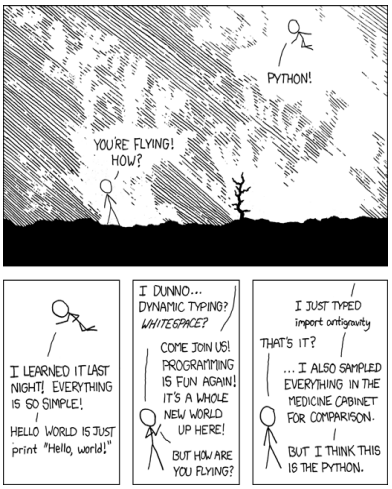

EEB 177

Practical Computing for Biologists

Tuesday Thursday, 9:30-10:45 AM,

Botany 325

Computer Lab Wednesdays 1-3 and 3-5; WGYoung 4067

Michael Alfaro

michaelalfaro@ucla.edu
www.pandorasboxfish.com
Terasaki 2147
Office Hours: TBD
(510) 794-5019

Course contents are tentative and may be updated to reflect class interests and guest lecturers.

Course Description: A general trend within nearly all fields of biology has been an explosion in the availability of data. As a result, biology is becoming increasingly computational. The goals of this course are to provide you with the tools you need to manipulate and analyze common sources of biological information such as such as text output from computer programs and electronic data bases. You will learn the basics of shell operations, regulation expressions, and the fundamentals of python programming including control statements, reading and writing of files, and scripting. You will also learn specialized libraries for programming in ecology and evolutionary biology.

Prerequisite(s): None.
Credit Hours: 4

Text(s): *Python for Biologists and Advanced Python for Biologists*, 1st Edition
Author: Stefano Allesina; downloadable **here**

Author: Martin Jones; You can pick up the book in pdf form through the following link. **Use this coupon code to get these books at a discount:** a834f87f

Course Objectives:
By taking this course, students will gain experience in:

1. working from the shell

2. using regular expressions
3. writing documents in latex
4. programming in python
5. visualizing data in R

Grades

1. Participation 10%
2. Homework 10%
3. Lab assignments 30%
4. Final Project 50%

Your grade in this course will be determined by participation, homework assignments, lab assignments, and projects. Your **participation** grade will include team exercises and unannounced in class challenges and quizzes. Regular class attendance is essential to earn a high participation score. **Homework** will be assigned regularly to get you using the tools we cover in lecture. You will be introduced to many new languages, packages, and programs in this course. The best way to learn this material and develop your skills is through immersion and repetition. The motivation behind the homework (and final project) assignments is to give you reasons to fire up a terminal and start hacking. **Lab Assignments** will provide an opportunity to apply skills we cover in lecture with practical examples and will help prepare you for your final project. **Final Project** The bulk of your course grade will be determined by a class project. The project will require you to integrate all of the tools that you learn over the quarter to manipulate and analyze data. The final project grade will be divided into the following stages and due dates:

- Project idea submitted to CCLE by Friday Week 4: Feb 3rd (1%).
- Git repository link with ReadMe and pseudocode for project due Friday Week 5: Feb 10th (4%).
- Commit with 1 working function due Friday, Week 6: Feb 17 (5%).
- Commit with data I/O and 2+ functions due Friday, Week 7: Feb 24 (5%).
- Latex or Markdown draft with outline of project, documented code, due Friday, Week 8: March 10th (5%).
- Latex or Markdown draft with expanded outline of project, expanded code, 1 or more integrated figures, and minimum 3 references due Friday, Week 9: March 10th (5%).
- Lightning Presentation (details to follow) will be given the week of March 13th (Week 10) 25%
- Completed Projects (reports, working code, everything) will be accepted the week of March 20th and must be received by Friday March 27th, 5PM (50%)

Course Policies:

- **General**

– Bring a laptop to class! We will be working through programming examples in every lecture.
– In class challenges and exercises must be turned in during class. Make up assignment will not be given but I will drop the two lowest in-class assignment scores before calculating your participation grade.
– Homework assignments should be submitted electronically to the CCLE or posted to your repository as instructed. Late homework will be penalized 10% per day late.

Tentative Course Outline:
The weekly coverage and readings will change depending on progress of the class and student interests. Key to sources: **scb**: Scientific Computing for Biologists (Allesina); **pb** or **apb**: Python (or Advanced Python) for Biologists (Martin).

Week	Content
Week 1	• Preliminaries, introduction to version control, Unix • Reading assignment: scb Chapters 0, 2.
Week 2	• The Shell, Unix, Text Editors, Regular Expressions • Reading assignment: scb Chapters 1, 5.
Week 3	• Python Programming I • Reading assignment: scb Chapters 3;
Week 4	• Python Programming II • Reading assignment: scb Chapter 4;
Week 5	• Scientific Typesetting (Latex and Markdown) • Reading assignment: scp Chapter 7
Week 6	• Python Programming III: • Reading assignment: scb Chapter 6
Week 7	• Web Scraping and Scripting: • Reading assignment: TBA
Week 8	• Statistical computing in R; • Reading assignment: scb Chapter 8
Week 9	• Data visualization in R • Reading assignment: scb Chapter 9.
Week 10	• Project Presentations • Reading assignment: TBA