

**San José State University**  
**Computer Science Department**  
**CS146, Sections 1 and 2, Data Structures and Algorithms, Fall 2015**

**Course and Contact Information**

<b>Instructor:</b>	Aikaterini Potika
<b>Office Location:</b>	MacQuarrie Hall 215
<b>Telephone:</b>	(408) (9245134)
<b>Email:</b>	katerina.potika@sjsu.edu
<b>Office Hours:</b>	M 12:00-12:30pm, 3-4pm, W 12:30-1pm or by appointment
<b>Class Days/Time:</b>	Section 1: MW 9-10:15am and Section 2: MW 10:30-11:45am
<b>Classroom:</b>	MacQuarrie Hall 422
<b>Prerequisites:</b>	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**

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.

Topics covered will include advanced tree structures, directed and undirected graphs, advanced searching and sorting techniques, priority queues and heaps, dictionaries, and mathematical tools and techniques (recursion, recurrence relations) useful for the design and analysis of data structures and algorithms.

**Course Objectives**

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

**Student Learning Outcomes**

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

- 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
- 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 (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
- 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
- Understand algorithms designed using greedy, divide-and-conquer, and dynamic programming techniques

### **Faculty Web Page**

Course materials such as syllabus, handouts, notes, assignment instructions, etc. can be found on the Canvas learning management system course website (<https://idp01.sjsu.edu/idp/Authn/UserPassword>). You are responsible for regularly checking with the messaging system.

### **Required Texts/Readings**

#### **Textbook**

Cormen, Leiserson, Rivest and Stein, Introduction to Algorithms, 3rd Edition

ISBN-10: 0262033844

ISBN-13: 978-0262033848

MIT Press, 2009

You can find errata (bug reports) for the book <http://www.cs.dartmouth.edu/~thc/clrs-bugs/bugs-3e.php>.

#### **Other Readings**

- Horstmann and Cornell, Core Java, Vol. I, Ninth edition, Prentice Hall, 2013.
- Kleinberg and Tardos, Algorithm Design, First edition, Addison Wesley, 2005.
- Dasgupta, Papadimitriou and Vazirani, Algorithms, McGraw-Hill, 2006.
- Handouts (through Canvas)

#### **Other equipment / material requirements**

Java Compiler (version 7 or later).

### **Assignments and Grading Policy**

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 S12-3](http://www.sjsu.edu/senate/docs/S12-3.pdf) at <http://www.sjsu.edu/senate/docs/S12-3.pdf>.

Final Grade:

15% Programming assignments

5% Quizzes

40% Midterms (20% each)

40% Final

Exams are closed book, final exam is comprehensive.

No make-ups exams except in case of verifiable emergency circumstances

A+	A	A-	>90
B+	B	B-	>74
C+	C	C-	>59
D+	D	D-	>44
F			<40

**Homework assignments:** individual, regularly assigned, not graded, will include written problem assignments, and perhaps some online exercises. Solutions will be not posted. The homework is a tool for you to learn the material and prepare you for the exams.

**Reading assignments:** Reading assignments will regularly be for the next class (see schedule).

**Quizzes:** Unannounced quizzes (at least 4) may be given during class, each taking about 5 minutes total. These will generally be problems from the reading assignment and/or the homework.

**Programming assignments:** Four programming assignments will be assigned. Programming assignments are done individually, unless otherwise specified. They can be discussed, but should be implemented individually. More information will be given at the time of the first programming assignment. Never use any code you find on the web, unless it is given by me. Penalty for late submission 5% for every 3 days up to 9 days, after that no submission will be accepted. Never email me your assignments.

Additionally, you may be disqualified from getting an A/B/C/D unless you get at least 4/3/2/1 program(s) to work, respectively.

**Midterm exams:** There will be two Midterm exams during the semester, 20% each.

**Final exam:**

The final exam will be worth 40% of your grade.

The exams will contain multiple choice questions, short answer questions and questions that require pseudocode and/or computations.

Students must obtain >50% in each component of the course (programming & quizzes & written exams) in order to be eligible for a grade of C- or better.

Note that “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 F13-1](http://www.sjsu.edu/senate/docs/F13-1.pdf) at <http://www.sjsu.edu/senate/docs/F13-1.pdf> for more details.

**Classroom Protocol**

Attendance is highly recommended. Please avoid disturbing the class: turn-off cell phones (or put them on vibrate mode), no text messaging in the class or the exams, **no taking pictures and video**, avoid coming late. The course material cannot be shared publicly without my approval. You may not publicly share or upload material for this course such as exam questions, lecture notes, or solutions without my consent (see [University Policy S12-7](http://www.sjsu.edu/senate/docs/S12-7.pdf), <http://www.sjsu.edu/senate/docs/S12-7.pdf>).

