

EE 5123 --- Computer Architecture (Spring 2018)

Course Syllabus

Instructor:

Lide Duan

Assistant Professor

Department of Electrical and Computer Engineering

University of Texas at San Antonio

<http://utsa-salsa.org/>

Office: AET 2.376

Phone: 210-458-5208

Email: lide.duan@utsa.edu

Class Meeting Time: 6:00-7:15PM (Mondays and Wednesdays)

Class Location: AET 0.214

Office Hours: 4:00 – 6:00PM (Mondays and Wednesdays) or by appointment.

Website:

All class materials will be available electronically through Blackboard Learn. These include:

- Slides, HW, solutions, and other course materials distributed by the instructor.
- Notifications sent out by the instructor (please do check your emails!)
- HW/project submissions by students.

Course Description:

Computer architecture fundamentals, instruction set architectures, memory and cache hierarchy, microprocessor pipelining, instruction-level parallelism, data-level parallelism, and thread-level parallelism.

Prerequisite:

Graduate standing. Non-experience in computer architecture is assumed. However, course projects will require programming skills in high-level languages (e.g., C, C++, Java, Python).

Textbook:

Computer Architecture: A Quantitative Approach, 5th Edition. (By John Hennessy and David Patterson). Copyright: © Morgan Kaufmann 2012. Published: 16th September 2011. Imprint: Morgan Kaufmann. eBook ISBN: 9780123838735. Paperback ISBN: 9780123838728.

Course Topics (tentative):

- Fundamentals of computer architecture
- Instruction set architectures (ISA)
- Cache and memory hierarchy design
- Pipelining
- Instruction-level parallelism (ILP)
 - Branch predictions, dynamic scheduling, multiple issue and static scheduling, speculations, compiler techniques, ILP limitations, etc.
- Data-level parallelism (DLP)
 - Vector architectures, SIMD, GPUs, loop-level parallelism.
- Thread-level parallelism (TLP)

- Centralized shared-memory, distributed shared-memory, multiprocessor cache coherence, synchronization, consistency, etc.

Grading Policy:

- Homework assignments: **30%**
 - 3 assignments of 10% each
- Exams: **35%**
 - Exam 1 (10%) + Exam 2 (10%) + Final (15%)
- Projects: **30%**
 - Project 1 (15%) + Project 2 (15%)
- Quizzes: **5%**
 - The instructor will randomly choose a few classes to ask the students to turn in an answer to a very simple question. These are mainly for checking attendance.
- Total: **100%**

About the Grading

- The final letter grades will be curved based on the scores and ranks. About 50% of the students will receive an A (including A+, A, and A-).
- After the grade of each assignment/exam/project/quiz is posted, you have up to a week to see me for any errors. After that, the grade is finalized.

About the HW Assignments

- There will be three HW assignments; each is assigned and due before an exam.
- Tentatively, the three assignments will be assigned in the 4th, 9th, and 15th week of the semester.
- All assignments must be submitted in pdf through Blackboard Learn before the specified deadlines. There will be no delay policy, and submissions missing the deadline will NOT be graded.
- HW questions are good examples of exam questions.
- No copying is allowed on any assignment. All parties involved (either copying or being copied) will lose all the points for the assignment.

About the Exams

- Tentatively, Exam 1 will be on Monday 2/12 (class time); Exam 2 will be on Monday 3/26 (class time); and Final Exam will be on Monday 5/7 (6PM).
- All exams are open-books and open-notes. You can bring your laptop for accessing class materials, but Internet access and communication with other are not allowed.
- The final exam will be comprehensive, covering what have been taught in the whole semester.
- No make-up exams (except for extremely special situations with legitimate proof and under discretion of the instructor).

About the Projects:

- There will be two projects. Project 1 will be well-defined by the instructor, and Project 2 will be proposed by the students.
- Project teaming policy:
 - Students can form teams to conduct the projects; each team can have up to 3 students.
 - Project evaluation will take into account the project workload, the number of team members, the project quality, etc.

- All team members will receive the same project score.
 - The teams must be the same for both projects.
- Project 1 is a disassembler of MIPS binary code. You will be given a MIPS binary file (in txt format), and need to disassemble it into the corresponding MIPS assembly code.
 - Timeline: assigned in the 2nd week; teams formed by the 3rd week; due in the 7th week.
- Project 2 will be proposed by the students.
 - Timeline: starts in the 8th week; topics proposed by the 11th week (after the Spring Break); due at the end of the semester.
 - A few examples will be given by the instructor, but the students are highly encouraged to propose their own topics.
 - Some implementation must be included; a report must be submitted.
- Students have the flexibility to choose the language and environment for the implementation of the projects. However, you should use a free environment in order for the instructor to reproduce the results. Therefore, developing under Linux using a free compiler is highly recommended. In your submission, you will need to include a readme file (in txt) that specifies the instructions to compile and run your program. It is your responsibility to make sure that your program can be run successfully by following your readme file with reasonable effort.
- Copying source code (from your classmates or students who took this course before) is NOT allowed. Automatic source code checking tools, in addition to manual effort, will be applied to check for coding plagiarism. All parties involved (either copying or being copied) will lose all the points for the project. So, protect your source code!