2022 Winter PIC 16A: Python with Applications I

Welcome to PIC 16A! Here's a bunch of stuff about the class that you may have questions about. But first, I want you to know that your number one priority should be your physical and mental health, not this nor any other class. Stay hydrated, sleep well, and try to enjoy your time at UCLA. Let me know if you need an extension for any assignment.

Lectures: MWF 2-2:50pm ~~Geology 4660~~
Discussion 4A: TR 2-2:50pm ~~Geology 6704~~
Discussion 4B: TR 3-3:50pm ~~MS 5138~~

Instructor: Harlin Lee. Please call me "Harlin". You can email me at harlin@math.ucla.edu or DM me at Campuswire.

Open-door office hours: On Mondays 3-4pm and Wednesdays 3-4pm, the door to my office (~~MS 7360~~ Zoom) will stay open. You are encouraged to drop by and ask questions about the homework, coding projects (PIC or personal), or just share feelings about the class. In fact, the first time you drop by and ask a question, you'll get extra credit.

One-on-one meetings: I am happy to schedule a closed-door office meeting or private zoom meeting if you want to talk about your performance in this class, grad school applications, career plans, or whatever feelings you have about the world.

I am here to listen to whatever you have to share, but if you disclose any incidents of sexual harassment or sexual violence, I am required by UCLA to report them to the Title IX Coordinator. If you would like to maintain confidentiality, you should talk to CaPS https://counseling.ucla.edu/about-us/confidentiality or CARE https://careprogram.ucla.edu/.

TAs and Office hours:
Vishnu Vardhan Bachupally (vishnu1911@ucla.edu)
Shruti Mohanty (shrutimohanty@g.ucla.edu)
Tuesday 12-1pm, Thursday 1-2pm, Friday 11am-12pm

Learning Assistants (LAs): James Chen, Rachel Yu, Mansa Krishna

Textbooks: None. When debugging, check the error messages -> official documentation → Stack Overflow → other resources in Google.

Official documentation: https://docs.python.org/3/library/index.html
Official tutorial: https://docs.python.org/3/tutorial/index.html

Course websites

- All class materials will be uploaded to Canvas https://bruinlearn.ucla.edu/courses/75178
- All student questions should be posted at Campuswire https://campuswire.com/c/GBDBAF4AE/feed
- All assignments will be uploaded to and graded in Gradescope https://www.gradescope.com/courses/336955

Waitlist policy
PIC doesn't use PTEs. Please see https://ww3.math.ucla.edu/enrollment-into-math-and-programming-in-computing-pic-courses/. If you're on the waitlist, you can expect to be enrolled after the waitlist period is over, unless the department tells me not to.

Official Math department description of PIC 16A:
(5) (Formerly numbered Programming in Computing 16.) Lecture, three hours; discussion, two hour. Requisites: course 10A, Computer Science 31 or equivalent. In depth introduction to the Python programming language for students who have already taken a beginning programming course in a strongly typed, compiled language (C++, C or Fortran). Core Python language constructs, applications, text processing, data visualization, interaction with spreadsheets and SQL data bases, and creation of graphical user interfaces. P/NP or letter grading.

… and course objectives:
The student will be familiar with the core Python language components, including program syntax, fundamental data types, flow control, file and console I/O, the creation of functions, the creation and use of classes. The student will be familiar with the Python interpreter, Python notebooks, and Python integrated development environments. In addition, the student will be capable of both creating and using Python modules.

How will this class differ from 2021 Fall PIC 16A?
The overall content of the class and activity-based discussion sections will be very similar to the Chowdrow, Perlmutter, Murray versions of 16A last quarter. But I am scheming to add a little more of the following topics: writing Python scripts with IDE, running them using terminal, understanding the Python package structure, object oriented programming, and a larger project component using Github + reproducible programming practices.

