

# 16.482 / 16.561: Computer Architecture and Design

Fall 2014

## Syllabus

### Course Meetings

Section 201: Th 6:30-9:20, Ball 314

### Course Website

Main site: <http://mgeiger.eng.uml.edu/compArch/f14>

Schedule: <http://mgeiger.eng.uml.edu/compArch/f14/schedule.htm>

### Course Discussion Group

All course announcements will be posted on the discussion group—you are responsible for checking the board regularly or enabling direct e-mail updates from Piazza.

Sign up link: <https://piazza.com/uml/fall2014/1648216561>

### Instructor

Dr. Michael Geiger

E-mail: Michael\_Geiger@uml.edu

Office: Perry Hall 118A

Phone: 978-934-3618 (x43618 on campus)

Office hours: Monday 1-2:30, Wednesday 1-2:30, Thursday 3-4:30

During the above hours, student questions are my top priority. I will also be in my office MWF 9-11:45 and Th 4:30-6:15, and I am available by appointment at other times.

Feel free to stop by my office, e-mail me questions, or schedule a one-on-one appointment. Office hours are subject to change.

### Textbook

David A. Patterson and John L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, 5<sup>th</sup> edition, 2013, ISBN: 9780124077263

### Course Overview

Description: Structure of computers, past and present: first, second, third and fourth generation. Combinatorial and sequential circuits. Programmable logic arrays. Processor design: information formats, instruction formats, arithmetic operations and parallel processing. Hardwired and microprogrammed control units. Virtual, sequential and cache memories. Input-output systems, communication and bus control. Multiple CPU systems.

Credits: 3

Prerequisites: 16.265 (Logic Design) and 16.317 (Microprocessors I)

### Course Overview (continued)

Course Objectives: By the end of this course, you should understand and/or be able to use all of the following:

1. **Instruction Set Architecture:** Operations and operands. MIPS instruction set.
2. **Computer Arithmetic:** Binary addition, subtraction, multiplication, and division. Floating point arithmetic.
3. **Processor Core Design:** Datapath and control design. Instruction-level parallelism. Pipelining. Multiple issue and dynamic scheduling.
4. **Memory System Design:** Principles behind memory hierarchy design. Caches. Virtual memory. Cache coherence protocols.
5. **Storage and I/O:** Disk and flash storage. I/O interfacing.
6. **Multiprocessing:** Multiprocessor systems. Multithreading. GPUs. Network topologies.

Grading: Grades will be computed on an A to F scale; A+ grades may only be assigned in the graduate course (16.561), in accordance with UMass Lowell policy. The weights assigned to the various items are:

|                      |     |
|----------------------|-----|
| Homework assignments | 55% |
| Midterm exam         | 20% |
| Final exam           | 25% |

Incomplete grades will only be given in exceptional situations, and the student must be passing the class at the time the grade is requested.

Class participation: You are responsible for all material discussed or announced in class. You are expected to attend class regularly and participate in any in-class discussions, as such exercises are essential to your learning. Although lecture attendance is not explicitly required, regular attendance will improve your understanding of the course concepts.

Exams: Make-up exams will only be offered in exceptional circumstances. You must notify Dr. Geiger as early as possible in order to determine an appropriate make-up date.

Assignment policies: Your assignments will be a mix of typical homework problems and simulation exercises that will require you to do some programming. All assignments will be posted on the course web page and discussion group.

Assignment policies include the following:

- All assignments must be completed individually unless explicitly specified.
- Late assignments are penalized at a rate of 10% per day, including weekends and holidays.
- All assignment solutions must be clearly legible.
- Although typewritten solutions are preferred, handwritten solutions are acceptable in some cases.
- All electronic submissions must be combined into a single file—archive files are not acceptable, nor are scanned pages in which each page is in a separate file.

### **Academic Honesty**

All assignments and exams must be completed individually unless otherwise specified. You may discuss concepts or material covered in class, but may not share any details of your solutions to assigned problems, including algorithms and code. Plagiarism (copying solutions from an outside source) is also unacceptable and will be treated as an instance of cheating.

Students are allowed to discuss assignments in general terms and to help one another fix specific errors. In this case, students are required to note that they received assistance from a classmate by listing that person's name and the nature of their assistance as part of their lab report or homework solution.

Any assignment or portion of an assignment that violates this policy will receive a grade of zero for all parties concerned. Depending on the severity of the infraction, or in cases of repeat violations, additional penalties may be given at the instructor's discretion, up to and including a failing grade in the course.

Further information on the university Academic Integrity policy can be found at:

<http://www.uml.edu/Catalog/Undergraduate/Policies/Academic-Integrity.aspx>

### **Acknowledgements**

My sincere thanks to Jim Moran for providing me with his material from previous semesters of this course, which I used as a starting point for the development of many course materials, including this syllabus.

## Course Schedule

This schedule contains a tentative schedule of topics we will cover throughout the term; the course website will contain the most up-to-date version. The web page will also describe which section(s) of the textbooks are associated with each lecture.

Please note that the exam dates are fixed—the midterm exam will be held on **Thursday, October 16 in class**, and the final exam will be held **during finals, at a date/time to be determined**.

| Week | Date (Th) | Lecture Topics  |
|------|-----------|---|
| 1    | 9/4       | Course overview<br>Computer architecture introduction<br>Instruction set architecture   |
| 2    | 9/11      | Digital arithmetic  |
| 3    | 9/18      | Datapath and control<br>Pipelining  |
| 4    | 9/25      | Instruction level parallelism<br>Branch prediction                                      |
| 5    | 10/2      | Dynamic scheduling  |
| 6    | 10/9      | Speculation<br>Midterm exam preview   |
| 7    | 10/16     | <b>MIDTERM EXAM</b>   |
| 8    | 10/23     | Multiple issue; multithreading  |
| 9    | 10/30     | Memory hierarchies and cache basics   |
| 10   | 11/6      | Virtual memory<br>Cache optimizations   |
| 11   | 11/13     | <i>Wednesday, 11/12: Last day to withdraw (UG)</i><br>Storage                           |
| 12   | 11/20     | Multiprocessors   |
| 13   | 11/27     | <i>Lecture on Wednesday--Thanksgiving</i><br>Topics TBD                                 |
| 14   | 12/4      | <i>Wednesday, 12/3: Last day to withdraw (Grad)</i><br>Topics TBD<br>Final exam preview |
| 15   | 12/11     | <i>No lecture--classes end Wednesday, 12/10</i>   |
|      | TBD       | <b>FINAL EXAM—DATE/TIME TBD</b>   |