San José State University
College of Science/Department of Computer Science
CS146, Data Structures and Algorithms, Summer, 2021

Course and Contact Information

Instructor:	Nada Attar
Office Location:	MH 218
Telephone:	(408) 924-5108
Email:	nada.attar@sjsu.edu
Office Hours: Zoom Link:	Mo 6:00-6:00 pm | We 5:00-6:00 pm https://sjsu.zoom.us/j/86922029219?pwd=WnRQdXF3SWh1V0l2WlhwNzJjOEsrdz09 Meeting ID: 869 2202 9219 Appointment slots (required for office hours): https://calendar.google.com/calendar/u/0/selfsched?sstoken=UUlPZVdhWkdtZFI4fGRlZmF1bHR8ZmJhMGUzOTQ3YTM5MjIxODRmZTI2YjJjMmE1M2QyNGE
Class Days/Time: Zoom Link:	Mo 7:00-9:00 pm | We 7:00-9:00 pm https://sjsu.zoom.us/j/86434031543?pwd=bElIaW1YSTdHenZpYm5XZEExUUdYZz09 Meeting ID: 864 3403 1543
Prerequisites:	MATH 42, MATH 30, CS 46B, or CS 49J (or equivalent knowledge of Java) (with a grade of "C-" or better in each); Computer Science, Applied and Computational Math or Software Engineering majors only; or instructor consent

Course Description

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

Course Format

Faculty Web Page and MYSJSU Messaging

Course materials such as syllabus, handouts, notes, assignment instructions, etc. can be found on my faculty web page at http://www.sjsu.edu/people/firstname.lastname and/or on Canvas Learning Management System course login website at http://sjsu.instructure.com. You are responsible for regularly checking with the messaging system through MySJSU at http://my.sjsu.edu (or other communication system as indicated by the instructor) to learn of any updates.

Course Objectives

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 (CLO)

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

Required Texts/Readings

Textbook

Recommended reading:

Introduction to Algorithms, 3rd Edition Cormen, Leiserson, Rivest, and Stein
ISBN-10: 0262033844ISBN-13: 978-0262033848MIT Press, 2009
https://www.amazon.com/Introduction-Algorithms-3rd-MIT-Press/dp/0262033844
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.

Programming Language

Java (version 7 or later)

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 %
- Assignments - 40 %

Final exam and quizzes are closed book; final exam is comprehensive. No extra point options in the final. No make-ups exams except in case of verifiable emergency circumstances. Any additional rules and regulations can be applied when taking exams to prevent dishonesty and cheating.

Determination of Grades
The following shows the grading scale to be used to determine the letter grade:

Percentage	Grade
95 and above	A+
92-94	A
90 - 91	A-
87 - 89	B+
83 - 86	B
80 - 82	B-
77 - 79	C+
73 - 76	C
70 - 72	C-
67 - 69	D+
63-66	D
60-62	D-
59 and below	F

Classroom Protocol
The lectures will be on an asynchronous mode (Mode 1). A video for each lecture will be posted at the beginning of each week. The recorded lectures will be shared only with the students enrolled in the course through Canvas. This section has no online lectures, but online office hours, that are in time slot format. The time slot should be reserved before the midnight of the previous day of the 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.

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/" Make sure to review these policies and resources.

Policies
Late homework/projects
All assignments and projects will be due on Thursday of the indicated week at 6:00pm. The assignment will be posted at least a week before the due date to give enough time to work and ask for help during my virtual office hours. Please do not email me few hours before the deadline asking me to help you understand concepts. If I feel that you just start working on your assignment at the due date, I will ignore your emails because I know you won't be able to finish understanding the problem, coding, testing, compiling, and debugging in a few hours. So, please start early and manage your time wisely.
Late Submission:

- 0-6hr -> no penalty
- +6hr -> 60% penalty
- +12hr -> 100% penalty

If you believe an error was made in the grading of your assignments, quizzes, or final exam, you can request a re-grade from the instructor. A request must be sent to the instructor no more than one week after the grades are posted.

Online Discussion Forum Etiquette

Ask clear questions to get better answers. Make sure your question has not been already asked and answered. Read these guidelines (http://superuser.com/help/how-to-ask) when you ask a question. It is fine to share a couple of lines of code but please don't just post your code or share a significant amount of code. Try to ask general questions.

Email Etiquette

I prefer if you use your school email. SJSU uses sjsu-email as an official means of communication. I demand that they use it when contacting me. If you decide to email me from your personal email (ex, gmail, yahoo, etc) that would be fine with me too, but I don't respond to anonymous account names. Also, if you email me during weekend and I am not responding, don't panic! Please give me some time as I am not ignoring your email, but I might be catching on errands. I promise to get back to your email on Monday. This is why I post assignments at least a week before the deadline and encourage you to use our online discission forum.

When you email me, please be on topic, be specific, and state your question clearly so I can assist you in the best way I can. I prefer if you use our online discission forum (private or public message) or Canvas messages. If you send your code due to some error or bug, state which lines of code you think the problem is. Paste the error message you get from the compiler is helpful

Tips to succeed

- This is a challenging, programing-intensive course. It gets more difficult very quickly as the semester progress. It is not about coding, but it teaches you how to write an "efficient" and "elegant" code. So, please do not fall behind.
- Ask for help. I want you to be successful in this course and I love questions. It's important to ask questions during the class. This is your chance to understand the materials before you go home and read it on your own. There is no stupid question and always know that other students will benefit from the questions you ask during class.
- Do not get frustrated. Programming takes patience. It is very common to spend hours doing your HWs/projects. Take short breaks. If you spend hours on one bug or an error, email me and I will help you. Always remember, the more you code, the better you will become.
- Always keep testing your code. After ever few lines of coding, test and compile your code. It is easy to find mistakes in a small scope. And remember to back up your code while you are working on it. I recommend saving it in the cloud.
- As the course is challenging, and required a significant time commitment for most students, ask your instructor for help, ask your friends on Piazza, but there is no excuse for plagiarism.

Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	W 6/2	Syllabus, Course mechanic & Logistics Introduction Review Data Structures (lists, stacks, queues, trees) (Ch.10)
2	M 6/7	Basic algorithms, Insertion sort (Ch.1, Ch.2) Growth of functions- O, Ω, Θ, o, ω (Ch.3)
2	W 6/9	Divide and Conquer technique: Merge Sort, other examples (Ch.4) Solving recurrences, Master Theorem (Ch.4)
3	M 6/14	Heapsort (Ch.6) Priority Queues (Ch.6)
3	W 6/16	Review
4	M 6/21	Exam 1
4	W 6/23	Quicksort, (Ch.7, Ch. 9) Sorting in linear time (Ch.8)
5	M 6/28	Counting sort, Radix Sort (Ch.8) Hash Tables (Ch.11)
5	W 6/30	Binary Search Trees (Ch.12)
6	M 7/5	Red-Black trees (Ch.13)
6	W 7/7	Dynamic Programming (Ch.15) Greedy Algorithms (Ch.16)
7	M 7/12	Exam 2
7	W 7/14	Elementary Graph Algorithms, Undirected graph (Ch.22) BFS, DFS (Ch.22)
8	M 7/19	Directed graph, Topological Sort (Ch.22) Strongly connected components (Ch.22)
8	W 7/21	Minimum Spanning Tree – Prim's and Kruskal's Algorithm (Ch.23)
9	M 7/26	Single Source Shortest Paths: Dijkstra's Algorithm (Ch.24)
9	W 7/28	NP-complete problems (Ch.34)
10	M 8/2	Review - make up class
10	W 8/4	Final Exam