SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Design and Analysis of Algorithms

CS 255

Spring 2026 Section 01 In Person 3 Unit(s) 01/22/2026 to 05/11/2026 Modified 01/22/2026

Contact Information

Instructor: Dr Katerina Potika

Email: katerina.potika@sjsu.edu
Website: https://www.sjsu.edu/people/katerina.potika/ (https://www.sjsu.edu/people/katerina.potika/)

Office Hours

Tuesdays and Thursdays, 9:30 -11 AM, or by appointment (hybrid)
MH 215

Join from PC, Mac, Linux, iOS, or Android: Zoom meeting link

https://sjsu.zoom.us/j/84355230699?pwd=XwOdRlyWrMEEcSzTeFoB9bDPr3s54w.1

Course Information

The course adopts an in-person classroom delivery format. An internet connection, a computer, a tablet, or a smartphone is required. Paper and pen.

Faculty Web Page and MYSJSU Messaging

Course materials, such as the syllabus, handouts, notes, assignment instructions, etc., can be found on the Canvas Learning Management System at http://sjsu.instructure.com. You are responsible for regularly checking the messaging system through MySJSU on the Spartan App Portal (http://one.sjsu.edu) (or other communication system as indicated by the instructor) for any updates. For help with using Canvas, see the Canvas Student Resources page (http://www.sjsu.edu/ecampus/teaching-tools/canvas/student_resources).

Course Description and Requisites

Randomized algorithms. Parallel algorithms. Distributed algorithms. NP-completeness of particular problems. Approximation algorithms.

Prerequisite(s): CS 155 and Graduate standing. Allowed Declared Major: Computer Science, Bioinformatics, Data Science. Or instructor consent.

Letter Graded

Classroom Protocols

Attendance is highly recommended. Please avoid disrupting the class: turn off cell phones (or put them on vibrate), no text messaging during class or exams, no taking pictures or videos, avoid coming late, and no talking or whispering with other students during the instructor's presentation. You may not publicly share or upload material from this course, such as exam questions, lecture notes, or solutions,s without my consent.

Academic Integrity & AI Policy

AI-based tools (such as ChatGPT Edu) are permitted only for brainstorming, summarizing, and grammar/syntax checking of assignments. They may not be used for drafting or writing content, code, pseudocode, or proofs. All submissions must be original, human-written work. If AI is used, you must include a "Use of AI" statement explaining its specific role and list the prompts used. You are solely responsible for the technical integrity of your work. Always verify AI output, as it may contain hallucinations. Presenting AI-generated text or code as your own will be treated as a violation of academic integrity.

Program Information

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

Course Learning Outcomes (CLOs)

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

1. analyze and code randomized algorithms
2. analyze or code a parallel algorithm, using an appropriate library if needed
3. evaluate a distributed algorithm’s runtime and prove its correctness
4. categorize various problems as being in P or NP-Complete, with proof
5. prove simple approximation ratios for optimization versions of NP complete decision problems

Course Materials

No required textbook, we will use chapters from various books:

1. Cormen, Leiserson, Rivest and Stein, Introduction to Algorithms, 3rd Edition
   MIT Press, 2009. You can find errata (bug reports) for the

book http://www.cs.dartmouth.edu/~thc/clrs-bugs/bugs-3e.php.
2. Kleinberg and Tardos, Algorithm Design, First Edition, Addison-Wesley, 2005.
3. Dasgupta, Papadimitriou, and Vazirani, Algorithms, McGraw-Hill, 2006.
4. Vazirani, Approximation Algorithms, Springer, 2003
5. Rajeev Motwani and Prabhakar Raghavan, Randomized Algorithms.

Course Requirements and Assignments

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 three hours per unit per week) for instruction, preparation/studying, or course-related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus.

Penalty for late submission 5% for every 3 days up to 9 days; after that, no submission will be accepted. Never email your assignments.

Homework assignments: individual, regularly assigned, will include written problem assignments, and perhaps some online exercises. Solutions will not be posted. The homework is a tool for you to learn the material and prepare you for the exams.

Reading assignments: Regular assignments for the next class.

Quizzes: Unannounced quizzes (at least 4) may be given during class, each lasting about 10 minutes. These generally are problems from the reading assignment and/or the homework.

Project (Programming and Presentation): A programming project of your choice related to the course, in groups of two students. At the end of the semester, you will present the project in class. Never use any code you find on the web, unless it is given to you by me.

Midterm exams: Two Midterm exams during the semester.

Final Examination or Evaluation: One final, written, and cumulative exam, split into two parts. The exams contain multiple-choice, short-answer, and pseudocode/computation questions.

Grading Information

No extra point options.

The final exam is comprehensive. No make-up exams except in case of verifiable emergency circumstances.

Determination of Grades

Final Grade:

- 20% Project (programming, report, and presentation)
- 10% Quizzes

- 10% Homework
- 5% Participation
- 5% Discussions
- 30% Midterms (each one 15%)
- 20% Final

Grade	*Percentage*
A plus	96 to 100%
A	93 to 95%
A minus	90 to 92%
B plus	86 to 89 %
B	82 to 85%
B minus	78 to 82%
C plus	74 to 77%
C	70 to 73%
C minus	65 to 69%
D plus	62 to 64%
D	58 to 61%
D minus	55 to 57%
F	<54%

University Policies

Per University Policy S16-9 (PDF) (http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on the Syllabus Information (https://www.sjsu.edu/curriculum/courses/syllabus-info.php) web page. Make sure to visit this page to review and be aware of these university policies and resources.

Course Schedule

The schedule is subject to change with fair notice and announced on Canvas.

Course Schedule

Lesson	Date	Topic	Assignments
1	1/22	Syllabus	
2	1/27	Introduction: Algorithms & Computers, Challenge exam	HW 1
3	1/29	Running time, growth of functions	Challenge exam
4	2/3	Graphs, BFS, DFS, topological sorting	
5	2/5	Parallel and Distributed Algorithms	
6	2/10	PRAM model	
7	2/12	Greedy Algorithms: Scheduling, Shortest paths, Caching, knapsack	
8	2/17	Greedy Algorithms: Minimum spanning tree, clustering	
9	2/19	Divide & Conquer: sorting, integer/matrix multiplication, max subarray	HW 2
10	2/24	Divide & Conquer: computational geometry	
11	2/26	Dynamic Programming: scheduling, knapsack	

12	3/3	Dynamic Programming: all-pairs shortest path	
	3/5	Midterm 1	
13	3/10	Network flow, applications	Project proposal (due)
14	3/12	Network flow, applications	
15	3/17	Heaps, Amortized Analysis	
16	3/19	Amortized Analysis cont.	
17	3/24	Randomization: Quicksort	Project Sprint 1 (due)
18	3/26	Randomization: Hashing	
	3/30-4/3	Break	
19	4/7	Intractability, P, NP, NP-completeness, reductions	Project Sprint 2 (due)
20	4/9	Intractability, P, NP, NP-completeness, reductions	
21	4/14	Intractability, P, NP, NP-completeness, reductions	HW 3
22	4/16	Midterm 2	
	4/21	Approximation Algorithms	
23	4/23	Approximation Algorithms	Project Sprint 3 (due)
24	4/28	Distributed Algorithms	
25	4/30	Distributed Algorithms	
26	5/5	Project Presentations	

27	5/7	Project Presentations	
	Final	Tue, May 19 10:45 AM-12:45 PM	