San José State University
Computer Science Department
CS159, Parallel Processing, Section 1, Spring 2020

Course and Contact Information

Instructor:	Robert Chun
Office Location:	MH 413
Telephone:	(408) 924-5137
Email:	Robert.Chun@sjsu.edu
Office Hours:	MW 4:30pm-5:30pm
Class Days/Time:	MW 1500 - 1615
Classroom:	MH222
Prerequisites:	A Computer Architecture Class and An Operating Systems Class

Faculty Web Page

Course materials such as presentation slides, notes, assignments, etc. can be found on my faculty web page at http://www.sjsu.edu/people/Robert.Chun/courses

Course Description

A hardware architecture and software development class focused on multi-threaded, parallel processing algorithms and techniques. A detailed study of high-performance parallel processing hardware architectures ranging from on-chip Instruction-Level Parallelism to multi-core microprocessor chips to large distributed supercomputing systems including Clusters, Grids, and Clouds. Discussion and hands-on exercises in a broad range of various parallel programming paradigms and languages such as Pthreads, MPI, OpenMP, Map-Reduce Hadoop, CUDA and OpenCL. The class will focus on the fundamental concepts associated with the design and analysis of parallel processing systems. Special emphasis will be placed on avoiding the unique non-deterministic software defects that can arise in parallel processing systems including race conditions and deadlocks. A term project and oral presentation on a topic selected by the student will be required.

Course Learning Outcomes (CLO)

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

- Understand the Technical and Business motivation and need for current state-of-the-art computing systems to incorporate Parallel Processing into the Hardware and Software Subsystems.
- Explain the Micro-Hardware Architectural Evolutionary Trends leading to on-chip Instruction-Level Parallelism, and Pipelining, SuperScalar, Multi-Function Unit Parallel Processing.
- Understand the Macro-Hardware Architectural Evolutionary Trends leading to Parallel Processing including Flynn's Taxonomy and the recent progression in high-performance supercomputing architectures from Clusters to Grids and to Clouds.
- Explain data dependency analysis & hazards, and Amdahl's Law, which limits the amount of practical speedup and scalability that can be achieved with Parallel Processing.
- Perform Design and Analysis Techniques for Parallel Processing Systems including the identification of data vs. task partitioning in algorithms and applications.
- Understand the Different Models for implementing parallelism in Computing Systems such as shared memory and message passing.
- Explain the software challenges associated with Parallel Processing including the difference between concurrent vs. parallel execution models, deadlocks and race conditions.
- Understand a sample of current parallel programming paradigms and languages and be able to write parallel programs using them.

Required Texts/Readings

Textbooks

Multi-Core Programming, Shameem Akhter and Jason Roberts, 2006, Intel Press, ISBN 0-9764832-4-6

Using OpenMP, Barbara Chapman, 2008, MIT Press, ISBN 978-0-262-53302-7. Students can access this entire textbook for free via the SJSU Library at: https://library.sjsu.edu/
Search on "9780262255905" (the ISBN Number)

Web Resources

See Informational Sheet: "Useful Web Links for Parallel Processing Course"

Course Requirements and Assignments

Students are expected to attend all class meetings. Assignments include two midterms, one final, a written and oral report, a set of projects (consisting of a combination of written problems and programming assignments), and active participation during student presentations, weighted as follows. Grading is based on a class curve. All assignments (especially the oral presentation) must be completed by the student on the due date specified to receive credit for the class. Late assignments (including the scheduled oral presentations) or late exams are not accepted. All students must uphold academic honesty, especially for the required term paper, per university policy detailed at
http://www.sjsu.edu/specialed/docs/current-forms/AcademicIntegrityPolicy.pdf

Final Examination

The final exam for the class will be held on Wednesday, May 13, 2020 at 1215-1430

Grading Information

Assignments include two midterms, one final, a written and oral report, a set of projects (consisting of a combination of written problems and programming assignments), and active participation during student presentations, weighted as follows. Grading is based on a class curve. All assignments (especially the oral presentation) must be completed by the student on the due date specified to receive credit for the class. Late assignments (including the scheduled oral presentations) or late exams are not accepted. All students must uphold academic honesty, especially for the required term paper, per university policy detailed at http://www.sjsu.edu/specialed/docs/current-forms/AcademicIntegrityPolicy.pdf

15%	Midterm Exam 1 Week 6 (Approximate)
15%	Midterm Exam 2 Week 12 (Approximate)
25%	Written Term Paper/Project & Oral Presentations Weeks 13-15
25%	Final Exam Wednesday, May 13, 2020 at 1215-1430
10%	Combined total of Three HW/Projects Due as announced in class
10%	Active Participation during Student Presentations Due as announced in class

Classroom Protocol

Students are expected to attend all classes, ESPECIALLY THE TERM PAPER ORAL PRESENTATIONS.

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/

CS159 Spring 2020 Tentative Course Schedule

Lecture	Topic
1 - 3	Introduction, Motivation and Overview of Parallel Processing with an emphasis on the Micro- and Macro-Hardware Evolutionary Trends leading to Parallelism and the Software Challenges of Parallelism
4 - 6	Hardware Pipelining and Instruction-Level Parallelism (ILP)
7 - 8	Multi-Function Parallelism in Hardware
9	Data dependency analysis and control hazard analysis including RAW, WAR, WAW, and Branch Prediction
10	Limitations of Hardware-based, Software-transparent ILP
11 - 17	Software Challenges of Parallel Processing including Concurrent vs. Parallel Execution Models, Amdahl's Law, Deadlocks, Race Conditions, Semaphores
18	Models of Parallelism such as Shared Memory, Message Passing
19 - 21	Parallel Programming Paradigms including Unix Process Forking, PVM, MPI, OpenMP, CUDA, OpenCL, Hadoop Map-Reduce, GPGPU Computing, Toolsets for Parallel Program Software Development and Debugging.
Final Exam	Wednesday, May 13, 2020 at 1215-1430