General Information

Instructor

Matt Bishop; *Email*: mabishop@ucdavis.edu *Web*: http://seclab.cs.ucdavis.edu/ bishop
Office: 2209 Watershed Science *Phone*: (530) 752-8060; *Skype*: checkcode
Office Hours: MW 9:00–9:50am, Tu 4:10–5:00pm, F 10:30–11:30am, or by appointment, or by chance

When you send me email, please begin the Subject field with "MHI 289I" so I see that the letter has to do with the class. I receive lots of email and, while I look at it all, I *sometimes* miss things, or skim the Subject fields to see which letters are very important. Putting "MHI 289I" in the Subject field will tell me it is very important.

My office is on the central campus in Davis. If coming to my office is not convenient, feel free to call me or use Skype.

Lectures

MW 3:10pm–4:00pm, in Room 3205, Education Building, UC Davis School of Medicine, 4610 X St., Sacramento

Course Outline

Basics of computer programming essential to the study of informatics. Impacts on systems within healthcare, public health, nursing, research, and others. The class will teach students to write programs in the Python programming language.

Course Goals

The overall goal is to learn how to use the Python programming language to solve problems. More specifically, we hope you will:

1. Learn how to write programs, and how to use an integrated development environment, debuggers, compilers, and interpreters;
2. Learn the basics of the Python programming language, and through it the basic control and data structures, operations and data types in programming languages; and
3. Learn how to design and write an algorithm.

Text

Charles Severace, *Python for Informatics: Exploring Information in Python 2*, CreateSpace Independent Publishing Platform, Seattle, WA, USA. ISBN: 978-1-4923-3924-3.
It is on-line at `http://www.pythonlearn.com/book.php` and is free.

Class Web Site

The class web site is on Canvas. To access it, go to `http://canvas.ucdavis.edu` and log in using your campus login and password. Then go to MHI 289I in your schedule. Announcements, assignments, handouts, and grades will be posted there, and you *must* submit any assignments there. The alternate web site, `http://nob.cs.ucdavis.edu/classes/mhi289i-2018-01` has everything except grades, and you cannot submit work there.

Extra Credit

Extra credit is tallied separately from regular scores. If you end up on a borderline between two grades at the end of the course, extra credit will count in your favor. But failure to do extra credit will never be counted against you, because grades are assigned on the basis of regular scores. You should do extra credit if you find it interesting and think that it might teach you something. Remember, though, it is not wise to skimp on the regular assignment in order to do extra credit!

Grading

We expect the weighting to be as follows (but reserve the right to change it, with notice):

Homework assignments	60%
Final project	40%

When we grade an assignment, we will look not simply at your answers but also the program(s) you used to obtain the answers. We grade on style, documentation, and a number of factors as well as correctness. See the handout **All About Homework** for more details.

Academic Integrity

The UC Davis Code of Academic Conduct, available at `http://sja.ucdavis.edu/cac.html`, applies to this class. In particular, for this course, all work submitted for credit must be your own. You may discuss your assignments with classmates, with the instructor, or with a teaching assistant to get ideas or a critique of your ideas, but the ideas and words you submit must be your own. Unless explicitly stated otherwise, collaboration is considered cheating and will be dealt with accordingly.

The single exception to this rule is debugging. Once you have written your program, if you need help debugging it, you are free to ask a classmate for help providing that classmate has also written the program. Sometimes having someone else look over a program that is not quite working right will lead you to the best way to fix it, and you both will gain valuable experience in looking at programs and figuring out what is going on. But you must not collaborate on writing the program.