CS 125 – Introduction to Computer Science
Fall 2009
Section 10129 10:30 am - 11:40 am MWF Ivers Science 217
Section 10137 1:20 pm - 2:30 pm MWF Ivers Science 217

Instructor: Jonathan Pikalek
Office: Ivers 234E
E-mail: pikalek@cord.edu
Web: http://www.cord.edu/faculty/pikalek/
Phone: 299-4237
Office hours: 10am-11:45am T & R, 1pm-3:00pm T, 1pm-2pm R
Additional times available by appointment

Overview:
"The first course in the (computer science) major/minor sequence. An introduction to the Java programming language, algorithm design, structured and object-oriented programming techniques. No prior programming experience is assumed. *Prerequisite: higher algebra*" [1]

Methodology
Problem solving techniques will be introduced. Students will learn to analyze problem statements, develop corresponding models, convert their models to computer programs and analyze their solutions. Programming concepts, data structures and algorithms will be discussed. Commonly used algorithms will be introduced and mathematically analyzed. Students will consider the use of computing to address topics and problems from a variety of disciplines.

Goals:
Preparation for continuing on to higher level computer science courses.
General knowledge about computing and information technology.
Basic programming and software development skills.
Develop strong problem solving skills.

Writing:
"In computer science, writing is broadly defined to include several types of activities, and this is reflected in student's engagement in writing in a variety of ways in computer science course." "Introductory computer science courses will place special emphasis on program code and documentation with a special eye on good program coding and documentation styles" [4].

Required Texts:
Understanding Computers Today & Tomorrow, 12th Ed by Morley & Parker.
Java Concepts, 5th Ed by Horstmann.

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.

Practice problems & short quizzes will be periodically given as preparation for exams. Both 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. The fourth exam combined with the comprehensive final exam and will be held during finals. Exams will account for 55% of your overall grade, meaning the final & each exam will contribute 11% to your final grade.

Makeup exams will be allowed at the instructor's discretion. Typically, makeup exams will be allowed for situations arranged with the instructor **prior** to the exam. While not strictly required, a doctor's statement is preferred for verifying absence due to illness or injury. Generally speaking, makeup exams must be scheduled within one week of the missed exam. The format and difficulty of a makeup exam may differ from that of the original 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 understand the material and concepts covered by each assignment. Since academic dishonesty is expected to be rare to nonexistent, infractions will be handled in a case by case manner. The immediate repercussions for academic dishonesty include, but are not limited to: failing the assignment or failing the course. Additionally, I wish to explicitly point out the following college policy:

"...each violation of academic integrity whether involving a student or a faculty member and the consequences levied must be reported to the Academic Dean's Office....Individuals found to have violated academic integrity in any form will be placed on probation. More serious violations may warrant a year's suspension or expulsion from the college. If an individual commits a second violation, the minimum penalty will include a semester's suspension from enrollment at Concordia" [3].

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 will have 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 textbook.

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

[4] Concordia College Department of Mathematics and Computer Science – Writing in Computer Science.