Spring 2-1-2019

CSCI 136.00: Fundamentals of Computer Science II

Marshall B. Pierce
University of Montana, Missoula

CSCI 136 Syllabus - Spring 2019

Instructor: Marshall Pierce (marshall.pierce@umontana.edu)

Overview

Building on the material from CSCI 135, this class is designed to give you basic skills for practical software development with Python. The focus will be on learning to express common concepts with object oriented designs.

Outcomes

By the end of this course, you will be able to:

- Perform basic operations with built in Python data types
- Write functions
- Use conditionals and loops
- Create new classes and use existing ones
- Model problem domains with inheritance and polymorphism
- Do file I/O and handle command-line arguments
- Decompose problems into simple steps
- In other words, write basic object oriented programs in Python

Requirements

- CSCI 135 or equivalent.
- Textbook: Introduction to Programming in Python: An Interdisciplinary Approach - Sedgewick, Wayne, and Dondero (http://amzn.to/1LO2eN6)
    - The book has a companion site with some of the book's content: https://introcs.cs.princeton.edu/python/home/
- The latest version of Python (currently 3.7)
- PyCharm Community
    - Install https://www.jetbrains.com/toolbox/app/ and make a JetBrains account, then use it to install PyCharm Community
    - If you use your .edu email address, you'll get access to all JetBrains tools: https://www.jetbrains.com/student/
-

Topics

Tentative schedule of one per week. We may touch on other topics if there is time.

1. Review of CSCI 135
2. Basic data types, control flow
3. Python built in collection types, I/O, CLI
4. Functions
5. Creating classes
6. Inheritance
7. Polymorphism
8. Exceptions, error handling
9. Data structures
10. Lambdas, functional programming

Course Format

Groups

Students will be randomly assigned in groups of four or so. The groups will be rearranged three times throughout the semester. It's OK to work with your group on homework (encouraged, even), but you must record the group members in the comments of all code done as a group, as well as who worked on what part of the code, e.g. "Alex worked on parsing the input, Jesse on producing the output, and everyone together on testing the program".

Board work

We will begin each lecture by randomly selecting students to go to the board and demonstrate their work. This will take about 15 minutes. Based on the student's performance, I will award the same grade to both the student and the group. This is done so that the group is responsible for each member's understanding. While presenting, students may ask their group two questions that have one sentence answers.

Homework

Homework will be assigned and you will need to submit it on time, but it will not be graded directly. Doing the homework is what will prepare you for board work and exams.

Lecture

After board work, I will lecture for about 35 minutes. This will typically be in the form of live coding to demonstrate the relevant topics.

Grading

Scale

A	94-100
A-	90-93
B+	87-89
B	83-86
B-	80-82
C+	77-79
C	73-76
C-	70-82
D+	67-69
D	63-76
D-	60-62
F	0-59

Students achieving the numerical scores above are guaranteed the associated letter grade. However, if average performance is low, I may decide to assign a higher letter grade for a lower score; e.g. a B+ for a numerical score of 84. Students taking the course pass/no pass are required to earn a grade of D or better in order to pass.

Weights

- In-class problems worked at the board: 20%
- In-class problems average score for your group: 20%
- Midterm 1: 20%
- Midterm 2: 20%
- Final exam: 20%

Attendance Policy

Attendance will not be taken. Students absent when called up to work problems on the board will be given a grade of 0%. Another group member will be selected to go to the board at random. Students informing the instructor of a valid reason for missing class in advance, via email, will not be called to the board. Valid reasons include family emergencies and illness. I may ask for documentation of absence (doctors note, death certificate, etc.).

Academic Integrity

All students must practice academic honesty. Academic misconduct is subject to an academic penalty by the course instructor and/or a disciplinary sanction by the University. All students need to be familiar with the Student Conduct Code. I will follow the guidelines given there. In cases of academic dishonesty, I will seek out the maximum allowable penalty. If you have questions about which behaviors are acceptable, especially regarding use of code found on the internet or shared by your peers, please ask me.

Disabilities

Students with disabilities may request reasonable modifications by contacting me. The University of Montana assures equal access to instruction through collaboration between students with disabilities, instructors, and Disability Services for Students. Reasonable means the University permits no fundamental alterations of academic standards or retroactive modifications.