

**The University of Texas at El Paso**  
**Fall 2008 Course Syllabus**

**Advanced Computer Architecture CS5341**

<b>Instructor:</b>	Dr. Rodrigo Romero
<b>Office:</b>	Geological Sciences Rm 124
<b>Office Phone Number:</b>	915.747.5916
<b>Email Address:</b>	raromero2@utep.edu
<b>Office Hours:</b>	MW 4:30 – 5:30 p.m. or by appointment
<b>Class Time &amp; Room:</b>	3:00 pm – 4:20 pm, MW, CS Rm 322
<b>Course Materials:</b>	<a href="http://mspace.utep.edu/raromero2/cs5341.f08">http://mspace.utep.edu/raromero2/cs5341.f08</a>
<b>Prerequisites:</b>	CS3320 (Computer Architecture 2) or equivalent

**Course Description:** Students will gain an understanding of the techniques used in modern high-performance computing systems, such as pipelining, superscalar execution, branch prediction, dynamic scheduling, optimizations in cache design, and shared-memory cache consistency protocols; will be introduced to multiprocessor architectures and interconnection networks; and will become familiar with trends in I/O.

**Learning Outcomes:**

**1. Knowledge and Comprehension**

- a. describe the various ways that computer performance can be measured and explain the pros and cons of each
- b. explain the issues associated with instruction set architecture design
- c. explain the concept of and describe the challenges associated with instruction level parallelism
- d. describe the general design of a processor that implements instruction level parallelism
- e. explain the purpose of speculative execution and describe several techniques used to support its implementation
- f. describe how dynamic scheduling overcomes data hazards
- g. describe how dynamic hardware prediction can reduce branch penalties
- h. discuss the various techniques to implement branch prediction
- i. explain how a multiple-issue processor works
- j. explain the role of compilers in terms of performance enhancement
- k. give examples of how a compiler can exploit instruction level parallelism
- l. describe how a cache works and the various cache configurations
- m. enumerate various methods for enhancing cache and memory performance
- n. describe the main difference between multiprocessor architectures that support shared memory and those that support only message passing
- o. list the different types of storage devices
- p. describe the bus that connects CPU/memory with I/O devices
- q. explain the purpose of RAID systems
- r. list the various topologies of interconnection networks

**2. Application and Analysis**

- a. given an address trace, be able to simulate the functioning of a cache
- b. use on-chip performance counters to evaluate the performance of different versions of an application

- c. apply a given snoopy cache consistency protocol to a particular code segment
  - d. apply a given directory cache consistency protocol to a given code segment
- 3. Synthesis and Evaluation**
- a. use the CPU performance equation to compare the performance of processor architectures
  - b. apply Amdahl's speedup law to understand the value of an architectural modification
  - c. analyze the performance of a code segment in a given instruction execution pipeline
  - d. identify data, control and structural hazards for a given processor architecture/code segment pair
  - e. evaluate the cache performance of a given architecture/application pair
  - f. compare the performance of different cache configurations

**Important Dates:**

First Class Meeting:	August 25
First Midterm Exam:	October 1 <sup>st</sup>
Second Midterm Exam:	November 12
Last Class Meeting:	December 3
Final Exam:	Mon, Dec. 8, 1:00 – 3:45 PM

**Textbook:** *Computer Architecture A Quantitative Approach*, Fourth Edition by John L. Hennessy and David A. Patterson

**Grade Basis:**

Quizzes:	15%
Midterm and Final:	45%
Homework and Assignments:	15%
Projects:	15%
Research Papers, summary and analysis:	10%

A grade on a quiz, exam, homework, or project must be contested within one week of notification. Research papers on topics that extend the covered material must be from a published journal or published conference proceedings.

**Guidelines:**

**Reading:** You are expected to do the reading assignment BEFORE the specified class meeting date. Reading assignments are noted on the tentative schedule, below, or will be announced during class meetings. Not all assigned material will be covered in class. It is your responsibility to ask questions in class regarding assigned material that is not fully understood. Portions of quiz questions will be given to assess whether or not you did the reading.

**Homework:** Homework is to be submitted on the due date during the class meeting. It will be graded on an "effort" basis, rather than a "correct answer" basis. Homework answer sheets will be distributed when homework is submitted. Thus, **no late homework will be accepted.** Homework can be group work since this provides a good venue for

learning. Note, however, that often homework will be the basis for quizzes and exams, thus, it is important that you understand the answers to homework problems.

**Projects and Research Papers:** Projects and research paper summaries/analyses are to represent individual work.

**Class Participation:** Class meetings will be interactive and you are expected to frequently participate in a meaningful way. Your meaningful participation will be based on your having completed reading and homework assignments and understood material presented in class meetings.

**Notes:**

**Course Withdrawals:** If you decide to withdraw from the course, you are responsible for ensuring that all steps are taken to formally withdraw. Do not assume that you will be dropped automatically.

**I Grades:** The grade of I (incomplete) will be given ONLY if you are unable to complete the course due to documented appropriate circumstances beyond your control that develop after the last day to withdraw from the course. Appropriate circumstances include illness and death or crisis in your immediate family. In NO case will an I grade be assigned to avoid a grade of D or F in the course.

**Tentative Schedule**

<b>Date</b>	<b>Meeting</b>	<b>Topic</b>	<b>Reading to be done <i>prior</i> to meeting</b>
Aug. 25	1	Introduction and Review of Pipelines Performance, Caches, and Virtual Memory	Appendix A, C
Aug. 27	2	Continuation of Review	
Sept. 3	3	Fundamentals of Computer Design.	Chapter 1
Sept. 8	4	Measuring and Reporting Performance	
Sept. 10	5	Instruction Set Design	Appendix B
Sept. 15	6	Further Discussion of topics in Chapters 1, Appendix B, and Appendix A	Appendix A
Sept. 17	7	Instruction Level	Chapter 2

		Parallelism (ILP)	
Sept. 22	8	ILP: Dynamic Scheduling	Chapter 2 and Appendix A
Sept. 24	9	ILP: Branch Prediction	
Sept. 29	10	ILP: Multiple Issue and Speculation	
Oct. 1	11	<b>Midterm 1</b> Chapters 1 and 2, and Appendices A and B	
Oct. 6	12	ILP: Limitations and Case Study	Chapter 3
Oct. 8	13	Exploiting ILP with software approaches	Handout (3ed. Chapter 4)
Oct. 13	14	Exploiting ILP: advanced compiler support and hardware support	
Oct. 15	15	Multiprocessors and Thread-level parallelism	Chapter 4
Oct. 20	16	TLP. Synchronization; cross-cutting issues	Chapter 4
Oct. 22	17	Caches	Appendix C
Oct. 27	18	Cache Performance I	
Oct. 29	19	Cache Performance II	Chapter 5
Nov. 3	20	Main Memory and Memory Technology	
Nov. 5	21	Virtual Memory	
Nov. 10	22	Memory Hierarchy Design	
Nov. 12	23	<b>Midterm 2</b> Chapters 1-5, Appendices A, B, C	
Nov. 17	24	AMD Opteron memory hierarchy	
Nov. 19	25	Multicore Memory Hierarchy	
Nov. 24	26	I/O I	Chapter 6
Nov. 26	27	I/O II	
Dec. 1	28	Networking I	Appendix E
Dec. 3	29	Networking II	
Dec. 8		<b>Final Exam</b>	