CPE 101: Fundamentals of Computer Science I

Instructional Information

Professor: Aaron Keen   E-mail: akeen@calpoly.edu

Phone: 756-2926   Office: 14-230

Office hours: M: 12-1pm, T: 1-3pm, W: 12-1pm, F: 12-1pm

Course Webpage: `http://users.csc.calpoly.edu/~akeen/courses/csc101`

Lecture Time and Location

- Section 1

    Lecture: MWF 9:10 am – 10:00 am, 02-212   Lab: MWF 10:10 am – 11:00 am, 14-301

- Section 9

    Lecture: MWF 2:10 pm – 3:00 pm, 192-242   Lab: MWF 3:10 pm – 4:00 pm, 20-127

Course Objectives

- Understand the basic principles of algorithmic problem solving.

- Apply top-down design, stepwise refinement, and procedural abstraction.

- Use basic control constructs and data types to solve problems.

- Lastly, exposure to the Python programming language.

Text: The course textbook is *Think Python - How to Think Like a Computer Scientist* by Downey. Available online.

Activities

Reading – The schedule outlines the order in which topics will be covered in lecture and the associated chapters/sections in the textbook that you should read. The lectures may not cover all of the material in the assigned reading, but such material may appear in labs, programming assignments, and exams.

Class Participation – The lectures are for your benefit. You should ask questions when you have them. I am more than happy to answer any questions that you may have relating to the course material.

Office Hours – Office hours are for your benefit. I am more than happy to answer any course related questions or to help with any programming issues you may have.

Labs – There are currently ten planned lab assignments. Each lab assignment must be demonstrated in lab by the end of the week (unless an alternate date is specified). You are allowed and encouraged to work together on lab assignments.

Once you have completed the lab assignment, you are encouraged to use the remaining lab time to work on your programming assignments.

Programming Assignments – There will be six programming assignments. These assignments are intended to be of greater length and complexity than the lab assignments.

Programming assignments must be completed individually. Collaboration on programming assignments is not permitted.

Lab Exams – There will be three lab exams.

Written Exams – There will be two midterm exams and one final exam. The exams will cover concepts presented in lecture and material based on the labs and programming assignments.

The exams will be **closed** book and **closed** notes.

Grading

The percentage breakdown for the course grade is as follows.

Activity	% per	% total
Labs	1	10
Assignments		30
Assignment 1	2	
Assignment 2	4	
Assignment 3	5	
Assignment 4	6	
Assignment 5	5	
Assignment 6	8	
Lab Exams	5	15
Midterm	10	20
Final	25	25
Total		**100**

Due Dates and Lateness

Programming assignments will be submitted electronically using `handin`. Except in the most extreme situations, **late assignments will not be accepted.** Even if you cannot complete an assignment, submit what you have for partial credit.

At a minimum, programming assignments **must compile** to be considered for grading. If a submitted program does not compile, you will receive a zero for that assignment.

Missed Exams

Make-up or early exams will not be given except in the most extreme situations. If you must miss an exam due to extreme illness, etc., contact the instructor immediately.

Collaboration and Honor Code

For each assignment that you submit, the following statements must be true.

- You wrote each line of code in your submission (excluding any code that the instructor has explicitly allowed you to use).

- Your submission is not a duplication (in any part) of another student's submission.

- Your submission is not a copy (in any part) of a single solution developed as part of a collaborative effort.

- You did not copy code from another source (student or otherwise).

- You did not provide your code to another student.

- You are able to answer detailed questions about each part of your submission.

If you do not agree to these rules, then you need to drop the course. Continued enrollment in the course is implicit agreement to adhere to these rules. Submissions will be compared using software that can reliably detect similarities among programs.

What is allowed?

- Help with proper syntax and run-time errors.

- Help finding (but not fixing) a bug, but only after a reasonable amount of time spent testing and debugging. Do not accept help until you have tried to find the bug yourself; this is part of learning. Using the wrong algorithm or incorrectly implementing the algorithm is not considered a bug in this sense.

- Discussion of general programming techniques.

- Discussion of algorithms and data structures required by the assignment description.

The Last Page

This page is so I can gather a little information about you at the beginning of the class. Please fill it out, tear it off and leave it with me on the way out.

Who are you?

Name: _____

Section: _____

Major: _____

Email: _____

Enrollment: _____ Enrolled

_____ Enrolled, thinking about dropping

_____ Trying to enroll

_____ Thinking about enrolling

How much programming
experience do you have?: _____

_____

Computing Goals?

Please share your interest in computing. How will knowing how to program help you in your career or in non-career-oriented endeavors?

Class Expectations?

Please take a minute to write out what your goals and expectations are for CPE 101. What do you want to learn? What do you expect to learn? Are these the same thing?