Computer Engineering 12
Abstract Data Types and Data Structures

Spring 2009
Mondays, Wednesdays, and Fridays
10:30 am – 11:35 am and 1:00 pm – 2:05 pm

Instructor

Instructor:	Darren Atkinson
E-mail:	datkinson@scu.edu
Office hours:	Mondays, 2:30–4:30 pm
Office:	EC 245
Website:	http://www.cse.scu.edu/~atkinson/teaching/sp09/012/

Teaching Assistants

Teaching assistants:	Deepti Mande and Christian Ochoa
E-mails:	dmande@scu.edu and cochoa@scu.edu
Lab hours:	Tuesdays, Wednesdays, and Thursdays, 2:15–5:00 pm

Textbooks

Required:	Gilberg and Forouzan, *Data Structures – A Pseudocode Approach with C*, Brooks/Cole.
Recommended:	Kernighan and Ritchie, *The C Programming Language*, 2nd edition, Prentice Hall.

Grading

Lab attendance:	5%	(each and every week)
Programming projects:	15%	(4/10, 4/24, 5/8, 5/22, and 6/5)
Midterm exams:	40%	(4/17 and 5/15)
Final exam:	40%	(6/10)

Overview

Abstract data types:	sets, lists (including stacks and queues), priority queues, and maps (dictionaries)
Data structures:	arrays, hash tables, linked lists, trees (including search trees), heaps, and graphs
Algorithms:	searching and sorting (selection-based, insertion-based, and exchange-based)

Course Objectives

Students will be able . . .

1. To understand abstract data types, data structures, and algorithms.
2. To develop programming skills, including the basics of software development such as the use of existing code and the compile-debug-execute cycle.
3. To use mathematics to reinforce importance of mathematical skills.

Learning Outcomes

Students will . . .

1. Discuss the relationships between an abstract data type and a data structure.
2. Separate an abstract data type into an interface and an implementation.
3. Compare and contrast simple container data types (lists, sets, maps, priority queues).
4. Compare and contrast classic data structures (arrays, hash tables, linked-lists, trees, graphs).
5. Implement the classic data structures in a low-level language such as C.
6. Know the average-case and worst-case running times for common operations (insertion, deletion, retrieval, minimum, maximum) on the classic data structures.
7. Compare and contrast classic searching and sorting algorithms.

Policies

Disability Accommodation Policy

To request academic accommodations for a disability, students must contact Disabilities Resources located on the second floor of Benson. Phone numbers are (408) 554-4111; TTY (408) 554-5445. Students must register and provide documentation of a disability to Disabilities Resources prior to receiving academic accommodations.

Academic Integrity Policy

The University is committed to academic excellence and integrity. Students are expected to do their own work and to cite any sources they use. A student who is guilty of a dishonest act in an examination, paper, or other work required for a course, or who assists others in such an act, may, at the discretion of the instructor, receive a grade of F for the course.

In addition, a student found guilty of a dishonest act may be subject to sanctions up to and including dismissal from the University as a result of the student judicial process as described in the *Community Handbook*.

A student who violates copyright laws, including those covering the copying of software programs, or who knowingly alters official academic records from this or any other institution is subject to similar disciplinary action.