Topics

- Python basics: variables, data types, control flow, functions, Python keywords

- More advanced Python syntax: classes, inheritance, object oriented programming, exceptions, magic methods, I/O

- Programming skills: using tools (e.g. Anaconda, Jupyter-lab, VScode), version control with Git, understanding Python package structure, understanding Python dependencies, debugging, commenting, reading and writing documentation, code reproducibility, computing and ethics

- Applications and packages:
  - Matrix operations with Numpy
  - Data wrangling with Pandas
  - Data visualization with Matplotlib
  - Machine learning with Scikit-learn
  - (maybe) Image processing and computer vision with openCV
  - (maybe) SQL database interaction with SQLite3
  - (maybe) Natural language processing with NLTK
  - (maybe) Graphic user interface with PyQt
  - (maybe) Regular expressions with re

Tentative schedule: https://math.ucla.edu/~harlin/16Aschedule.pdf

Usual grading scheme

- Homework: 26%
- Discussion assignments: 15%
- (Very low pressure) mini-quizzes: 9%
- Project: 20%
- Midterm (Feb 7 Monday): 15%
- Final exam: 15%

If you miss the midterm for any reason (incl. illness, travel, family emergency)

- Homework: 26%
- Discussion assignments: 15%
- (Very low pressure) mini-quizzes: 9%
- Project: 20%
- Midterm (Feb 7 Monday): 0%
- Final exam: 30%

Extra credit opportunities

- Completion of surveys throughout the quarter: 0.25 each
- First time coming to open-door office hour and asking a question: 0.25
- Amazing meme about the content of the course: up to 0.5
- Outstanding quality and quantity of participation on Campuswire or discussions: up to 0.5

Letter grade cutoffs

- A+: 100 + awesome project and participation
- A: 93
- A-: 90
- B+: 88
- B: 83
- B-: 80
- C+: 78
- C: 73
- C-: 70
- D+: 68
- D: 63
- D-: 60

There will be *no* curving, because these scores are supposed to measure how much you've mastered the class material, and nothing else. But don't worry—if class average is too low, I'll make sure to give everyone a chance to bring their grades up via extra assignments like exam wrappers. Collaboration can only help you in this setting, so please ask lots of questions and talk through tough problems with your peers! I have complete faith that every one of you can master the content of this class and get an A. :)

Lecture structure

MWF lectures will be delivered in-person. I will *not* take attendance, but I encourage you to join as often as you can. If there are any slides, notes, videos, or code files used during the lecture, I will upload them to Canvas by the end of the day. I will try to record and post lectures in the beginning of the quarter while people are shopping for classes, but I can't promise lecture recordings after the first week or two. Lectures will *not* be streamed online via Zoom unless otherwise specified.

Discussion structure

This is where you will learn the most. TR discussions will be centered around group activities with the help of TAs and LAs. Your worksheet will be graded mostly on effort, but you do need to be present to get credit for that session. You can still get full credit if you miss up to 3 discussion sections, so if you have to miss a day for any reason, don't worry and just count it towards one of your drops.

