Course Syllabus

Jump to Today ✎ Edit

Syllabus

Computer Programming for Lawyers - Spring 2020

Overview

This course provides a rigorous introduction to programming as taught in the Python programming language. It largely covers the same content as taught in an introductory undergraduate computer programming course, with presentation and applications tailored for law students and lawyers. The first half of the course covers core concepts in programming and software engineering. The second half of the course covers applications relevant to the everyday workflows of a working lawyer.

This is a hands-on class. Each student will spend most class sessions using his or her own computer to read, write, and debug code. Every student must bring a computer to every class, on which freely available software will be installed. Students will be required to complete weekly problem sets between class meetings. To obtain a passing grade, students must complete problem sets, participate in class sessions, and demonstrate they have learned the assigned skills.

No Experienced Programmers Allowed: This class is intended for beginners only. If you have taken any classes in college or graduate school that featured computer programming, or if you have mastered any computer programming language, you are not permitted to take this class. If you are unsure whether you fall within the excluded categories, please contact the course instructor immediately. Students who are too experienced to qualify for this class should explore taking Intermediate Computer Programming for Lawyers instead.

First Week Attendance Mandatory: All enrolled and waitlisted students must be in attendance at the start of the first lecture and the first lab in order to be eligible for a seat in the class.

The Waitlist: There is a large waitlist for this class. We have no control over who gets into the course from the wait list. Once again, you must attend both the first lecture and the first lab to be eligible for a seat off of the waitlist. If you are on the waitlist for more than one section of the course, you are required to attend only one of the labs for which you are on the list. If you officially enrolled in one section and are on the waitlist for a second section, attend the lab

section for which you are already enrolled. If any of the scenarios described in this paragraph apply to you, be sure to tell a TA at the week one lab session you attend.

Course Mechanics

Course Format:

Lectures: 3:30pm-5:30pm in McDonough 200 with Prof. Ohm. Attendance is vital and mandatory.

Labs: Various times on Thursday for one hour. You are enrolled in a specific fifteen person lab that will be led by one or two lab TAs. Be sure to attend the lab in which you are enrolled.

Video Reviews: For the first few weeks of the semester, we will release a video of a course instructor working through the process of solving the previous week's homework.

Reading Due: Readings will primarily come from PDFs of chapters of *Computer Programming for Lawyers*, a draft textbook by Jonathan Frankle and Paul Ohm. These chapters will be distributed on Canvas. This is a draft textbook, so please do not circulate it.

Homework: Students will complete fifteen weekly homework assignments (in the form of problem sets) throughout the semester. These assignments will be distributed on and submitted to Canvas. Problem Sets 0-9 must be completed individually. Problem Sets 10-14 may be completed in pairs. Problem sets are assigned immediately after Tuesday's lecture and are due one week later on Tuesday at 3:00 PM, a half hour before the next lecture; for each day late that you submit a problem set, we will deduct 20% from your score on that assignment. We will assign problem sets on weeks when there is no lecture (i.e., faculty retreat and spring break). Under no circumstances will we provide solutions to any problem sets; however, for the first few weeks we will release a video of a course instructor walking through the process of solving the problem set. **Problem set feedback will be provided by TAs, but only Prof. Ohm will give grades.**

In our experience, the problem sets are both the distinctive hallmark and the greatest stressor of the course. The course requires graded homework every week of the semester, and these assignments are unlike almost any other work you've completed in law school. We've written up some of the key differences in **this blog post (https://cp4l.org/post/problem-sets/)**, which you should read to assess whether this class is for you. You might find other posts on the blog interesting or informative.

Piazza: We will use a collaboration tool called Piazza to centralize all course discussion. Piazza functions as a shared, interactive FAQ for the course: you can ask questions about lecture/lab /textbook material and problem sets visible to the entire class, and instructors and fellow students can answer. Because of the nature of the course collaboration policy (below), we have

configured Piazza so all messages get posted privately initially. We will publish questions we think are potentially of general interest.

You can access Piazza through Canvas. We encourage you to use Piazza as much as possible, and to answer questions asked by others as much as possible. Piazza allows you to ask questions in a way that is anonymous to other students, so please don't be shy about asking questions even if you think they are simple or obvious; chances are that, if you have a question, several others in the class have it too.

