.
- Electronic handins only.

Makeup exams and assignments

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

Appealing grades

- Within 7 days of due date.
- Assignments: Talk to lead TA first, then Prof. O'Hallaron
- Exams: Talk to Prof. O'Hallaron

Policies: Grading

Exams (50%)

- **Two in class exams (12.5% each)**
- **Final (25%)**
- **All exams are open book/open notes.**

Assignments (50%)

- **4 homeworks (~1 week, 3-5% each)**
- **4 labs (~2 weeks, 8-12% each)**

Grading Characteristics

- **Assignment scores tend to be high**
  - Serious handicap if you don't hand one in
- **Tests have big bearing on letter grade**
  - Wider range of scores
  - Only chance for us to evaluate individual performance

Facilities

Assignments will use “Colour Machines”

- 20 Digital Unix systems (black.ece, white.ece, …)
- 433 MHz Alpha 21164 processor with 128 MB memory.
- Fast machines with simple and regular assembly language.

Part 1: Programs (12)

Topics

- Bits operations, arithmetic, assembly language programs, representation of C control and data structures, object files, processes, asynchronous processing
- Includes aspects of of architecture, OS, and compilers

Assignments

- H1: Integer arithmetic
- H2: Human decompiler
- L1: “Defusing a Binary Bomb”

- H3: IEEE FP conversion

Part 2: Memory (6)

Topics

- **Memory management, memory technology, memory hierarchy, address translation**
- **Includes aspects of architecture and OS.**

Assignments

- **L2: “Malloc Contest” and “Unpeeling an Onion”**

- **H4: Address translation**

Part 3: Performance (5)

Topics

- Code optimization (control and data), performance evaluation, benchmarking
- Includes aspects of architecture and compilers

Assignments

- L3: “Hashing Contest”

Part 4: Networking (3)

Topics

- Network technology, protocol stacks, TCP/IP, routing, sockets
- Includes aspects of networking and architecture

Assignments

- L4: “Feeding a Hungry Cookie Monster”

Lab Rationale

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

- Defusing a binary bomb.
- Unpeeling an onion.
- Winning a performance contest.
- Feeding a cookie monster.

Doing a lab should result in new skills and concepts

- Bomb: assembly language, using a debugger.
- Onion: general strategies for nasty memory bugs.
- Hash: profiling, measurement, performance debugging.
- Cookie: packet monitors, client/server computing.

Reverse engineering is a recurring theme

- And a key job skill!!!