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 %
Quizzes - 10 %
Assignments - 30 %

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:

Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	Tu 8/23	Introduction: syllabus, Course mechanic & Logistics
2	Th 8/25	Review Data Structures (lists, stacks, queues, trees) (Ch.10)
2	Tu 8/30	Basic algorithms, Insertion sort (Ch.1, Ch.2) Growth of functions- O, Ω, Θ, o, ω (Ch.3)
3	Th 9/01	Divide and Conquer technique: Merge Sort, other examples (Ch.4)
3	Tu 9/06	Solving recurrences (Ch.4)
4	Th 9/08	Master Theorem (Ch.4)
4	Tu 9/13	Heapsort,