CM 151 – Introduction to Computational Science – Fall 2015

Instructor: Eric S. Ho (hoe@lafayette.edu)
Office: Kunkel 13
Office hours: TTh 2-4 pm (or by appointment)

Lecture: MWF 9:00-9:50 am, Venue: AEC 513
Lab: F 1:10-4:00 pm, Venue: AEC 519

Course Description:
The goal of this course is to instill computational thinking in problem solving. Students will learn techniques in formulating computational solutions to solve problems using the general-purpose computer language Python. Python is a simple but deep programming language, which is widely acclaimed by scientists, engineers, economists, mathematicians, IT developers, etc. for its simplicity, portability, versatility, and extensibility. In this course, students will acquire basic techniques in writing and troubleshooting Python programs through intensive hands-on exercises.

Learning Outcomes:
At the completion of this course, students should be able to:
- Think algorithmically
- Formulate programming approach to solve problems
- Master semantics and syntax of Python
- Define and utilize Python's built-in data structures: list, dictionary, and set
- Read and store information in computer files
- Plot basic graphs using Python's graph library
- Design simulations to study complex problems

Grading Policy:
- 9 lab reports – 4% each (total 36%)
- Two lab exams – 10% each (total 20%)
- Three 1-1/2 -hour in-class cumulative exams. The first, second, and third exams carry 10%, 10%, and 20%, respectively (total 40%)
- A four-page essay to summarize your experience in learning computer programming (5%)

Basically, lab reports are Python codes that solve the lab exercises. As practice is extremely important in mastering a programming language, active participation in lab is highly recommended. You are allowed to discuss the exercises with your classmates. But programming work must be done individually. You must submit your lab reports to Moodle in order to be graded. Lab report is due 24 hours after lab session. Students are highly encouraged to submit their lab work on time. However, if extra time is needed, you must seek prior permission from Prof. Ho. In most cases, requests are granted without penalty.

Lab exams will be conducted **during lab hours**. Students are allowed to bring a 2-page hand-written note for reference, but access to books and Internet are prohibited during exam. Two types of questions will appear in lab exams: debugging, and problem solving. For debugging questions, you are given programs that contain syntax and/or semantics errors. Your task is to mend them until they produce expected output. Problem solving questions are problems similar to lab exercises, thus, attending labs and working on lab exercises can greatly help to achieve high grade.

Three written exams are scheduled for this course. **Two will be conducted during lab hours and one will be scheduled in final exam period.** Check the timetable at the end of the syllabus. Exams are cumulative. All written exams are conducted in closed book/notes format. Test materials are based on lectures, reading assignments, and lab exercises. Past exam questions and extra practice problems will be posted in Moodle prior to exam.

Final grade is based on the percentage of total points earned for the course i.e. 3 written exams, 2 lab exams and 9 lab reports. Letter grades are assigned according to the following scale:

A	93-100%	C	73-76
A-	90-92	C-	70-72
B+	87-89	D+	67-69
B	83-86	D	63-66
B-	80-82	D-	60-62
C+	77-79	F	0-59

Required Textbook:
Python programming in context 2nd, Bradley N. Miller, David L. Ranum. Jones & Bartlett Learning.

Students are required to read the assigned readings from the textbook. Exams are based on materials from lectures, labs AND the textbook.

Attendance Policy:
Attendance and class participation are critical to learning. Although attendance will not be taken for lectures and labs, statistics tells that attendance highly correlates with the final grade. If there is in-class quiz, absentees will receive a zero for that quiz.

Federal Credit Hour Policy:
The student work in this course is in full compliance with the federal definition of a four [two or one as appropriate for half and quarter unit courses] credit hour course. Please see the Registrar's Office web site (http://registrar.lafayette.edu/additional-resources/cep-course-proposal/) for the full policy and practice statement.

Academic Honesty:
You are expected to abide by the college policy on Intellectual Honesty (see student handbook and attached document).

Disability Statement:
In compliance with Lafayette College policy and equal access laws, I am available to discuss appropriate academic accommodations that you may require as a student with a disability. Requests for academic accommodations need to be made during the first two weeks of the semester, except for unusual circumstances, so arrangements can be made. Students must register with the Office of the Dean of the College for disability verification and for determination of reasonable academic accommodations.

Privacy Policy:
Moodle contains student information that is protected by the Family Educational Right to Privacy Act (FERPA). Disclosure to unauthorized parties violates federal privacy laws. Courses using Moodle will make student information visible to other students in this class. Please remember that this information is protected by these federal privacy laws and must not be shared with anyone outside the class. Questions can be referred to the Registrar's Office.

CM 151 – Introduction to Computational Science – Fall 2015

Week		Major Topics	Assigned Reading
1	Aug 31	Introduction and overview of the course	Ch 1
	Sep 2	IDLE, Python interpreter, PythonTutor	
Lab 1	Aug 4	Computing with numbers	
2	Sep 7	Variables	Ch 1
	Sep 9	Turtle graphics	
Lab 2	Sep 11	Functions & For-loop	
3	Sep 14	Problem solving strategies	Ch 1 + additional
	Sep 16	Code jam	exercises
	Sep 18	**Sep 18: Lab exam 1**	
4	Sep 21	Archimedes & Function return	Ch 2
	Sep 23	Archimedes & Function return	
Lab 3	Sep 25	Leibniz & Wallis: Accumulator Pattern	
5	Sep 28	Leibniz: Accumulator Pattern	
	Sep 30	Monte Carlo: Conditional statements	
	Oct 2	Wallis**, Oct 2: Written Exam 1 (week 1-5)**	
6	Oct 5	Strings	
	Oct 7	Built-in string functions	Ch 3
Lab 4	Oct 9	Transposition	
7	Oct 12	*Fall break Oct 12-13*	
	Oct 14	Substitution	Ch 3
Lab 5	Oct 16	Vignere	
8	Oct 19	Lists	Ch 4
	Oct 21	Code jam	
	Oct 23	**Oct 23: Lab exam 2**	Ch 1-3
9	Oct 26	Lists	Ch 4
	Oct 28	Lists	
Lab 6	Oct 30	Dictionaries	
10	Nov 2	Dictionaries	Ch 4
	Nov 4	Dictionaries	
Lab 7	Nov 6	File operations	Ch 5
11	Nov 9	File operations	Ch 5
	Nov 11	File operations	
	Nov 13	**Nov 13: Exam 2 (week 1-11)**	
12	Nov 16	Exam 2 Review	
	Nov 18	File operations, while-loop	
Lab 8	Nov 20	Matplotlib installation	Additional materials
13	Nov 23	Pylab graphics	
	Nov 25	*Thanksgiving*	
Lab 9	Nov 27	*Thanksgiving*	
14	Nov 30	Pylab graphics	
	Dec 2	Pylab graphics	
Lab 10	Dec 4	Stock chart	
15	Dec 7	*No class due to special academic duty*	
	Dec 9	*No class due to special academic duty*	
	Dec 11	*No class due to special academic duty*	

Useful Links:

1. Python: http://python.org/
2. Python Tutor: http://www.pythontutor.com/
3. Stackoverflow: http://stackoverflow.com/
4. pythonanywhere https://www.pythonanywhere.com/
5. http://www.econpy.org/ (tailored for economists)