CSCI 244 – Fall 2012
Data Structures and Analysis of Algorithms

Ben Coleman
coleman@cs.moravian.edu
PPHAC 214
Office Phone: 610-625-7781

Office Hours: MWF 2:20 - 3:00
T 9:00 - 12:00
or by appointment

Course Description

An examination of issues dealing with static and dynamic aggregates of data. Topics covered include logical characteristics of various data organizations, storage structures implementing structured data, design and implementation of algorithms to manipulate such storage structures, and classical applications of data structures. Representative data structures include stacks, queues, ordered trees, binary trees, and graphs. Both contiguous and linked storage implementations are considered and performance issues discussed.

Course Goals

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

- Describe the strengths and limitations of linear data structures, trees, graphs, and hash tables.
- Select appropriate data structures for a specified problem.
- Describe classic sorting techniques.
- Analyze sequential and parallel algorithms using appropriate mathematical notation.
- Describe common patterns in parallel algorithms.

Required Texts

In addition to the following text, supplementary readings will be given periodically during the semester.

- *Data Structures and the Java Collections Framework, Third Edition* by William Collins

You should expect to spend about an hour before each class session working through the readings. This means reading the text for detail, studying the syntax for new language features, and working to learn vocabulary – not just skimming through the material before class.

Graded Material

- **Homework** – The goal of homework problems is for you to practice using the current course content and to explore the topics in more detail. Problems will be assigned nearly every class session and will be due the next class.
- **Tests** – Two tests will be given during the semester on Wednesday, October 10 and Wednesday, November 28. You may only re-schedule a test for college-approved absences or documented illness. In either case, you must contact me *before* the beginning of the test.
- **Programming Assignments** – Various programming assignments will be assigned during the semester. In some instances, you will simply implement a small stand-alone program. At other times, a sequence of assignments will build upon each other to produce a final program. All programming assignments will be graded based on correctness and the quality of testing.
- **Final** – The final will be cumulative and will be given in-class on Tuesday, December 11 at 1:30 p.m. Any change to the final exam schedule must be approved by both me and the dean of students.

Grade Determination

- (40%) Homework
- (25%) Tests
- (20%) Programming Assignments
- (15%) Final

All grades will be calculated on the standard scale using pluses and minuses.

Course Policies

- **Late Policy** – I understand that life sometimes gets in the way of getting work done. Consequently, late assignments will be accepted without penalty in the class after the assignment was due. However, this policy should not be used as a crutch, and if you frequently use it I will deduct from your grade. After the next class session, late work will not be accepted unless there are exceptional circumstances.
- **Extensions** – In a similar vein, I am generous with extensions on work if you approach me *before* the day the assignment is due.
- **Absences** – Your attendance is expected at each class meeting, but I understand that students occasionally get sick, have obligations outside Moravian, and even over sleep. If you do miss class, please send me an email explaining your absence – preferably before the class session. Regardless of your reason for missing class, you are responsible for the contents of reading assignments, handouts, class activities, and class email.
- **Academic Honesty** – Except on tests, you are *encouraged* to discuss the material and work with other students in the course. Specifically, on homework and programming assignments you may discuss any portion of the assignment with your fellow students. This policy does

not allow you to copy another student's work verbatim – you must produce your own code or write-up of the material. Work together to learn the concepts, but keep in mind that you are ultimately responsible for the material on the tests.

- **Disabilities** – Students who wish to request accommodations in this class for a disability should contact the assistant director of Academic and Disability Support in the Academic Support Center, Monocacy Hall, lower level (extension 7625). Accommodations cannot be provided until authorization is received from the Academic Support Center.

Course Outline

Date	Reading	Topic
M Aug 27		• Day 1 Activities
W Aug 29	• Chapter 0 • Chapter 1	• Java and OO
F Aug 31	• Chapter 2	• Additional Java Features
M Sept 3		• Labor Day
W Sept 5		• Netbeans and the Debugger
F Sept 7	• Chapter 4	• Java Collections Framework
M Sept 10	• Chapter 3	• Algorithm Analysis
W Sept 12		• Algorithm Analysis
F Sept 14		• Algorithm Analysis
M Sept 17	• Chapter 5 pp. 155-191	• Recursion
W Sept 19	• Chapter 5 pp. 191-210	• Backtracking
F Sept 21		• Divide and Conquer
M Sept 24	• Handout	• Dynamic Programming
W Sept 26	• Chapter 11 pp. 457-468	• Basic Sorts
F Sept 28	• Chapter 11 pp. 468-477	• Merge Sort
M Oct 1	• Chapter 11 pp. 477-489	• Quicksort
W Oct 3	• Chapter 11 pp. 489-493	• Radix Sort
F Oct 5		• Slip Day
M Oct 8		• Fall Break
W Oct 10		• Test #1
F Oct 12	• Chapter 6 • Chapter 7	• Array Lists • Linked Lists
M Oct 15		• Array Lists • Linked Lists

Date	Reading	Topic
W Oct 17	• Chapter 9 pp. 377-386	• Binary Trees
F Oct 19		• Recursion on Binary Trees
M Oct 22	• Chapter 9 pp. 386-393	• Binary Tree Traversals
W Oct 24	• Chapter 10 pp. 401-430	• Binary Search Trees
F Oct 26	• Chapter 10 pp. 430-442	• AVL Trees
M Oct 29	• Chapter 11 pp. 501-525	• Red/Black Trees • Tree Maps
W Oct 31	• Chapter 11 pp. 525-536	• Tree Sets
F Nov 2		• No Class - CCSC Conference
M Nov 5	• Chapter 13 pp. 551-566	• Priority Queues
W Nov 7	• Chapter 13 pp. 567-590	• Heap Sort
F Nov 9	• Chapter 14 pp. 599-626	• Hashing
M Nov 12	• Chapter 14 pp. 626-635	• Collisions
W Nov 14	• Chapter 15 pp. 643-650	• Graphs
F Nov 16	• Chapter 15 pp. 650-659	• Breadth-First Search • Depth-First Search
M Nov 19	• Chapter 15 pp. 659-669	• Minimum Spanning Trees • Shortest Paths
W Nov 21 & F Nov 23		• Thanksgiving Break
M Nov 26		• Slip Day
W Nov 28		• Test #2
F Nov 30	• TBD	• Parallel Algorithms
M Dec 3	• TBD	• Parallel Patterns
F Dec 5	• TBD	• Parallel Analysis
W Dec 7		• Review

The details of this syllabus and schedule are subject to change based on our progress through the material.