

CSCI-223

Data Structures and Algorithms

Spring 2008

Lecture MWF 9:00-9:50 Thompson 316

Instructor : Dr. M.A. Francel
e-mail : francelm@citadel.edu
Office : 226 Thompson
Phone : 953-6987 (w), 958-9491 (h)

Office Hours :

11:00 -11:50, 1:30 – 3:00 Mon, Wed, Fri
or by appointment or phone

Feel free to call me at home between 8 am and 11 pm for help with your work any day of the week, including weekends.

Text :

Data Structures & Algorithms in JAVA (4th Edition)
Michael Goodrich and Roberto Tamassis

Course Description :

Formal specification and implementation of abstract data types and analysis of algorithms. Topics include list and set representation methods, sorting, trees and graphs. Data structures used include arrays, queues, binary trees, hash tables, priority queues, search trees, and balanced trees.

Lecture : 3 hours;

Course Prerequisites:

CSCI 202 (or a working knowledge of a object oriented programming language)
and MATH 206

Course objectives :

- Introduce the classical abstract data types (ADTs)
- Present criteria for deciding how the various data structures can be applied in a variety of problems of recurring interest in Computer Science
- Practice describing, evaluating, and implementing ADTs.
- Introduce basic concepts of algorithm analysis.

Course Outcomes :

Upon successfully completing this course, a student can

- Write programs of moderate to high complexity using classic ADTs such as List, Stack, Queue, Binary Tree, Heap, Hashtable, Graph. *Assessment: programming projects, homework, exams.*
- Choose an appropriate ADT and ADT implementation, and give a rationale for the choice, when given a problem description.
- Perform Big-O analysis of algorithms.

Class Attendance :

Class attendance is expected. However, if for some reason you do miss class it is up to you to (1) let me know that you will be missing class BEFORE that class period, (2) get any assignments due that class period to me BEFORE class, (2) learn the material presented in class that day BEFORE the next class, and (3) prepare any assignments given that day in class for the due date.

There is a distinct difference between missing class and missing the material presented. The first is sometimes necessary, the second is unacceptable.

As prescribed by the College catalog, students who miss 20% of the class meetings (nine classes) will be dropped from the course and will receive a grade of F in the course.

The student class absence system will serve as the official record for class attendance.

Class Discussion :

It is expected that everyone will take an active role in class. This means answering questions and contributing to the development of algorithm designs. Students who sleep in class will be treated as if ABSENT from class. This means being reported as absent on the class roll for the day.

Programs and Homework :

Programs and homework will be assigned, collected, and graded. Both are an important part of the course. You are expected to devote appropriate time outside of class to the completion of this work. DO NOT get any outside help on any graded work except from me. Homework will NOT be accepted late for any reason. If you are unable to come to class on the day a homework assignment is due, it is up to you to see that homework is passed in BEFORE class begins. Programs will be accepted up to a week late. Each day they are late five points will be deducted from the final grade.

Programming concepts build on each other. One can not understand a concept unless one understands all of the previous ones. Thus, besides working on programs and homework outside of class, it is expected that each student will read the text and regularly review his or her lecture notes.

Quizzes:

Quizzes over the class readings and lectures will be given regularly. NO make-ups will be given. Over the term all types of test questions will be used for the quizzes including:
essay, short answer, true/false, multiple choice, and matching.

Exam Information and Dates :

All tests will be in-class, closed book exams. Exams will cover all material since the last test. If you are a student who has been approved by Dr. Zaremba for testing accommodations, I will need to see your letter of accommodation at least one week before the first test.

Test 1:	FEB 06
Test 2:	MAR 07
Test 3:	APR 11

Final Exam Date:

THURS APR 24 1 pm

Assessment :

Course evaluation is performance based.

- 30% labs and programs
- 10% homework and quizzes
- 30% tests
- 30% final exam

Grading Scale:

- | | |
|---|-----------|
| A | 100 - 90% |
| B | 89 - 80% |
| C | 79 - 70% |
| D | 69 - 55% |
| F | below 55% |

CSCI-223

Data Structures and Algorithms

Spring 2008

Unit Outline and Reading List

All text reading refers to: *Data Structures & Algorithms in Java*
(4th Edition)
Michael Goodrich and Roberto Tamassia
John Wiley & Sons

Unit 1: An introduction to data structures

TOPICS:

- An abstract point of view
 - Data types -
 - Definition (values+operations), examples
 - Operation arity
- A coding point of view
 - Implementation of a data type
 - Built-in
 - Enum
 - Class
 - Constructors, methods, parameters
 - Recursion

READING:

Review of CSCI-201 and CSCI-202 text and class notes

Unit 2: Algorithms and complexity

TOPICS:

- Definitions, etc.
- Complexity measures
 - Calculating complexity
 - Standard algorithm complexity

READING:

Class handout, Chapter 4

Unit 3: Hashing

TOPICS:

What is a hash table?
What is a hash function
 Function, 1-1, onto
Handling collisions
 Linear probing, Chaining, Rehashing
Implementation, complexity analysis
Applications

READING:

Chapter 9
Review of linked-list material

Unit 4: Binary trees

TOPICS:

Heaps
 Definition of
 Inserting, deleting, displaying
 Implementation
 Recursion and not, complexity analysis
 Applications
 Sorting, others
Binary Search Trees
 Definition of
 Inserting, deleting, searching, displaying
 Implementation
 Recursion and not
 Linked-list and not
 Complexity analysis
 Applications
 Sorting, others

READING:

Chapters 8 and 10
Review of linked-list material
Review of recursion material

Unit 5: Text Processing

TOPICS:

- Pattern Matching Algorithms
 - Brute Force
 - The Boyer-Moore Algorithm
 - The Knuth-Morris-Pratt Algorithm
- Text Compression
 - Huffman Coding
 - Greedy Methods

READING:

Chapter 12

Unit 6: Graphs

TOPICS:

- The Graph as an Abstract Data Type
- Data Structures for Graphs
- Graph Traversals
- Shortest Paths
- Minimum Spanning Trees

READING:

Chapter 13

Unit 7: Back to Lists (as time permits)

TOPICS:

- Queues and Priority Queues
- Implementation
 - Array, linked, circular
 - Complexity analysis
- Applications
 - Event-driven simulation

READING:

Chapter 8