CS10 – Introduction to Computer Science I
Spring, 2005

Course Information

OVERVIEW: The course introduces students to the field of Computer Science, focusing mainly on programming in the language Scheme without assuming any prior programming experience. We will introduce various concepts from CS including: logical thinking and problem solving, algorithm development, software development process, abstraction. The majority of time will be spent learning and using Scheme, however, the focus throughout will be on learning foundational concepts of Computer Science using computer programming (in Scheme) as the primary vehicle.
PREREQUISITES: Fulfillment of the mathematics competency requirement..
COURSE OBJECTIVES: (from the Westmont catalog) "No prior computer or programming experience required. Basics of programming including language features, disciplined programming style, and documentation. Problem solving, algorithm design, and the software development process. Control structures including branches, loops and functions. Files and program input/output. Arrays, structures and classes. Array-based lists. Pointers and linked structures. Object-oriented design, inheritance." The successful student in this course will have the programming skills to take moderately scaled problems, formulate an algorithmic solution, and implement that solution in Scheme. Such students will also have acquired the conceptual framework for continuing study in Computer Science.
LOCATION: *To be determined*
DAYS & TIME: Tuesday and Thursday, 10:00-11:50 am.
TEXT: *How To Design Programs: An Introduction to Programming and Computing*, by Matthias Felleisen, Robert Bruce Findler, Matthew Flatt, & Shriram Krishnamurthi. MIT Press: Cambridge, Mass. (2001).
COURSE WEBPAGE: (temporarily) http://homepage.westmont.edu/iba/CS010/CS010.html and (tentatively) http://www.westmont.edu/iba/CS010/CS010.html

Instructor Information

INSTRUCTOR: Dr. Wayne Iba
OFFICE: New Math/CS Modular Building
OFFICE HOURS: TBD
EMAIL: iba@westmont.edu
PHONE: (805)565-6799

Course Assignments, Requirements and Policies

EXERCISES: Exercises are an ideal way to learn the skills and concepts that are the intent of this course. You are expected to have the competence to solve every exercise at the end of each chapter of your text. However, in recognition that each of us learns different skills at different rates, you are not expected to complete and submit every exercise. You are expected to exercise discretion in your time-management policies regarding homework assignments. If you desire credit for exercises you completed, I must receive them electronically by 5:00pm on the Saturday of the week we cover a given Chapter. If you intend to submit exercises for credit, do not wait until the last minute to start working on them. Plan to turn your assignments in well before the deadline as late work may not be accepted. (Exceptions may be made at the sole discretion of the instructor.)

PROJECT: During the latter portion of the semester, students will be given a more substantial programming assignment that will involve several phases of development. Each phase will have an associated deliverable that will be submitted for grading; together, the project will represent the primary component of the overall grade for the course. Failure to complete one or more installments of the project will significantly impact your grade in a negative manner.

TESTS & EXAMS: There will be two exams during the semester and, a final exam. Exams must be taken at the scheduled time unless arrangement is made with the instructor in advance.

ATTENDANCE: Attendance is not required although it is highly recommended. Students are responsible for all material covered either in class or assigned readings.

GRADING: Students will be evaluated on how well they master concepts of computing and skills of program design as demonstrated using the Scheme programming language. A final weighted score will be based on the following: project (50%), collective exams (10 or 40%), and exercises (40 or 10%). I will apply either a 10/40 or 40/10 weighting to exams and exercises so as to maximize your final weighted score. The weighted final score, $0 \le x \le 1$, will yield a letter grade according to $\lfloor 10\,x \rfloor$ where 9 or 10 gets an A, and 8, 7, 6 and 5 or less get a B, C, D and F, respectively. A signifier, + or –, will be attached to the letter grade if $10\,x - \lfloor 10\,x \rfloor$ is > 0.7 or < 0.3 respectively. (A score of 1.0 will also merit a +.)

ACADEMIC HONESTY: As in every area of life, I expect that you will conduct yourself honestly within the context of this class. You are expected to have read and agreed to the Academic Dishonesty policy as part of the general Westmont Academic Policies, as well as the specifics of the policy on Plagiarism. Do not attempt to receive credit for work that is not your own without properly acknowledging sources via appropriate citations or references. You are encouraged to get help from your peers, but be sure that you acknowledge such help and that you understand the problem on which help was received. You should not be using or allowing electronic copies of any portion of code for the exercises or project. The consequences of violating the trust I implicitly extend to you will be according to the Westmont policy; but more serious will be the damage done to our academic relationship.