Instructors

This course is a true team effort. It is run by a staff of a professor, a head TA, and several section TAs.

Bios

Professor Paul Ohm: Professor Ohm teaches the course and *Intermediate Programming for Lawyers*, the course that you will be qualified to take after this one and the course that all of your TAs are currently taking. He created this course with Jonathan Frankle and has taught *Programming for Lawyers* for four years.

Riana Freedman (Head TA): Riana Freedman is a third-year law student at Georgetown. She took *Intermediate Computer Programming for Lawyers* last spring and will serve as the head TA this semester.

Contact Information

Name	Email	Office Hours
Prof Ohm	ohm@law.georgetown.edu	TBD
Riana Freedman	rjf53@georgetown.edu	TBD

You should always send questions of any kind to Piazza. If you cannot find an answer in Piazza, send questions about TAs or the feedback they provide on problem sets first to Riana. Professor Ohm can answer any questions you cannot otherwise get answered. Feel free to send any questions about sensitive matters directly to Professor Ohm without going to Piazza first.

Collaboration Policy

Preamble

Part of the process of learning to program a computer is getting stuck and working to get

unstuck. Part of getting unstuck is talking to your classmates, but part of it is rolling up your sleeves and working on the problem alone. For this reason, although we encourage many forms of collaboration, we are implementing a strict, bright-line collaboration policy for the class. Any violation of this policy will be considered a violation of the school's Student Disciplinary Code.

Rules

1. For graded assignments, you may not view the code of anybody else who has taken or is taking the class. If an assignment has been explicitly designated to be completed in groups or pairs, you may view the code of your assigned groupmates or partner.
2. For graded assignments, you may not show your code to anybody except the course staff. Course staff maintain a strict "hands-in-pockets" policy - they cannot touch your computer or type anything for you.
3. You may discuss general concerns or concepts with your classmates, but keep this at a general level. For example, you are allowed to discuss questions such as, "what does this error message mean?" or "what is a tuple?"
4. You may discuss answers to class exercises, those from the book, or those found elsewhere that have not been assigned, and in doing so, you may share or view code with others. Do not use this as an end-around this policy.
5. You must acknowledge specifically any assistance or collaboration in code that you submit. This includes any online or outside resources you consulted in preparing your solution.

Piazza

We encourage the active use of Piazza, and we do not want this collaboration policy to have a chilling effect. If you have a question, ask it in the most general, conceptual way possible so that the question can be asked publicly and the entire class can benefit from your effort. By default, all questions will be posted as private. If we think a question is of general interest, we will publish it to all.

Office Hours

To encourage students to attend and take advantage of office hours, we offer a "loophole" to the collaboration policy for office hours. If other students at office hours are stuck on the same point as you are (whether a point about Python or a specific part of the problem set), you can feel free to discuss concepts and strategies for overcoming that problem. This loophole is only available when you are attending office hours. Students who are stuck on a different point of the problem set (e.g., students who are further along or who have not made it that far) may not participate in this conversation; doing so will be considered a violation of the collaboration policy.

Violations

Violations of this collaboration policy will result in - at a minimum - a score of 0 on the relevant problem set. We will pursue violations of the collaboration policy pursuant to Georgetown Law's policies on plagiarism. Please do not cheat. We have been at this for much longer than you have, and - if you think you found a way to cheat - we've already thought of it and are scanning your submissions for it.

Week by Week topics

Below is a list of topics we will cover in this class. This is a tentative schedule, subject to change. For each week's assignment, consult the class Canvas page.

Week 0 Before Class

You must complete and submit "Problem Set 0" before coming to your first class! See Canvas for details.

Week 1 Jan 14 (Lecture) and Jan 16 (Lab): Introduction to Python

This week, you will learn about the fundamental types of data in Python and rudimentary ways of processing that data.

Week 2 Jan 21 and 23: Conditionals and While-Loops

This week, you will learn basic control flow that allows programs to change their behavior based on their inputs.

