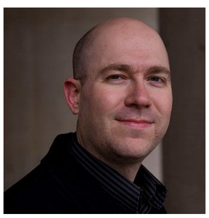
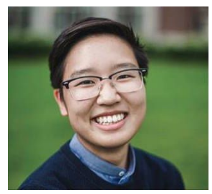
PS50: Computing for Science Studio

Course Syllabus

Instructor

	Alán Aspuru-Guzik aspuru@chemistry.harvard.edu Website: http://aspuru.chem.harvard.edu

Teaching Fellows

	Hannah Sim ssim@g.harvard.edu
	Tim Menke tim_menke@g.harvard.edu

Meetings

Fridays 1-5 PM
Science Center B-10

Course Websites

Canvas:
https://canvas.harvard.edu/courses/36395
Piazza:https://piazza.com/harvard/spring2018/ps50

Course description

The modern practice of basic and applied science requires proficiency with computers and programming. In this course, the students will learn applied computer programming with an emphasis on practical examples related to the simulation of matter. Students will be exposed to interdisciplinary computational problems drawn from several scientific disciplines including chemistry, biology, materials science and physics. The studio format allows for a mixture of lectures, guided computer programming exercises and open scientific problem solving in a 4 hour-long session. The studio session is complemented with computer programming exercises solved by teams of students outside of class and office hours. The computer language employed will be Python. The students will be exposed to several numerical and data analysis libraries that can serve them in a variety of modern contexts.

Prerequisites: High-school chemistry, biology, math and physics.

Materials

A personal laptop or notebook computer running MacOS, Windows, or Linux is required to attend class.

Course structure

Each **weekly 4-hour session** (Friday 1-5) will consist of a lecture where the objective is to give an overview of the main material to cover for the week. The lecture will be followed by a self-paced quiz and computer programming exercise that will further go into these concepts. The final segment of the studio session will consist of time where the students will be able to get started with their weekly programming exercise. We will often assign reading that will be quizzed before the next lecture.

Two **office hours** (TBD) will be available for consultation with the teaching fellows about course content and exercises. Forums in Piazza will allow for further discussion

Grading

The course will be graded as follows:

- 30% of the grade will be obtained by the automatically-graded in-class Python notebooks and quizzes
- 30% of the grade will be your weekly assignments turned in on time (the deadline is Fridays at 12:50 PM). No late assignments will be graded under any circumstance.
- 40% of the grade is your final project.

Resurrection

If for some chance you miss a lecture, and therefore you don't turn in your in-class IPython notebook or your weekly assignment, these scores will be "stored" in a bank of points to recover for a total of up to 10% of your grade. This percentage will be added to the total contribution of the final project, for up to a 50% of the grade contribution from it. This will allow you to *resurrect* up to 10% of your grade if the final project is well carried out.

Drop-out

If you feel you want to drop the course, please discuss with the instructor before the deadline for drop/add classes.

Textbooks

Throughout the classes we will assign readings from different books. All the required textbooks will be available for free through Safari Online library, using your Harvard ID (http://proquest.safaribooksonline.com.ezp-prod1.hul.harvard.edu/). These books are also a great complement to the lecture material:

Name	Author	Link
How computers work	Ron White; Timothy Edward Downs	**http://id.lib.harvard.edu/aleph/010107656/catalog**
Introduction to Programming in Python	Robert Dondero; Kevin Wayne; Robert Sedgewick	**http://id.lib.harvard.edu/aleph/014448065/catalog**

Mastering Python Scientific Computing.	Hemant Kumar Mehta	http://id.lib.harvard.edu/aleph/014539581/catalog
Learning scikit-learn: Machine Learning in Python.	Raúl Garreta; Guillermo Moncecchi	http://id.lib.harvard.edu/aleph/013948908/catalog

Lecture topics

Lecture 1	Jan 26	**Computers and programming basics**
Lecture 2	Feb 2	**Algorithms and debugging**
Lecture 3	Feb 9	**Managing and visualizing data**
Lecture 4	Feb 16	**Object-oriented programming**
Lecture 5	Feb 23	**Python APIs and databases**
Lecture 6	Mar 2	**Molecular dynamics and using NumPy**
Lecture 7	Mar 9	**Optimization and the Monte Carlo method**
	Mar 16	**Spring Break**
Lecture 8	Mar 23	**Machine learning basics**
Lecture 9	Mar 30	**Electronic structure, chemoinformatics and molecular screening**
Lecture 10-12	April 6-20	**Final project: Robotics (more information to follow)**