Computer Architecture (CS 2420) Spring 2012

Time and Place: 2 - 3:15 PM Tuesday and Thursday in ITTC 328

Web-site: **http://www.cs.uni.edu/~fienup/cs2420s12/**

Class Email List: Send messages to CS-2420-01-spring@uni.edu from your UNI account

Instructor: Mark Fienup (fienup@cs.uni.edu)
Office: ITTC 313
Phone: 273-5918 (Home 266-5379)
Office Hours: M 9-11:45, 1:10-2; T 1:10-2; W 10-11:45 (← in 339 Wright Hall lab); 1:10-3 (← in ITT 313);
Th 1:10-2; F 9-11:45

Prerequisite: Computer Organization (CS 1410/810:041)

Required Text: *Essentials of Computer Organization and Architecture*, 3rd edition, by Linda Null and Julia Lobur. Jones and Bartlett Learning, 2010. ISBN-13: 978-1449600068.

Course Goals: Very few people will become computer architects, but the vast majority of students will be employed as a software "somethings" (software engineer, software tester, programmer, etc.). Unfortunately, efficient software systems can only be produced by people with a thorough understanding of computer hardware and its relationship to software. Therefore, the major goal of this course is to understand the relation between the hardware and software, and how to efficiently use the hardware. To achieve this you will learn about (1) the hardware organization of sequential and parallel computers, (2) the memory hierarchy including cache and virtual memory, (3) system I/O and communication, (4) interrupts, (5) hardware support for operating systems, and (6) utilitization of mult-core processors & many-core GPUs.

Assignments: The majority of your assignments will be "pencil-and-paper" exercises. However, we might have an occasional programming assignments to highlight the importance of understanding the computer architecture when designing and writing software.

Pedagogic Approach: In class, I'll tend to break up the lecture with active (and group) learning exercises to aid learning. While this is not formally graded, part (5%) of your grade will be based on your participation in (and attendence for) these in-class activities. Students benefit by (1) increased depth of understanding, (2) increased comfort and confidence, (3) increased motivation, and (4) being better prepared to work in groups on the job. This might sound great, but it will require you (and me) to work differently to prepare for class. Before the class, you must read the assigned reading, thought about what I asked you to think about, etc.; otherwise you won't be able to effectively participate during class.

Grading policy: There will be three tests (including the final). I'll announce tests at least one week in advance to allow you time to prepare. Tentative weighting of course components is:

In-class Work:	5 %	
Assignments:	29 %	
In-class Test 1:	22 %	(about Feb. 16)
In-class Test 2:	22 %	(about March 29)
Final:	22 %	(Wednesday, May 2 from 1-2:50 PM in ITT 328)

Grades will be assigned based on straight percentages off the top student score. If the top student's score is 92%, then the grading scale will be, i.e., 100-82 A, 81.9-72 B, 71.9-62 C, 61.9-52 D, and below 52 F. Plus and minus grades will be assigned for students near cutoff points.

Scholastic Conduct: You are responsible for being familiar with the University' Academic Ethics Policies (http://www.uni.edu/pres/policies/301.shtml). Copying from other students is expressly forbidden. Doing so on exams or assignments will be penalized every time it is discovered. The penalty can vary from zero credit for the copied

items (first offense) up to a failing grade for the course. If an assignment makes you realize you don't understand the material, ask questions designed to improve your understanding, *not* ones designed to discover how another student solved the assignment. The solutions to assignments should be **individual, original** work unless otherwise specified. Remember: discussing assignments is good. Copying code or test-question answers is cheating.

Any substantive contribution to your assignment solution by another person or taken from a publication (**or the web**) should be properly acknowledged in writing. Failure to do so is plagiarism and will necessitate disciplinary action. In addition to the activities we can all agree are cheating (plagiarism, bringing notes to a closed book exam, etc), assisting or collaborating on cheating is cheating. Cheating can result in failing the course and/or more severe disciplinary actions.

Special Notices:

- In compliance with the University of Northern Iowa policy and equal access laws, I am available to discuss appropriate academic accommodations that may be required for students with disabilities. Requests for academic accommodations are to be made during the first three weeks of the semester, except for unusual circumstances, so arrangements can be made. Students are encouraged to register with Student Disability Services, 103 Student Health Center, to verify their eligibility for appropriate accommodations.

- I encourage you to utilize the Academic Learning Center's free assistance with writing, math, science, reading, and learning strategies. UNI's Academic Learning Center is located in 008 ITTC. Visit the website at http://www.uni.edu/unialc/ or phone 319-273-2361 for more information.

Computer Architecture Spring 2012

Lect #	Tuesday		Thursday	
1	1/10	Computer Organization Review: Sections 4.8-4.12	1/12	Instruction-set Design Issues: Sections 5.1-5.4; 9.1-9.2
3	1/17	Instruction Pipelining: Section 5.5	1/19	Pipeline Data Hazards and forwarding
5	1/24	Branch/Control Hazards: Section 11.5	1/26	
7	1/31	Memory Hierarchy and cache memory: Section 6.1 – 6.4	2/2	
9	2/7	Virtual Memory: Section 6.5	2/9	More Virtual Memory: Multi-level page tables, page-replacement algorithms, frame-allocation algorithms
11	2/14	Review for Test 1	2/16	**Test 1**
13	2/21	Segmentation; Paging of segments; Examples: Section 6.6	2/23	
15	2/28	Hardware support for the operating system support; I/O Control Methods: memory-mapped vs isolated I/O	3/1	programmed-I/O, interrupt-driven, and DMA Sections 7.1-7.4
Spring Break - March 5-9				
17	3/13	Bus: Section 7.5	3/15	
19	3/20	Magnetic Disk, RAID: Sections 7.6, 7.9-7.10	3/22	Superpipeling, Superscalar, and VLIW Processors: Sections 9.1-9.4.1
21	3/27	Tomasula's algorithm and Pentium Example: superscalar pipeline, and cache	3/29	Multiprocessors - SMPs: Section 9.4.4
23	4/3	Test 2 Review	4/5	**Test 2**
25	4/10	Network connected multiprocessors: Section 9.4.5	4/12	Threads and Multi-core processors
27	4/17		4/19	GPU programming
29	4/24		4/26	Review for Final
Final: Wednesday, May 2 from 1-2:50 PM in ITT 328				