Advanced Python Programming

"Any fool can write code that a computer can understand.
Good programmers write code that humans can understand."
— Martin Fowler

Fall 2017
CPE 2.216
Tuesdays 7:00 - 8:00 p.m.

Instructors: Aaron Comen, aaron.comen@gmail.com
David Garza, davidgarza95@utexas

Office Hours: EER 2.628 Mondays 4:00 - 6:00 pm
Tuesdays 2:30 - 3:30 pm
Wednesdays 4:00 - 5:00 pm

COURSE DESCRIPTION

This course has been designed to pick up where Introduction to Python Programming left off. We will quickly review the basic of programming, and then start with more advanced topics, including object-oriented programming. This course is more advanced than the previous course, and so requires more time and effort on your part.

Course Aims and Outcomes:

If you already know the basics of computer programming, i.e if statements, loops, functions, lists, and recursion, and you still want to dive deeper into the work of computer programming, then this is the course for you. If you put in the work required in this course, which includes active participation, completing the assignments, and programming on your own, you will learn many tools you will be able to use in your programming.

Learning Outcomes:

By the end of this course, students will:

- Understand the usefulness of object-oriented programming
- Create objects to write efficient algorithms
- Use different data structures to solve specific problems

Format and Procedures:

The course consists of eleven half hour classes held once a week. In class, we will cover the material assigned for each class. During class we will first explain the theory behind the programming technique we will be learning, followed by some examples. Then we will work on class assignments to get hands-on experience.

How to Succeed in this Course

The key to successfully learning how to program is to program. To do this you must read the assigned material and attentively listen to the class lectures. Then, work through the class assignments with your instructors. The homework assignments are the best chance you have to improve your programming skills. Spending time and effort working on the homework assignments by yourself will be what sets you apart as a student who *can* program as opposed to student who *knows* about programming.

COURSE REQUIREMENTS

Required Materials and Devices

You will be required to bring your computer. We will help you install the necessary applications in the first class. The textbook we will be using throughout the course is *Introduction to Programming Using Python* by Y. Daniel Yang, 1st edition. The textbook is not necessary, but if you want to have a useful resource we highly recommend it.

Classroom Expectations

Class attendance While you are not required to assist to every class, we will not cover material from a previous class in another. If you wish to learn programming, come to every class. If you miss class, you can come to office hours to catch up.

Class participation Just by being in class is not enough. If you have any questions we expect you to ask them. You are also expected to complete the class assignments and the homework assignments every week.

Assessments and Grading

Although the course is not officially graded, we will be grading homework assignments. This will help you see how well you are doing in the course, as well as to help identify what topics you should put more effort in. Your grades will not be made public, and are for in-class use only.

Course Schedule

Date	Main Topic(s)	Readings – to be completed before class	Assignments
9/19	Introduction		
9/26	Multidimensional Lists	Chapter 11	HW 1
10/3	Files	Chapter 13	HW 2
10/10	Exception Handling	Chapter 13	HW 3
10/17	Tuples, Sets, and Dictionaries	Chapter 14	HW 4
10/24	Objects	Chapter 7	HW 5
10/31	Objects	Chapter 7	HW 6
11/7	Inheritance	Chapter 12	HW 7
11/14	Sorting	Chapter 17	HW 8
11/21	Thanksgiving Break		
11/28	Sorting	Chapter 17	HW 9
12/5	Data Analytics	Class Notes	HW 10