SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Data Structures and Algorithms Section 09

CS 146

Fall 2023 3 Unit(s) 08/21/2023 to 12/06/2023 Modified 08/18/2023

Contact Information

Instructor	Dr. Tahereh Arabghalizi (Prof. Tɒh)
Office	MH 217
Email	tahereh.arabghalizi@sjsu.edu
Office Hours	M/W 12-1:15 pm and TU/TH 3-4:15 pm (Email (mailto:tahereh.arabghalizi@sjsu.edu) OR by appointment (https://calendly.com/tahereh-arabghalizi-sjsu-fall23)) - All questions should be asked during the office hours, unless they are short/simple questions.
Teaching Assistant	Pranathi Kunadi (pranathi.kunadi@sjsu.edu)

Course Description and Requisites

Implementations of advanced tree structures, priority queues, heaps, directed and undirected graphs. Advanced searching and sorting (radix sort, heapsort, mergesort, and quicksort). Design and analysis of data structures and algorithms. Divide-and-conquer, greedy, and dynamic programming algorithm design techniques.

Prerequisite(s): MATH 30, MATH 42, CS 49J (or equivalent knowledge of Java), and CS 046B (with a grade of "C-" or better in each); Computer Science, Applied and Computational Math, Forensic Science: Digital Evidence, Software Engineering, or Data Science majors only; or instructor consent.

Letter Graded

Classroom Protocols

Students are expected to adhere to the Student Conduct Code found at the SJSU Student Conduct website (http://www.sjsu.edu/studentconduct/) (http://www.sjsu.edu/studentconduct/). Additionally, students should regularly attend lectures and labs (if applicable), treat instructors and peers with respect, and refrain from the use of cell phones during any classroom activities.

- Regular class attendance is highly recommended and strongly encouraged.
- Please arrive to class on time so that you benefit fully from the course experience and you do not disturb classmates and the instructor while class is in session.
- Students are responsible for knowing all materials covered in class lectures, readings, assignments, and other course-related work.
- Laptops, tablets, and other devices should only be used for course-related purposes.
- Do NOT share any course material publicly (on Canvas, GitHub, etc.) without permission, including but not limited to lecture notes, lecture videos, passwords, homework/exam solutions, and class meeting links.
- All students are required to have access to a wireless laptop (running OSX, Windows, or some version of UNIX), upon which you can install required software. Technology used will include Canvas, programming in Java, and an IDE (Integrated Development Environment).

- Unannounced in-class quizzes and pop questions may be given anytime during class. The purpose of in class quizzes and pop questions is to encourage you to learn, study and review the concepts and materials presented/discussed in the lecture. These will generally be problems covered in the today's or previous lecture.
- You must prove that your prerequisite courses have been satisfied. If you do not submit prerequisites by due date, you might be dropped from the course, if other students are waiting for your space.

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. Implement lists, stacks, queues, search trees, heaps, and graphs and use these data structures in programs they design.
2. Prove basic properties of trees and graphs.
3. Perform breadth-first search and depth-first search on directed as well as undirected graphs.
4. Use advanced sorting techniques (radix sort, heapsort, mergesort, quicksort).
5. Determine the running time of an algorithm in terms of asymptotic notation.
6. Solve recurrence relations representing the running time of an algorithm designed using a divide-and-conquer strategy.
7. Comprehend the basic concept of NP-completeness and realize that they may not be able to efficiently solve all problems they encounter in their careers.
8. Comprehend algorithms designed using greedy, divide-and-conquer, and dynamic programming techniques.

Course Materials

Required Texts/Readings

Cormen, Leiserson, Rivest, and Stein, Introduction to Algorithms, third edition. MIT Press, 2009. ISBN-10: 0262033844 ISBN-13: 978-0262033848

Course Requirements and Assignments

- **Reading Assignments**: You may be assigned readings from the textbooks or papers/articles.
- **Videos**: Videos maybe be posted to introduce new topics not covered in class or as supplementary materials.
- **Online Quizzes**: These will be used to test your understanding of the material.
- **In Class Discussions and Activities**: We use some time to discuss and gain intuition about topics covered in lecture.
- **Programming Assignments**: Programming assignments are to be done individually, unless otherwise specified. They can be discussed but should be implemented individually. More information is given at the time of the first programming assignment. Never use any code you find on the web, unless I provide it. Some assignments may have an oral discussion or examination.
- **Midterm Exam**: there will be one exam (midterm) during the semester.
- **Final Exam**: The final exam could be comprehensive or non-comprehensive (TBD).

Grading Information

Course weightings will be as follows:

- 10% Quizzes
- 30% Programming Assignments
- 30% Midterm Exam

- 30% Final Exam

Exams may be curved (up) to raise their grades if needed.

Your course grade will be determined by your final weighted average:

- *A plus = 97% or higher*
- *A = 93% up to 97%*
- *A minus = 90% to 93%*
- *B plus = 87% to 90%*
- *B = 83% to 87%*
- *B minus = 80% to 83%*
- *C plus = 77% to 80%*
- *C = 73% to 77%*
- *C minus = 70% to 73%*
- *D plus = 67% to 70%*
- *D = 63% to 67%*
- *D minus = 60% to 63%*
- *F = 0% to 60%*
- Boundary cases count as the higher of the two grades.

All students have the right, within a reasonable time, to know their academic scores, to review their grade-dependent work, and to be provided with explanations for the determination of their course grades. See University Policy S20-2 for more details.

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 course schedule is subject to change with fair notice. Changes will be announced on Canvas.

Week	Topics
1	Course Introduction, Prerequisites Check
1	Review (Recursion, Lists, Stacks, Queues)
2	Loop Invariants
2	Asymptotic Growth
3	No class on 09/04 and 09/05 (Labor Day)
3	Recurrence Relations

Week	Topics
4	Master Theorem
4	Heaps and Heapsort
5	Quicksort, Quickselect, and Mergesort
5	Sorting Lower Bounds
6	Linear Time Sorts (Counting, Bucket, Radix)
6	Balanced Search Trees
7	Balanced Search Trees
7	Review
8	**Midterm Exam**
8	Graph Introduction (Representation, BFS, DFS)
9	Topological Sort, Strongly Connected Components
9	MSTs
10	Disjoint Sets
10	Shortest Paths
11	Shortest Paths
11	Dynamic Programming
12	Dynamic Programming
12	Floyd-Warshall
13	NP

Week	Topics
13	NP
14	NP
14	No class on 11/22 and 11/23 (Thanksgiving)
15	Review
15	Review
Final Exam	Check Date and Time Here (https://www.sjsu.edu/classes/final-exam-schedule/fall-2023.php)