CS 146
Data Structures and Algorithms

GREEN SHEET
Summer Semester 2015

Department of Computer Science
San Jose State University
Instructor: Ron Mak

Class:	TuTh noon-1:55 pm, MacQuarrie Hall, room 422
Office hours:	TuTh 2:00-3:00 PM
Office location:	MacQuarrie Hall, room 413
E-mail:	ron.mak@sjsu.edu

Catalog 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." *3 units*

Course Goal

To examine various ways to represent data used by programs and to compare these representations in terms of their memory requirements and the resulting program execution times.

Course Objectives

- Ensure that students are familiar with ways to implement elementary data structures and their associated algorithms.
- Introduce students to the implementation of more complex data structures and their associated algorithms.
- Acquaint students with advanced sorting techniques (radix sort, heapsort, mergesort, quicksort).
- Teach students how to determine the time complexity of algorithms.
- Introduce students to algorithm design techniques.

Student Learning Outcomes

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

- Implement lists, stacks, queues, search trees, heaps, union-find ADT, and graphs and use these data structures in programs they design
- Prove basic properties of trees and graphs
- Perform breadth-first search and depth-first search on directed as well as undirected graphs
- Use advanced sorting techniques (radix sort, heapsort, mergesort, quicksort)
- Determine the running time of an algorithm in terms of asymptotic notation
- Solve recurrence relations representing the running time of an algorithm designed using a divide-and-conquer strategy
- 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
- Comprehend algorithms designed using greedy, divide-and-conquer, and dynamic programming techniques

Prerequisites

Math 030	Calculus I
Math 042	Discrete Mathematics
CS 049J	Programming in Java *or equivalent knowledge of Java*
CS 046B	Introduction to Data Structures

A grade C- or better, or instructor's consent. **The Department of Computer Science strictly enforces prerequisites.** If you are not already pre-enrolled, you must come to the first class meeting and pick up an Add Form from the instructor. If applicable, show the instructor your card that indicates you're a graduating senior. It will be the instructor's and the department's decision whether or not to send you an add code by email.

The instructor may drop any student who does not show up during the first two class meetings.

Material assumed from prerequisite courses

Math 30 (Calculus I)

- Limits and integration

Math 42 (Discrete Mathematics)

- Proof by induction, proof by contradiction
- Sequences and summations
- Equivalence relations, equivalence classes
- Recursive definition, recurrence relations
- Permutations, r-combinations, simple counting and probability

CS 46B (Introduction to Data Structures) and CS49J (Programming In Java)

- Linked lists, stacks, queues
- The Comparator interface
- Concepts of an iterator and iteration over a collection
- Binary search, binary search trees, hashing
- Elementary sorting (insertion, selection, and bubble sorts)
- Big-Oh notation
- Javadoc tags @param, @return, @throws

Laptops and lab time

Bring your laptops to class — we'll try to set aside some "lab" time for all the students to get together and share tips and accomplishments. This class will move rapidly and cover a lot of material. Class attendance and participation are very important.

Required text

Title:	**Data Structures and Algorithm Analysis in Java,** *3rd edition*
Author:	Mark Allen Weiss
Publisher:	Pearson
ISBN-13:	978-0-13-257627-7
Errata:	http://users.cis.fiu.edu/~weiss/dsaajava3/errata.html

Schedule

Subject to change!

Weeks	Dates	Topics and activities	Chapters
1	June 2 June 4	Introduction From arrays to generics Algorithm analysis	1, 2
2	June 9 June 11	Lists	3
3	June 16 June 18	Stacks and queues Trees	3 4
4	June 23 June 25	Trees	4
5	June 30 ***July 2***	Priority queues (heaps) ***Midterm***, July 2	6
6	July 7 July 9	Sorting	7
7	July 14 July 16	Sorting Disjoint set class	7 8
8	July 21 July 23	Graphs Hashing	9 5
9	July 28 July 30	String pattern matching Algorithm design techniques	 10
10	Aug 4 Aug 6	NP-completeness ***Final exam***, August 6	9

Programming assignments

There will be several programming assignments requiring you to use the major data structures and algorithms covered in the course. Programs must be appropriately documented via javadoc comments and should adhere to the coding style posted on the CS Department web page: http://www.cs.sjsu.edu/web_mater/java_code.html. If you work together with another student on an assignment, both of you will receive the same score.

Each assignment is worth a maximum of 100 points. Late assignments will lose 20 points and an additional 20 points for each 24 hours after the due date.

Exams

The midterms and final examinations will be open book, notes, and laptops. Instant messaging, e-mails, texting, tweeting, or other communication with anyone else during the exams will be strictly forbidden.

Class grade

Your individual class grade will be weighted as follows:

65%	Assignments
15%	Midterm exam
20%	Final exam

Each project and exam will be scored (given points) but not assigned a letter grade. The mean score and standard deviation will be announced after each project and exam.

Final individual class letter grades will be assigned based on the class curve. Your final class grade can be adjusted up or down depending on your level and quality of participation on your project team as determined by the project tracking tools and your team members' assessments of your performance.

Classroom protocol

It is important for each student to attend classes and to participate. Cell phones in silent mode, please.

Academic integrity

Your own commitment to learning, as evidenced by your enrollment at SJSU, and the University's Academic Integrity Policy requires you to be honest in all your academic course work. Faculty members are required to report all infractions to the Office of Student Conduct and Ethical Development. The policy on academic integrity can be found at http://www.sjsu.edu/studentaffairs/.

University policies

If you need course adaptations or accommodations because of a disability, or if you need special arrangements in case the building must be evacuated, please make an appointment with me as soon as possible, or see me during office hours. Presidential Directive 97-03 requires that students with disabilities requesting accommodations register with the SJSU Disability Resource Center to establish a record of their disability.

Please familiarize yourself with SJSU policies and procedures at http://info.sjsu.edu/static/catalog/policies.html, particularly the add/drop policy. It is your responsibility to know and observe these policies. However, if there is something about a policy that you don't understand, please feel free to ask! You can also find answers to many questions at the Academic Advising and Retention Services web site.

The Summer Session Calendar contains important dates and deadlines: http://www.sjsu.edu/summer/.

Recording lectures

Common courtesy and professional behavior dictate that you notify someone when you are recording him/her. You must obtain the instructor's permission to make audio or video recordings in this class. Such permission allows the recordings to be used for your private, study purposes only. The recordings are the intellectual property of the instructor; you have not been given any rights to reproduce or distribute the material.

Course material developed by the instructor is the intellectual property of the instructor and cannot be shared publicly without his/her approval. You may not publicly share or upload instructor generated material for this course such as exam questions, lecture notes, or homework solutions without instructor consent.