
Lamar University

COSC 2336-01, Spring 2005

Data Structures

Instructor: Chung-Chih Li, Ph.D.

Office: Mase 69, Tel:(409) 880-8748

URL: <http://hal.lamar.edu/~licc>

E-mail: licc@hal.lamar.edu

Office Hours: Tue & Thu 2:00 ~ 3:30 PM or by appointment

Classroom and meeting time:

Mase 106, MWF 11:15 AM ~ 12:05 PM
(Attendance will be taken impulsively)

HomePages of the course: <http://hal.lamar.edu/~licc/cosc2336>

From there, you may find important information about assignments, assignment data, due dates, sample programs, or announcements. **Note:** *An announcement made in the class will be considered as an official one, since I may not be able to update every announcement.*

Course Description and Purposes:

This is the last course of the three-course-series, COSC 1336-1337-2336. I call it the trilogy of CS major, which is required for every student in CS major.

In this course, we assume that students already have certain maturity of programming skill, especially C++. Nevertheless, we will briefly review the basic yet necessary features of C++ at the beginning of the semester. A better data structure is meant to speed up our programs. Thus, we will scratch the principle of analyzing the complexity of programs in order to give students a motivation of learning data structures. Afterwards, we will learn some standard data structures such as *linked list*, *stacks*, *queues*, *trees*, and *graphs*. Also, we will learn some complicate programming skills to solve problems and handle data. These skills includes *recursion*, *sorting*, and *hashing*.

Prerequisites:

COSC 1336 and COSC 1337 (each with grade B or better).

Textbooks:

Data Structures and Algorithms in C++, by Adam Drozdek, Thomson Course Technology, 3rd Edition 2005

Reference:

Absolute C++, by Walter J. Savitch, Addition Wesley Publishers, 2002

Examinations: (300 points) Two midterms and one Final Exam (100 points for each test)

- Unless announced otherwise, all tests are accumulative, closed book, and indispensable. No makeup test will be given unless a documented absence is authorized by the university.
- Every student is allowed to bring a self-prepared crib sheet to the test. You can **write** down anything on both sides of **one** letter-sized paper. No circulation during the test.

Midterm I	100 points	Feb. 18, 6 th week's Friday
Midterm II	100 points	Mar. 21, 11 th week's Monday
Final Exam	100 points	May 4, Wednesday, 8:00 AM ~ 10:30 AM

Assignments: (240 ~ 280 points) About 6 or 7 programming assignments will be given. Students are encouraged to discuss assignments and help each other. However, this does not mean that you can either entirely or partially copy or modify someone else's works.

Any form and any degree of plagiarism will receive 0 point.

Late works will be graded with penalty: -10 points per day after the due date.

Attendance: (50 points) Attendances will be taken impulsively.

Pop quizzes: (50 points)

About 5 (or more) pop quizzes will be given impulsively. Each quiz carries 10 points towards students' final scores. The coverage of every quiz is also accumulative, including the materials covered in the class right before the quiz. A typical quiz takes about 10 minutes. No makeup quiz will be given if missed. If you miss a quiz due to a university authorized absence, we will use the average of your rest quizzes; otherwise, you get a 0 for the absent quiz.

Academic Honesty:

Cheating, plagiarism, collusion, abuse of resource materials, and their consequences are defined and described under the section of Academic Affairs in the *Student Handbook*.

Students giving away academic works for assignment offered for credit to other students working on the same assignment will be considered as guilty as academic dishonesty, and will receive the same penalty.

Grading Policy:

Considering 650 points the perfect score, your grade is based on the scheme shown in the table.

I do not curve!!

Points	Grade	
540 ~ 650	A	Excellent
420 ~ 539	B	Good
300 ~ 419	C	Satisfactory
200 ~ 299	D	Passing
0 ~ 199	F	Failure

Tentative Topics and Schedule:

Week	Topics	Reading
1: Jan. 12	(Jan. 12, first class day), programming preliminaries	Syllabus
2: Jan. 17	(Jan. 17, no class), C++ programming preliminaries	Chapter 1
3: Jan. 24	Algorithm analysis, asymptotic notations	Chapter 2
4: Jan. 31	Singly, doubly, and circularly linked lists, skip lists	3.1 ~ 3.5
5: Feb. 7	Linked lists in applications, stacks, queues, priority queues	3.6, 4.1, 4.2, 4.3
6: Feb. 14	C++ STL for linked list, stacks, and queues (Midterm 1, Feb. 18, Friday)	3.7, 3.8, 4.5, 4.6
7: Feb. 21	(Feb. 23, last day to drop without penalty) Recursion	Chapter 5
8: Feb. 28	Binary trees: implementation, searching, traversal, maintenance	6.1 ~ 6.6
9: Mar. 7	Balancing a tree: AVL trees, heaps, polish notations	6.7 ~ 6.9
10: Mar. 14	Spring break, no class	
11: Mar. 21	(Midterm 2, Mar. 21, Monday) B-Trees, (Mar. 25, good Friday, no class)	7.1
12: Mar. 28	Graph, shortest paths, cycles, spanning trees, topological sort	8.1 ~ 8.7
13: Apr. 4	Maximum flows of networks, matching, Eulerian and Hamiltonian Graphs, coloring	8.8 ~ 8.11
14: Apr. 11	NP-complete problems, sorting basics	8.12, 9.1
15: Apr. 18	Decision trees, efficient sorting algorithms	9.2, 9.3
16: Apr. 25	Hashing functions and collision resolution	10.1, 10.2
17: May 2	(May 3, last class day) Final Exam, May 4, Wed. 8:00 ~ 10:30 AM	