NOTE that [University policy F69-24](http://www.sjsu.edu/senate/docs/F69-24.pdf) at <http://www.sjsu.edu/senate/docs/F69-24.pdf> states that “Students should attend all meetings of their classes, not only because they are responsible for material discussed therein, but because active participation is frequently essential to insure maximum benefit for all members of the class. Attendance per se shall not be used as a criterion for grading.”

## Dropping and Adding

Students are responsible for understanding the policies and procedures about add/drop, grade forgiveness, etc. Refer to the current semester's [Catalog Policies](http://info.sjsu.edu/static/catalog/policies.html) section at <http://info.sjsu.edu/static/catalog/policies.html>. Add/drop deadlines can be found on the current academic year calendars document on the [Academic Calendars webpage](http://www.sjsu.edu/provost/services/academic_calendars/) at [http://www.sjsu.edu/provost/services/academic\\_calendars/](http://www.sjsu.edu/provost/services/academic_calendars/). The [Late Drop Policy](http://www.sjsu.edu/aars/policies/latedrops/policy/) is available at <http://www.sjsu.edu/aars/policies/latedrops/policy/>. Students should be aware of the current deadlines and penalties for dropping classes. Information about the latest changes and news is available at the [Advising Hub](http://www.sjsu.edu/advising/) at <http://www.sjsu.edu/advising/>.

## Academic integrity

Your commitment, as a student, to learning is evidenced by your enrollment at San Jose State University. The [University Academic Integrity Policy S07-2](http://www.sjsu.edu/senate/docs/S07-2.pdf) at <http://www.sjsu.edu/senate/docs/S07-2.pdf> 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 [Student Conduct and Ethical Development website](http://www.sjsu.edu/studentconduct/) is available at <http://www.sjsu.edu/studentconduct/>.

## Campus Policy in Compliance with the American Disabilities Act

If you need course adaptations or accommodations because of a disability, or if you need to make 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](http://www.sjsu.edu/president/docs/directives/PD_1997-03.pdf) at [http://www.sjsu.edu/president/docs/directives/PD\\_1997-03.pdf](http://www.sjsu.edu/president/docs/directives/PD_1997-03.pdf) requires that students with disabilities requesting accommodations must register with the [Accessible Education Center](http://www.sjsu.edu/aec) (AEC) at <http://www.sjsu.edu/aec> to establish a record of their disability.

## CS146: Data Structures and Algorithms, Fall 2015

The schedule is subject to change with fair notice.

### Course Schedule

Lectures	Date	Topic	Chapter
1	8/24	Introduction: Algorithms & Computers	Ch 1 & Appendix A
2	8/26	Insertion Sort - Review	Ch 2.1
3	8/31	Divide and Conquer, Merge Sort	Ch 2.2, 2.3
4	9/2	Growth of functions- $O$ , $\Omega$ , $\Theta$ , $o$ , $\omega$	Ch 3
	9/7	<i>Lab day Closed</i>	
5	9/9	Solving recurrences - Master Theorem	Ch 4.3-4.5
6	9/14	Master Theorem - Intro to Heaps	Ch 6.1

7	9/16	Heapsort, Priority Queues	Ch 6
8	9/21	Quicksort	Ch 7 (not 7.3)
9	9/23	Analysis of Quicksort	Ch 7 (not 7.3)
10	9/28	Sorting in linear time, Counting sort, Radix Sort	Ch 8
11	9/30	Order statistics - Selection Algorithm	Ch 9 (not 9.2)
12	10/5	Data Structures & Hashing	Ch 10 & 11
	10/7	Midterm 1	
13	10/12	Union Find, Dynamic sets Binary Search Trees	Ch 12
14	10/14	Red Black trees	Ch 13
15	10/19	B-trees	Ch 18
16	10/21	Greedy technique	Ch 16
17	10/26	Dynamic Programming	Ch 15
18	10/28	Graphs BFS, DFS	Ch 22.1-2, Appendix B.1, B.4-5
19	11/2	Topological sort	Ch 22.3-5
	11/4	Midterm 2	
20	11/9	Minimum Spanning Tree – Prim's and Kruskal's Algorithm Data Structures for Disjoint Sets	Ch 23, Ch 21
	11/11	<i>Veterans Day Closed</i>	
21	11/16	Single Source Shortest Paths: Bellman Ford, Dijkstra's Algorithm	Ch 24
22	11/18	Single Source Shortest Paths: Bellman Ford, Dijkstra's Algorithm	Ch 24

23	11/23	All-Pairs Shortest Paths: Floyd-Warshall	Ch 25.1-2
24	11/25	Dynamic Programming - Knapsack	Ch 15
25	11/30	NP-completeness	Ch 34.1-2
26	12/2	Reductions	Ch. 34.3-4
27	12/7	NP-complete problems	Ch. 34.5
	12/15-12/16	<b>Final exam</b> Section1, 12/16, 7:15-9:30AM Section2, 12/15 9:45-12:00PM	