

Course Syllabus

[Jump to Today](#)

Goals and Description

The course is designed for students with little or no programming experience but with a desire to learn how to develop software to aid in neuroscience research. Given the popularity of Python and the SciPy library among neuroscience labs worldwide, the course will be conducted in Python. However, much of what is learned will be easily translated into other programming languages (e.g., Matlab). In addition to lecture material, each class will include laboratory time during which students will work on specific problems and get direct instruction and assistance.

The course will begin with a basic introduction to programming (variables, functions, using the debugger, etc.) and to NumPy (vectors, matrices, vector-based math). We will quickly transition into writing programs to aid in the analysis of various sources of neuroscience data using these domains to learn programming skills and techniques. Topics will include signal processing, image processing, and data visualization. Once completed, students should be proficient enough in Scientific Python to apply their skills to their own laboratory research.

Objectives

The course uses seven problem sets to assess progress and understanding of programming concepts. Programming is a skill that must be developed through practice and requires not only basic levels of comprehension (knowledge, comprehension, and application), but also deeper levels of understanding (analysis and synthesis) to be able to creatively solve programming challenges. The problem sets are central to developing these skills.

Each problem set consists of Python code written to solve a particular task. Students will be graded based on:

- Whether the stated goals of the problem set were met (does it run and give the correct output?)
- Whether good coding practices were employed (e.g., including appropriate documentation, clarity of code, etc.)
- Whether the program demonstrates a presence or absence of knowledge of the concepts currently being developed (e.g., applying loops, sub-functions, etc. when asked)

Throughout the course, we will share solutions to problems (including the instructor's solutions) to help understand the range of possible solutions and to evaluate what constitutes good or poor solutions.

Logistics











- Course website
 - <https://canvas.eee.uci.edu/courses/554>
- Location
 - McGaugh 2246 (1201 on 10/4 and 10/6) **Not in MSTB 210 - Lab A**
 - Tuesday / Thursday 9:30-10:50
 - No class on 9/29, 10/18-20, 11/3, 11/15
- Instructor
 - Craig Stark
 - cestark@uci.edu
- Access to Scientific Python via Anaconda: <https://www.continuum.io/downloads> (<https://www.continuum.io/downloads>)
- Dive Into Python 3: <http://www.diveintopython3.net> (<http://www.diveintopython3.net>)
















Topic Overview

- Part 1: Basics of programming and Python
 - Anatomy of a program (Dive Into Python: Chapter 1)
 - Data types in Python (Dive Into Python: Chapter 2)
 - ints, floats, and bools
 - strings in Python 3
 - lists
 - tuples, sets, dictionaries
 - Preview of NumPy arrays
 - Understanding scope, variables, and functions
 - Using the IDE

- Directories, Files, and Comprehensions (Dive Into Python: Chapter 3)
 - The filesystem structure
 - Portable code issues
 - Managing and creating paths
 - Globbing
 - List comprehension
- Part 2: Basics of NumPy and SciPy
 - Differences between NumPy arrays and Python lists
 - Creating arrays
 - empty, ones, zeros, etc
 - The make-it-like functions
 - "Views" or "slices" and the power / danger of the equals sign
 - Indexing arrays
 - Simple indexing
 - Slices
 - Logical indexing
 - Math functions in ndarray objects
 - Built-in function sampler
 - Array-based math vs. iteration (why we use NumPy)
 - File I/O
 - Masked arrays
 - Statistical functions in SciPy
- Part 3: Plotting with Matplotlib
 - Basic line, bar, etc. graphs
 - Subplots
 - Plot annotation and customization

Assignments Summary:

Date	Details	
Thu Sep 22, 2016	 Class	9:30am to 10:50am
Tue Sep 27, 2016	 Problem Set 1	due by 9am
	 Class	9:30am to 10:50am
Thu Sep 29, 2016	 NO CLASS	12am
Tue Oct 4, 2016	 Class	9:30am to 10:50am
Wed Oct 5, 2016	 Problem Set 2	due by 9am
Thu Oct 6, 2016	 Class	9:30am to 10:50am
Tue Oct 11, 2016	 Class	9:30am to 10:50am
Thu Oct 13, 2016	 Class	9:30am to 10:50am
Mon Oct 17, 2016	 Problem Set 3	due by 11am

Date	Details	
Tue Oct 18, 2016	 NO CLASS	12am
Thu Oct 20, 2016	 NO CLASS	12am
Mon Oct 24, 2016	 Problem Set 4: Basics of NumPy	due by 10am
Tue Oct 25, 2016	 Class	9:30am to 10:50am
Thu Oct 27, 2016	 Class	9:30am to 10:50am
Tue Nov 1, 2016	 Class	9:30am to 10:50am
Thu Nov 3, 2016	 NO CLASS	12am
Tue Nov 8, 2016	 Class	9:30am to 10:50am
Thu Nov 10, 2016	 Class	9:30am to 10:50am
Tue Nov 15, 2016	 NO CLASS	12am
Thu Nov 17, 2016	 Class	9:30am to 10:50am
Tue Nov 22, 2016	 Class	9:30am to 10:50am
Thu Nov 24, 2016	 Class	9:30am to 10:50am
Tue Nov 29, 2016	 Class	9:30am to 10:50am
Thu Dec 1, 2016	 Class	9:30am to 10:50am

