

Computer Organization Fall 2001

Dave Hollinger

<http://www.cs.rpi.edu/~hollingd/comporg>

comporg@cs.rpi.edu

Computer Organization

- High-level study of hardware.
- Focus on relationship between hardware design and
 - performance
 - cost

Topics

Overview of Operating Systems

- Unix
- Handling multiple processes
- Making use of Operating System Services
 - C Programming for Unix

Topics

- Performance
 - parameters, measurements, estimates
- Logic Design
- Instruction Sets
 - Assembly Language Programming

Topics

- Computer Arithmetic
- Datapath and Control
 - Building an instruction interpreter in hardware
- Pipelining
 - Overlapping instructions
- Memory Organization
 - Cache design

Topics (if we have time...)

- Buses
 - Peripherals
- Multiprocessing
 - Multiple processors
- Case Studies (organization of real processors).



- Very widely used!
- Each Chapter Includes:
 - Fallacies and Pitfalls
 - Key terms

Grading

- 2 Tests (20% each).
 - Oct 1st Nov 19th
- Homework (30% total)
 - Some small assignments
 - Some medium-large programming projects
- Final Exam (50%)
 - Last few weeks in depth, and put it all together

-
- All problems must be addressed within one week of receiving grades.
 - No grades older than one week will be changed!
 - Test grades and some homeworks will be returned in class. Some homework grades will be sent via email.

Homework

- Must be done individually unless assigned as a group project.
Discussion is great, but no sharing of answers/code.
- My hobby is catching cheaters – check out my web site:
– www.catchingandfailingcheaters.com

Homeworks

- Lots! This will force you to stay caught up with the course (thanks Dave!).
- It will not be possible to be very flexible with deadlines.
– extensions are possible, but not at the last minute! Ask ahead of time (at least 6 months before project due date).

Possible Homework Assignments

- C Programming (interface with Unix kernel via system calls).
- Logic Design project.
- Assembly Language Programming.
- Corewars
- Datapath and Control Design.
- Develop benchmarks.
- Write programs to make best/worst use of pipeline and memory hierarchy.

Getting Help

- email: comporg@cs.rpi.edu
 - 3 people will ~~ignore~~ see it!
 - At least one will respond soon!
- FAQs: questions and answers for each project will be available on the course home page.

How to do well

- If you don't think a topic is "relevant", say so (at a minimum you will feel better for speaking your mind).
- Dave thinks he is funny – laugh at his jokes.
- Put all personal relationships on hold for the semester.
- Get a large coffeemaker.
- Bring a TA a cookie now and then.
