Course Logistics
15-213 (18-213): Introduction to Computer Systems
1<sup>st</sup> Lecture, Aug. 26, 2014

Instructors:
Greg Ganger, Greg Kesden, and Dave O’Hallaron

The course that gives CMU its “Zip”!
Textbooks

- Randal E. Bryant and David R. O’Hallaron,
  - [http://csapp.cs.cmu.edu](http://csapp.cs.cmu.edu)
  - This book really matters for the course!
    - How to solve labs
    - Practice problems typical of exam problems

- Brian Kernighan and Dennis Ritchie,
  - Still the best book about C, from the originators
Course Components

■ Lectures
  ▪ Higher level concepts

■ Recitations
  ▪ Applied concepts, important tools and skills for labs, clarification of lectures, exam coverage

■ Labs (7)
  ▪ The heart of the course
  ▪ 1-2 weeks each
  ▪ Provide in-depth understanding of an aspect of systems
  ▪ Programming and measurement

■ Exams (midterm + final)
  ▪ Test your understanding of concepts & mathematical principles
Getting Help

- Class Web page: [http://www.cs.cmu.edu/~213](http://www.cs.cmu.edu/~213)
  - Complete schedule of lectures, exams, and assignments
  - Copies of lectures, assignments, exams, solutions
  - Clarifications to assignments

- Blackboard and Piazza
  - We won’t be using Blackboard or Piazza for the course
Getting Help

- Staff mailing list: 15-213-staff@cs.cmu.edu
  - Use this for all communication with the teaching staff
  - Always CC staff mailing list during email exchanges
  - Send email to individual instructors only to schedule appointments

- Office hours (starting Tue Sept 2):
  - SMTWR, 5:30-8:30pm, WeH 5207

- 1:1 Appointments
  - You can schedule 1:1 appointments with any of the teaching staff
Policies: Labs And Exams

■ Work groups
  ▪ You must work alone on all lab assignments

■ Handins
  ▪ Labs due at 11:59pm on Tues or Thurs
  ▪ Electronic handins using Autolab (no exceptions!)

■ Exams
  ▪ Exams will be online in network-isolated clusters
  ▪ Held over multiple days. Sign up for a slot

■ Appealing grades
  ▪ In writing to Prof O’Hallaron within 7 days of completion of grading
  ▪ Follow formal procedure described in syllabus
Facilities

- Labs will use the Intel Linux Computer Systems Cluster
  - AKA the “shark machines”
  - Login to the any shark machine:
    - `linux> ssh -X shark.ics.cs.cmu.edu`
  - Login to a particular machine:
    - `linux> ssh -X angelshark.ics.cs.cmu.edu`
  - Login using your Andrew credentials

- List of machines at
  - `http://www.cs.cmu.edu/~213/labmachines.html`
Shark Machines

- 21 servers donated by Intel for 213
  - 10 student machines (for student logins)
  - 1 head node (for Autolab server and instructor logins)
  - 10 grading machines (for autograding)

- Each server:
  - Core i7 system with 8 Nehalem cores, 32 GB DRAM, Linux
  - Rack mounted in Gates machine room

- Getting help with the cluster machines:
  - Please direct questions to staff mailing list
Timeliness

- **Grace days**
  - 5 grace days for the semester
  - Limit of 2 grace days per lab used automatically
  - Covers scheduling crunch, out-of-town trips, illnesses, minor setbacks
  - Save them until late in the term!

- **Lateness penalties**
  - Once grace day(s) used up, get penalized 15% per day
  - No handins later than 3 days after due date

- **Catastrophic events**
  - Major illness, death in family, ...
  - Formulate a plan (with your academic advisor) to get back on track

- **Advice**
  - Once you start running late, it’s really hard to catch up
Cheating: Description

Please pay close attention, especially if this is your first semester at CMU

What is cheating?

- Sharing code: by copying, retyping, looking at, or supplying a file
- Describing: Verbal description of code from one person to another.
- Coaching: helping your friend to write a lab, line by line
- Copying code from a previous course or online solution
  - Only allowed to use code we supply, or from CS:APP website

What is NOT cheating?

- Explaining how to use systems or tools
- Helping others with high-level design issues

See the course syllabus for details.
Cheating: Consequences

- Penalty for cheating:
  - Removal from course with failing grade (no exceptions!)
  - Permanent mark on your record
  - Your instructors’ contempt

- Detection of cheating:
  - Last Fall, 17 students were caught cheating and failed the course.

- Don’t do it!
  - Start early
  - Ask the staff for help when you get stuck
Other Rules of the Lecture Hall

- Laptops: permitted

- Electronic communications: forbidden
  - No email, instant messaging, cell phone calls, etc

- Presence in lectures, recitations: voluntary, recommended

- No recordings of ANY KIND
Policies: Grading

- Exams (50%): midterm (20%), final (30%)

- Labs (50%): weighted according to effort

- Final grades based on a combination of straight scale and possibly a tiny amount of curving.
Programs and Data

- **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
The Memory Hierarchy

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

- Assignments
    - Learn how to exploit locality in your programs.
Exceptional Control Flow

Topics
- Hardware exceptions, processes, process control, Unix signals, nonlocal jumps
- Includes aspects of compilers, OS, and architecture

Assignments
- L5 (tshlab): Writing your own Unix shell.
  - A first introduction to concurrency
Virtual Memory

■ Topics
  ▪ Virtual memory, address translation, dynamic storage allocation
  ▪ Includes aspects of architecture and OS

■ Assignments
  ▪ L6 (malloclab): Writing your own malloc package
    ▪ Get a real feel for systems-level programming
Networking, and Concurrency

■ 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
    ▪ Learn network programming and more about concurrency and synchronization.
Lab Rationale

- Each lab has a well-defined goal such as solving a puzzle or winning a contest

- Doing the 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 Autolab scoreboard for glory!
Autolab (https://autolab.cs.cmu.edu)

- Labs are provided by the CMU Autolab system
  - Project page: http://autolab.cs.cmu.edu
  - Developed by CMU faculty and students
  - Key ideas: Autograding and Scoreboards
    - **Autograding**: Using VMs on-demand to evaluate untrusted code.
    - **Scoreboards**: Real-time, rank-ordered, and anonymous summary.
  - Used by over 2,500 CMU students each semester, since Fall, 2010

- With Autolab you can use your Web browser to:
  - Download the lab materials
  - Handin your code for autograding by the Autolab server
  - View the class scoreboard
  - View the complete history of your code handins, autograded results, instructor’s evaluations, and gradebook.
  - View the TA annotations of your code for Style points.
Autolab accounts

- Students enrolled 10am on Mon, Aug 25 have Autolab accounts

- You must be enrolled to get an account
  - Autolab is not tied in to the Hub’s rosters
  - If you add in, contact 15-213-staff@cs.cmu.edu for an account

- For those who are waiting to add in, the first lab (datalab) will be available on the Schedule page of the course Web site.
Waitlist or enrollment questions

- 15-213: Catherine Fichtner (cathyf@cs.cmu.edu)
- 18-213: Chelsea Mastilak (cmastila@andrew.cmu.edu)

Please don’t contact the instructors with waitlist questions.
Welcome and Enjoy!