San José State University
Department of Computer Science

CS 247 Advanced Computer Architecture

Instructor: Soon Tee Teoh
Email: *soontee.teoh@sjsu.edu*
Office Hours: By appointment only, via Zoom

Class Days/Time: MW 3:00 - 4:15 pm
Zoom link: https://sjsu.zoom.us/j/132407857
Prerequisites: CS 147 (undergraduate computer architecture) and CS 149 (undergraduate operating systems)

Course Format

This course will be taught primarily via classroom presentations. Zoom lectures will be recorded. Students are expected to be present during the 4 scheduled exams during the class period. Students are also expected to be present during one class presentation session. Otherwise, attendance is optional. In addition, homework demos are scheduled during regular class time. However, students can schedule other times to demo if the regular class time is inconvenient.

Canvas Course Website

Course materials, syllabus, assignments, grading criteria, exams, and other information will be posted on the Canvas Learning Management System course login website at http://sjsu.instructure.com.
You are responsible for regularly checking these websites to learn of any updates. You can find Canvas video tutorials and documentations at http://ges.sjsu.edu/canvas-students.

Course Catalog Description

Advanced topics in vector architectures, including: pipelined architectures, dataflow computers, VLSI architectures, butterfly connections; bus and memory architectures; cache structures; hardware implementations of algorithms.

Course Description

Detailed analysis of high-performance, fault-tolerant computer systems. Survey various machine architectures including implementation alternatives for major processor sub-systems. Pipelined, vector, VLSI, multi-core and dataflow architectures are examined. Discussion includes data representation, arithmetic logic unit operations and algorithms, rounding algorithms, control unit operation and instruction formats. Performance measurement and speedup techniques are studied to perform tradeoff analysis and design optimization. A written report and oral presentation on a relevant and approved topic of interest to the student will be required. Class participation during student oral presentations will be required.

Course Learning Outcomes (CLO)

Upon successful completion of this course, students will be able to:

- Understand combinatorial and sequential circuit structures and Boolean number representation schemes
- Appreciate how the fundamental core mathematical operations such as addition, subtraction, multiplication, and division can be optimized with appropriate number representation, rounding, and digital circuit implementation schemes.
- Explain the tradeoffs between complex instruction set computers (CISC) and reduced instruction set computers (RISC).
- Discuss non-classical architectures such as parallel processors, multi-core chips, pipelined and VLIW machines which are used to accelerate hardware performance without impacting legacy sequential software programming languages or techniques.
- Emphasize the importance of fault-tolerant design techniques and examine various methods of error detection and correction such as TMR and Hamming Codes.
- Analyze and perform tradeoffs between the cost, performance, and reliability of alternative computer architectures.
- Utilize computer-aided design tools and hardware description languages useful to computer architects in performing functional verification and performance measurements of digital systems.
- Use industrial-grade field programmable gate array chips and their associated CAD toolsets.
- Appreciate how hardware and software (especially the operating system and compilers) must work synergistically together to provide optimum throughput.
- Perform an in-depth investigation of an architecture related topic of interest to them and present their findings to their classmates in an oral and written report using a venue similar to that used in formal professional technical conferences.

Academic Integrity

You may study together and discuss the assignments, but what you turn in must be your individual work. Copying code from another student's program or sharing your program code are equally serious violations of academic integrity. Never use code you find on the web, unless you have the instructor's permission, and then you must give proper attribution in your comments. This is similar to giving attribution to a quote that you use in a term paper. Assignment submissions will be checked for plagiarism using Moss from the Department of Computer Science at Stanford University. See http://theory.stanford.edu/~aiken/moss/. Violators of academic integrity will suffer severe sanctions, including academic probation. Students who are on academic probation are not eligible for work as instructional assistants in the university or for internships at local companies.

Recommended Texts

Computer Organization and Design: The Hardware/Software Interface, 4th Ed., Revised Printing, D. Patterson, 2009, Morgan Kaufmann, ISBN 9780123744937

Software and Computer

You need to write code for some of the homework assignments. Your code can be in C, C++, Java or Python. For these programming homework assignments, you will also need to demo your program, so it is necessary for you to have a computer where you can share your screen on Zoom and communicate verbally with the instructor. You will also have to make a class presentation via Zoom, and also take your exams with your camera on using Zoom.

Course requirements and assignments

There will be homework assignments, exams and class presentations.

Homework Assignments

There will be multiple programming assignments throughout the semester. Each assignment will be worth a specified maximum number of points. Assignments can be turned in within 48 hours late for 20% deduction. After 48 hours, no submission is allowed (it will get a 0 score). You are also required to demo your homework to the instructor. During the demo, you will be asked questions about your code. Points will be deducted for wrong answers. Failure to demo your homework results in zero points for the assignment.

Exams

All exams are open book and must be taken at the scheduled time. All students taking the exams must be present on Zoom during the exam. The exams will test understanding (not memorization) of the material taught during the semester. Instant messaging, e-mails, texting, tweeting, file sharing, or any other forms of communication with anyone else during the exams will be strictly forbidden. There can be no make-up exams unless there is a documented medical emergency.

Class Presentation

Each student has to give one class presentation, during one of the days on the schedule. For this presentation, each student needs to read up on a topic and tell the class about it. More instructions on this assignment will be given later in the term.

The university's syllabus policies:

• University Syllabus Policy S16-9 at http://www.sjsu.edu/senate/docs/S16-9.pdf

• Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/.

"Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally 3 hours per unit per week with 1 of the hours used for lecture) for instruction or preparation/studying or course related activities including but not limited to internships, labs, clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus."

Grading Information

Your final class grade will be weighted as follows:

4 Written Homework Assignments: 20%

2 Programming Assignments: 30%

2 Exams: 30%
1 Class presentation: 20%

Final course grades will be based on a curve. The median total score will earn a B+. Approximately one third of the class will earn higher grades, and another one third will earn lower grades.

Classroom Protocol

It is very important for each student to attend classes and to participate. Cell phones in silent mode, please. No use of electronic devices (phones, tablets, laptop computers etc.) in class, except to take notes.

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/.

Course Schedule

This is the tentative course schedule. The topics covered may be changed. However, the demo, presentation and exam dates are expected to be fixed.

Week	Mon	Wed
0		8/19/2020
		Introduction
1	8/24/2020	8/26/2020
	Decoders and Multiplexers	Numbers and Arithmetic
2	8/31/2020	9/2/2020
	Flip-Flops	Registers
3	9/7/2020	9/9/2020
	Labor Day (no lecture)	Datapath
4	9/14/2020	9/16/2020
	Architectures	Floating-point

6	9/21/2020	9/23/2020
	Review	Exam 1
7	9/28/2020	9/30/2020
	Pipelined Datapath	Parallel Architectures
8	10/5/2020	10/7/2020
	Memory and Caches	Caches
9	10/12/2020	10/14/2020
	Cache Prefetching	Branch Prediction
10	10/19/2020	10/21/2020
	Virtual Memory, Disks, RAID	Homework Demo
11	10/26/2020	10/28/2020
	Review	Exam 2
12	11/2/2020	11/4/2020
	GPGPU	Vector Processor
13	11/9/2020	11/11/2020
	Homework Demo	Veterans Day (no lecture)
14	11/16/2020	11/18/2020
	Big Data	Machine Learning
15	11/23/2020	11/25/2020
	Office hours (no lecture)	Non-Instructional Day (no lecture)
16	11/30/2020	12/2/2020
	Office hours (no lecture)	Student Presentations
17	12/7/2020	
	Student Presentations	