CS 225 – Fundamental Structures Spring 2009
Section 1210 10:30am – 11:40am M W F

Instructor: Jonathan Pikalek
Office: Ivers 234E
E-mail: pikalek@cord.edu
Web: http://www.cord.edu/faculty/pikalek/
Phone: 299-4237
Office hours: Tuesday 10:00am-12pm, 1pm-3:00pm
Thursday 10:00am-12pm, 1pm-2:00pm
Additional times available by appointment

Overview:
Intermediate data structures and techniques of object-oriented and structured programming. Discrete data types and structures, including arrays, files, sets, lists, trees, hash tables, sorting and recursion. Small to medium-scale programs are developed. *Prerequisite: C SC 125 — Introduction to Computer Science* [1]

Goals:
Preparation for continuing on to higher level computer science courses
Solid understanding of elementary data structures & abstract data type programming
Enhanced programming and software development skills.
General knowledge about algorithm analysis
Continued development of problem solving skills.

Required Texts:
Data Structures and Other Objects Using Java, 3rd Edition by Michael Main.

Attendance:
"Regular attendance and participation in class is critical to a student's success at Concordia College. Because any absence, excused or unexcused, detracts from the learning experience, students are expected to attend all classes." [2] If you are absent, you are responsibility for learning materials covered during class. If you have a scheduled absence due to extracurricular activities, it is your responsibility to provide prior notice.

Grading:
Hardcopy assignments are due at the beginning of class on the date indicated by the schedule. Digital (Moodle) assignments are due as posted online. Late work will be subject to a 10% per day penalty. Programming assignments will be graded based on program correctness, clarity and documentation. Assignments will account for 45% of your overall grade.

Short quizzes will be periodically given as preparation for exams. Quizzes will be categorized as assignments for the purposes of grading weights.

Since later materials depend on earlier materials, exams are semi-comprehensive; however the emphasis will be on materials covered since the last exam. A final project will be due during finals week. Exams & the final project will evenly account for 55% of your overall grade, meaning each exam and the final project will contribute 11% to your final grade.

Makeup exams will only be allowed for situations arranged with the instructor **prior** to the exam (and only for a very compelling reason) or for an illness or injury verified by a doctor's statement. Generally speaking, makeup exams must be scheduled within one week of the mixed exam.

The tentative grade cut off scale is as follows:

94% A	90% A-	
87% B+	84% B	80% B-
77% C+	74% C	70% C-
67% D+	64% D	60% D-

Academic Honesty:
You are expected to read and understand Concordia's policies regarding academic integrity [3]. While students are both allowed an encouraged to collaborate on assignments (especially programming assignments), it is imperative that as an individual you each understand the material and concepts covered by an assignment. Since academic dishonesty is expected to be rare to non existent, infractions will be handled in a case by case manner.

Special Needs:
If you require special accommodations in this course, please contact the Counseling Center in Academy Hall 106 (299-3514).

General Advice:
This course has a significant programming component. If it has been a while since you've programmed or your programming skills need improvement, I strongly suggest picking up a supplementary Java reference or textbook.

Make sure you understand the fundamental properties of various data structures. Try describing them without relying on programming jargon (this is an important skill when interacting with non-programmers). Be aware of their individual strengths and weaknesses.

The best way to learn programming is to write programs. You are encouraged to work on additional exercises and personal projects.

Strong problem solving skills are useful in many disciplines, including computer science. When working on problems don't just focus solely on the solution, actively think about your problem solving process as well.

Good software depends on more than code. Clear documentation and design is equally important; both will be factored into the grading of programming assignments.

Plan ahead: even experienced programs are prone to underestimating the amount of time required to complete a project. Set aside ample time to document, code, test and debug your programs.

Later topics are highly dependent on early material. Falling behind early on will make learning subsequent material significantly more difficult.

Collaborate with your peers. There is rarely a single best way to write a program. Different perspectives lead to greater insight into the process of problem solving and programming.

Collaboration means more than copying. You must take responsibility for understanding the materials. After all, exams will be a zero collaboration environment.

Get help early: the longer you wait to get help with the material, the harder it will be to get back on track.

References:
[1] 2007-08 Academic Catalog:
http://www.cord.edu/academic/catalog/departments/c_scdept.course.html

[2] College Handbook: General and Academic Polices - class attendance:
http://www.cord.edu/student/handbook/class_attendance.php

[3] College Handbook: General and Academic Polices – Academic Integrity:
http://www.cord.edu/student/handbook/academic_integrity.php