San José State University

Department of Computer Science

CS 146, Data Structures and Algorithms, sections 7 and 8, Fall 2016

Course and Contact Information

Instructor: Jenny Lam
Office location: MacQuarrie Hall 211 (MH 211)
Email: jenny.lam01@sjsu.edu
Office hours: MW 3:00PM-4:00PM or by appointment
Section 7: MW 4:30PM–5:45PM, MH 422
Section 8: MW 10:30AM–11:45AM, MH 422
Prerequisites: MATH 030, MATH 042, CS 049J (or equivalent knowledge of Java), and CS 046B (with a grade of "C-" or better in each); or instructor consent.

Course Description

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.

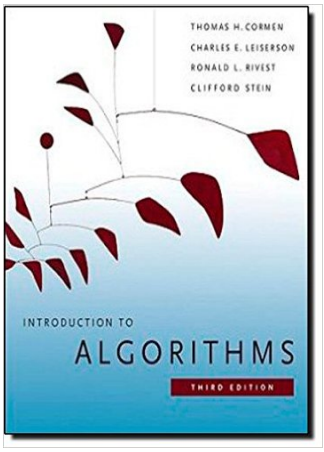
Course Learning Outcomes (CLO)

Upon successful completion of this course, students will 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



Cormen, Leiserson, Rivest and Stein, Introduction to Algorithms, 3rd Edition
ISBN-10: 0262033844
ISBN-13: 978-0262033848
MIT Press, 2009
Errata (bug reports) at http://www.cs.dartmouth.edu/~thc/clrs-bugs/bugs-3e.php

Other technology requirements / equipment / material

Java compiler (version 8)

Course Requirements and Assignments

The grade in this course will be weighted as follows:

Participation: although there is no grade incentive, students are expected to attend every class and actively participate.

Assignments: (10%) assignments are assigned roughly every week and will consist of a mix of programming and written problems. Students may discuss approaches to solving problems, but all work that is turned in must be done individually. The two lowest grades will be dropped. Late work is not accepted.

Exams: there will be three in-class exams (20% each) and a comprehensive final exam (30%), to be taken individually. Missed exams cannot be made up except for reasons of illness as certified by a doctor, or documentable extreme emergency. Makeup exams may be oral.

Extra credit problems may be given as part of assignments and exams, the scoring of which may not result in changing the relative weights between assignments and exams.

Letter grade: the final numerical score will translate to the course letter grade according to the following scale:

Percentage	Letter grade
[90-100)	A
[75-90)	B
[60-75)	C
[50-60)	D
[0-50)	F

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/

CS 146, Data Structures and Algorithms, sections 7 and 8, Fall 2016, Course Schedule

This schedule is tentative and subject to change during the semester. Changes will be reflected on the course webpage as well as announced during lecture. Students should consult the schedule on the course webpage at http://jennylam.cc/courses/146-f16 regularly for the most up-to-date information.

Lecture	Date	Topic
1	W 8/24	Introduction
2	M 8/29	Recursion: divide-and conquer
3	W 8/31	Recursion: divide-and-conquer
	M 9/5	*Labor day*
4	W 9/7	Recursion: backtracking
5	M 9/12	Recursion: dynamic programming
6	W 9/14	Recursion: dynamic programming
7	M 9/19	Data structures: stacks and finding the convex hull
8	W 9/21	Data structures: priority queues, heaps, selection and heapsort
9	M 9/26	Data structures: dictionaries, BSTs
10	W 9/28	Data structures: red-black trees
11	M 10/3	**Exam 1**
12	W 10/5	Randomization: hashing
13	M 10/10	Randomization: hashing
14	W 10/12	Randomization: randomized quicksort
15	M 10/17	Randomization: treaps, skiplists
16	W 10/19	Graphs: DFS, topological sort, strongly connected components
17	M 10/24	Graphs: shortest path problem, Bellman-Ford
18	W 10/26	Graphs: BFS, Dijkstra
19	M 10/31	Graphs: minimum spanning trees

Lecture	Date	Topic
20	W 11/2	**Exam 2**
21	M 11/7	Hardness: sorting and search lower bounds
22	W 11/9	Hardness: radix and bucket sort
23	M 11/14	Hardness: NP-completeness
24	W 11/16	Hardness: NP-completeness
25	M 11/21	Hardness: approximation algorithms
26	W 11/23	Hardness: approximation algorithms
27	M 11/28	**Exam 3**
29	W 11/30	Recursion: fast exponential-time algorithms
30	M 12/5	Data structures: segment intersection, dynamic order statistics
31	W 12/7	Randomization: bloom filter, string matching, tries
32	M 12/12	Graphs: all-pairs shortest path
		Section 7 Final exam F 12/16 2:45PM–5:00PM, MH 422
		Section 8 Final exam W 12/14 9:45AM–12:00PM, MH 422