SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Data Structures and Algorithms Section 07

CS 146

Spring 2023 3 Unit(s) 01/25/2023 to 05/15/2023 Modified 01/24/2023

Contact Information

Office Location:	MH 218
Class Day/Time	MW 10:30am-11:30pm
Location	MacQuarrie Hall 422

Instructor: Professor Nada Attar

Email: nada.attar@sjsu.edu
Phone: (408) 924-5108
Website: https://www.sjsu.edu/people/nada.attar/ (https://www.sjsu.edu/people/nada.attar/)

Office Hours

A day following class
Tuesday, Thursday, 11:30 AM to 12:30 PM, Zoom Only

https://sjsu.zoom.us/j/86030605405?pwd=Qmhya3NhL0lXVWxUb0lPdHBJb0VqUT09 (https://sjsu.zoom.us/j/86030605405?pwd=Qmhya3NhL0lXVWxUb0lPdHBJb0VqUT09)

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

The lectures will be as an in-person mode. Regular class attendance is highly recommended and strongly encouraged. This section has online office hours. Please have your camera on during office hours. Do not publicly share or upload material for this course such as exam questions, lecture notes, or solutions without my consent.

Students are not allowed to share any of the materials of the course without the instructor's consent.

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 Goals

1. To ensure that students are familiar with ways to implement elementary data structures and their associated algorithms.
2. To introduce students to the implementation of more complex data structures and their associated algorithms.
3. To acquaint students with advanced sorting techniques.
4. To teach students how to determine the time complexity of algorithms.
5. To introduce students to algorithm design techniques.

Course Learning Outcomes (CLOs)

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

1. Understand the implementation of lists, stacks, queues, search trees, heaps, union-find ADT, and graphs and be able to 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 (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. Understand 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. Understand algorithms designed using greedy, divide-and-conquer, and dynamic programming techniques

Course Materials

Programming Language

Java (version 7 or later)

Introduction to Algorithms

Author: Cormen, Leiserson, Rivest, and Stein
Publisher: MIT Press, 2009
Edition: 3rd Edition
ISBN: ISBN-10: 0262033844ISBN-13: 978-0262033848

You can find errata (bug reports) for the book http://www.cs.dartmouth.edu/~thc/clrs-bugs/bugs-3e.php,for whichever printing of the book you get.

Course Requirements and Assignments

SJSU classes are designed such that in order to be successful, it is expected that students will spend

a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on. More details about student workload can be found in University Policy S12-3at http://www.sjsu.edu/senate/docs/S12-3.pdf.

Homework assignments will be individual, regularly assigned, will include written problem assignments, and perhaps some online exercises. The homework is a tool for you to learn the material and prepare you for the exams.

Final Examination

One final cumulative exam.

The exams will contain multiple choice questions, short answer questions and questions that require pseudocode and/or computations. Students must obtain >50% in quizzes and final exam in order to be eligible for a passing grade.

✔ Grading Information

Your grade for the course will be based on the following components:

- Exam 1 - 20%
- Exam 2 - 20%
- Final Exam - 20 %
- Quizzes - 15 %
- Assignments - 25 %

🏛 University Policies

Per University Policy S16-9 (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 Syllabus Information web page (https://www.sjsu.edu/curriculum/courses/syllabus-info.php) (https://www.sjsu.edu/curriculum/courses/syllabus-info.php). Make sure to visit this page to review and be aware of these university policies and resources.

📅 Course Schedule

This schedule is subject to change with fair notice via the Canvas messaging system.

Week	Date	Topic	Reading
1	W 1/25	Introduction: syllabus, Course mechanic & Logistics	
2	M 1/30	Review Data Structures (lists, stacks, queues, trees)	Ch.10
2	W 2/1	Basic algorithms, Insertion sort Growth of functions- O, Ω, Θ, o, ω	Ch1, 2, 3
3	M 2/6	Divide and Conquer technique: Merge Sort, other examples	Ch.4
3	W 2/8	Solving recurrences	Ch.4
4	M 2/13	Master Theorem	Ch.4
4	W 2/15	Heapsort, Priority Queues	Ch.6
5	M 2/20	Sorting in linear time	Ch.8
5	W 2/22	Counting sort, Radix Sort	Ch.8
6	M 2/27	Review	
6	W 3/1	Exam 1	
7	M 3/6	Hash Tables	Ch.11

7	W 3/8	Quicksort	Ch.7, 9
8	M 3/13	Binary Search Trees	Ch.12
8	W 3/15	Red-Black trees	Ch.13
9	M 3/20	Red-Black trees	Ch.13
9	W 3/22	Dynamic Programming	Ch.15
10	M 3/27	Spring Break - No Classes	
10	W 3/29	Spring Break - No Classes	
11	M 4/3	Dynamic Programming	Ch.15
11	W 4/5	Greedy Algorithms	Ch.16
12	M 4/10	Elementary Graph Algorithms, Undirected graph	Ch.22
12	W 4/12	BFS, DFS	Ch.22
13	M 4/17	Review	
13	W 4/19	Exam 2	
14	M 4/24	Directed graph, Topological Sort	Ch.22
14	W 4/26	Strongly connected components	Ch.22
15	M 5/1	Minimum Spanning Tree – Prim’s and Kruskal's Algorithm	Ch.23
15	W 5/3	Minimum Spanning Tree – Prim’s and Kruskal's Algorithm	Ch.23
16	M 5/8	Single Source Shortest Paths: Dijkstra’s Algorithm	Ch.24
16	W 5/10	NP-complete problems	Ch.34
17	M 5/15	NP-complete problems	Ch.34
18	M 5/22	Final Exam 9:45 AM-12:00 PM	