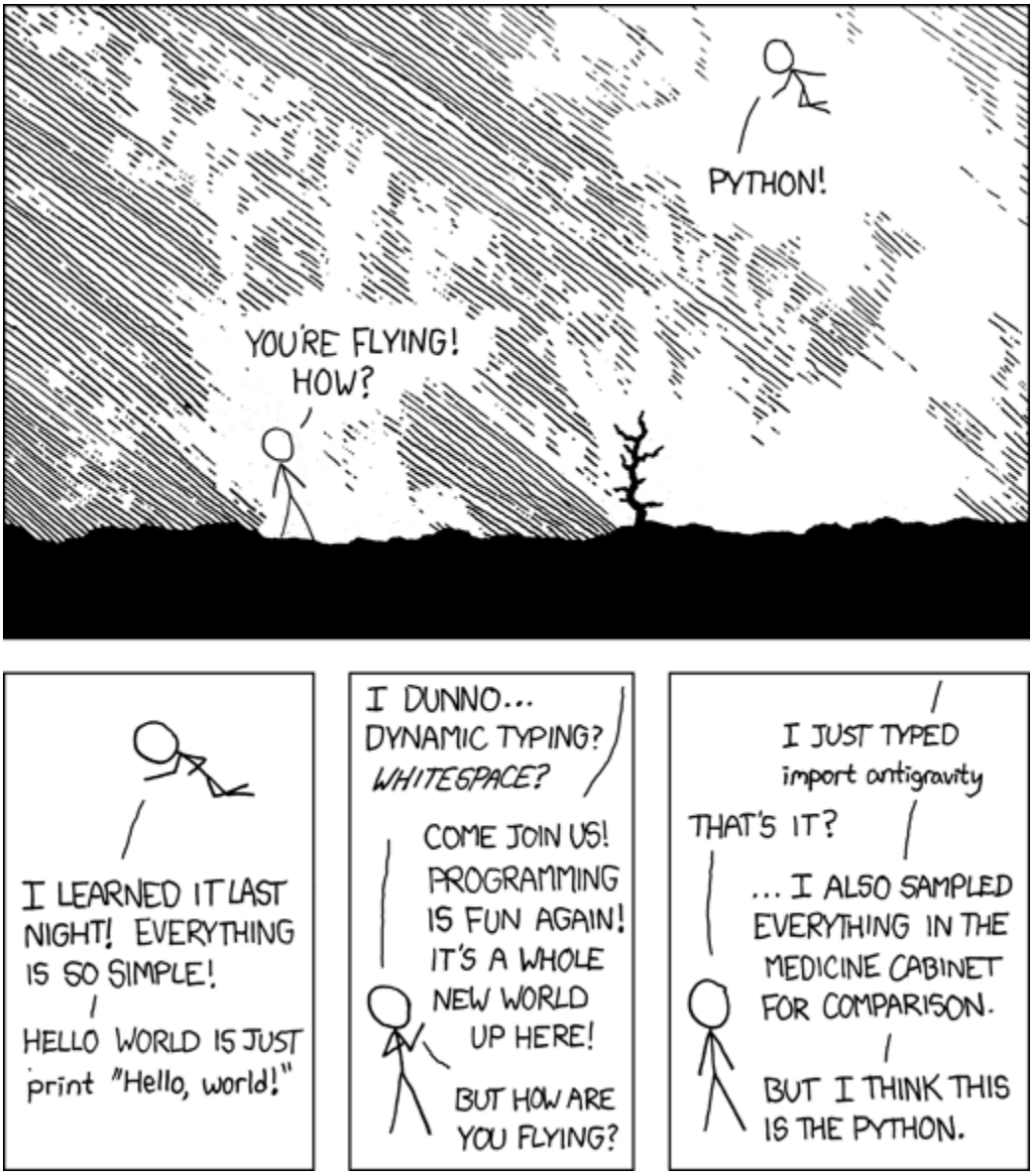
Homework structure

Homeworks are tough, but you will learn a lot. Collaboration is encouraged, but everyone should still write and submit their own homework. The lowest homework score will be weighted only half as much as the other homeworks. Homeworks can be submitted up to 2 days late, at a discount of 10% per 24 hours past the deadline. For example, if you turn in a homework 20 hours late that would have received a 98%, your grade will be an 88%. If you turned it in 39 hours late your grade would be a 78%, and so on. Homeworks turned in more than 2 days (48 hours) late will receive 0% unless you've received an extension from me.

Project structure
You will work on a project with your discussion group, which will be assigned based on project interests (to the best of my abilities) in the beginning of the quarter. Working on a project with real data and/or users is the best way to learn "how to make things from scratch," and I want you to have something published on Github that you can be proud of by the end of this course. If you want to use something from another class or research project, that is allowed as long as you have the PI/other course instructor's approval and you have convinced me that there is a significant PIC-16A-specific component. The project will happen in phases, and there will be many opportunities for you to get feedback as you go. More detailed instructions to follow after week 1.

Mini-quiz structure
These quizzes are only here to help you identify the big gaps in your understanding right away instead of waiting until exam week. I will not drop any mini-quiz scores, but you get multiple attempts.

Exam structure
~~Midterm will happen during normal lecture time on Feb 7 Monday~~. Midterm will be a 24-hour take home exam on Feb 7 Monday, per departmental policy. The final will *not* be 3 hours long. I promise.



https://xkcd.com/353/