Week 3 Jan 28 and 30: Lists and Iteration I

This week, you will learn how to store collections of data in lists and create lists with for-loops. You will also learn some advanced behaviors of strings.

Week 4 Feb 4 and 6: Lists and Iteration II

This week, you will learn advanced behaviors of lists and strings. You will also learn how to store structured data in dictionaries.

Week 5 Feb 11 and 13: Datastructures

This week, you will learn how to create sophisticated datastructures that organize collections of

information. You will also learn how to store and retrieve information from files on your computer.

Week 6 Week of Feb 18 (No Lecture or Lab): Functions I

This week is faculty retreat, so there is no lecture or lab. You will read the first part of Chapter 12, which covers the basics of functions, and complete a short problem set on that material.

Week 7 Feb 25 and 27: Functions II

This week, you will learn about functions, which make it possible to separate large programs into smaller, reusable parts.

Week 8 March 3 and 5: Files and Directories

This week, you will learn how to manipulate files and directories using Python.

Spring Break Week of March 10 (No Lecture or Lab): Regex 101

This week, you will complete an online tutorial introducing you to regular expressions.

Week 9 March 17 and 19: Regex I

This week, you will learn about regular expressions, a highly sophisticated way of searching databases of text.

Week 10 March 24 and 26: Regex II and Excel

This week, we will further study regular expressions. In addition, we will discuss how to use Python to manipulate Excel files.

Week 11 March 31 and April 2: Web Scraping and PDF files

This week, we will introduce HTML, the basics of web scraping, and PDF document manipulation.

Week 12 April 7 and 9: Web Crawling

This week, we will discuss web crawling.

Week 13 April 14 and 16: APIs and CSV files

This week, we will discuss interacting with remote sources of data via APIs and CSV file manipulation.

Week 14 April 21 and 23: Machine Learning

This week, we will close by discussing the fundamentals of modern machine learning and neural networks.

Course Summary:

Date	Details	
Tue Jan 14, 2020	Problem Set 0: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362793)	due by 3pm
	Problem Set 0: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362954)	due by 3pm
Tue Jan 21, 2020	Problem Set 1: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362794)	due by 3pm
	Problem Set 1: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362968)	due by 3pm
Tue Jan 28, 2020	Problem Set 2: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362795)	due by 3pm
	Problem Set 2: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362969)	due by 3pm
Tue Feb 4, 2020	Problem Set 3: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362955)	due by 3pm
	Problem Set 3: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362970)	due by 3pm
Tue Feb 11, 2020	Problem Set 4: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362956)	due by 3pm

Date	Details	
	Problem Set 4: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362971)	due by 3pm
Tue Feb 25, 2020	Problem Set 5: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362957)	due by 3pm
	Problem Set 5: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362972)	due by 3pm
	Problem Set 6: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362958)	due by 3pm
	Problem Set 6: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362973)	due by 3pm
Tue Mar 3, 2020	Problem Set 7: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362959)	due by 3pm
	Problem Set 7: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362974)	due by 3pm
Tue Mar 17, 2020	Problem Set 8: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362960)	due by 3pm
	Problem Set 8: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362975)	due by 3pm
	Problem Set 9: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362961)	due by 3pm
Tue Mar 31, 2020	Problem Set 10: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362962)	due by 3pm
	Problem Set 10: Submission Page (https://georgetown.instructure.com/courses/101719	due by 3pm

Date	Details	
	/assignments/362977)	
Tue Apr 7, 2020	Problem Set 12: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362963)	due by 3pm
	Problem Set 12: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362978)	due by 3pm
Tue Apr 14, 2020	Problem Set 13: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362964)	due by 3pm
	Problem Set 13: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362979)	due by 3pm
Tue Apr 21, 2020	Problem Set 14: Opinion Survey (https://georgetown.instructure.com/courses/101719/assignments/362965)	due by 3pm
	Problem Set 14: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362980)	due by 3pm
	Final Survey (https://georgetown.instructure.com/courses/101719/assignments/362966)	
	Problem Set 9: Submission Page (https://georgetown.instructure.com/courses/101719/assignments/362976)	
	Roll Call Attendance (https://georgetown.instructure.com/courses/101719/assignments/380950)	