

COMP 383 Computational Biology

Introductory Lecture

Spring 2016

Loyola University Chicago

hwheeler1@luc.edu

Bioinformatics Major Courses



Biology



**Computer
Science**

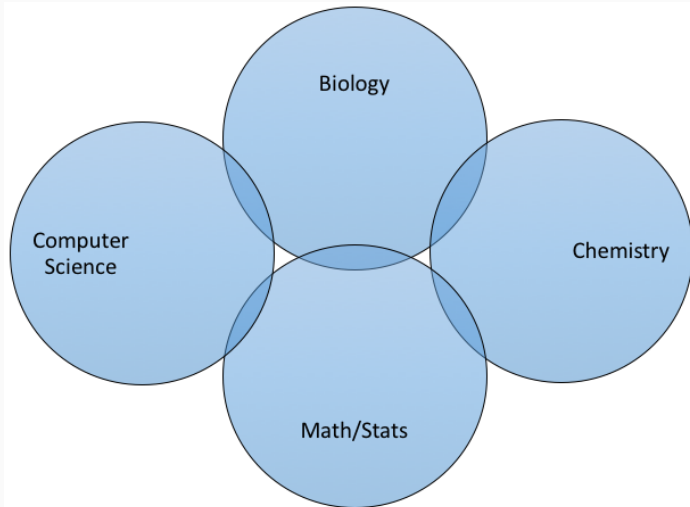


**Math /
Statistics**

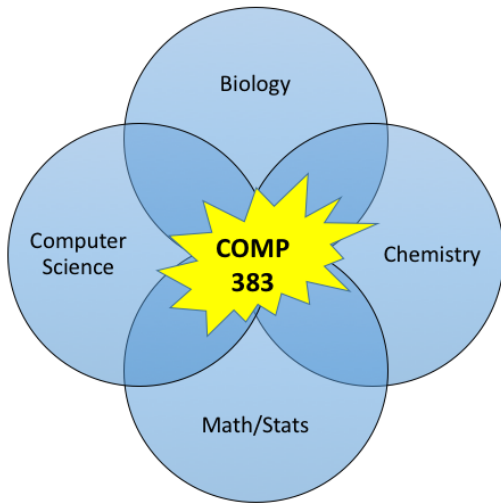


Chemistry

In BIOL 388 Bioinformatics, you started to see an intersection of these fields



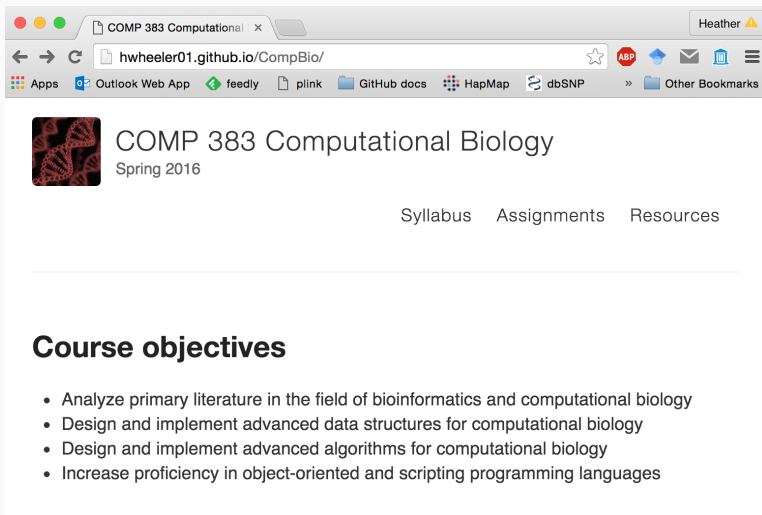
COMP 383 Computational Biology integrates all you've learned



COMP 383 Course Structure

1. First few weeks, learn Python, apply it to common biological problems (toy examples) - 20%
2. Group **Research** Project: You will be given freedom to work on a computational biology problem - 80%
 - This could involve automating a research pipeline or building an analysis tool
 - This is a “real life” research project (options from Loyola faculty will be available), so data will be messy and unanticipated obstacles will occur
 - It will be important to set benchmarks and stay on task in class
 - Don't procrastinate, get things done and strive to make your code better
 - Learn all you can, the future of biology/medicine/data analysis needs you!

The syllabus and assignments are posted on the class website



The screenshot shows a web browser window with the address bar displaying `hwheeler01.github.io/CompBio/`. The page title is "COMP 383 Computational Biology" with a subtitle "Spring 2016". A navigation bar contains links for "Syllabus", "Assignments", and "Resources". Below this, the section "Course objectives" is followed by a bulleted list of four goals.

COMP 383 Computational Biology
Spring 2016

Syllabus Assignments Resources

Course objectives

- Analyze primary literature in the field of bioinformatics and computational biology
- Design and implement advanced data structures for computational biology
- Design and implement advanced algorithms for computational biology
- Increase proficiency in object-oriented and scripting programming languages

`http://hwheeler01.github.io/CompBio/`

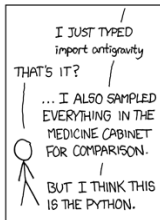
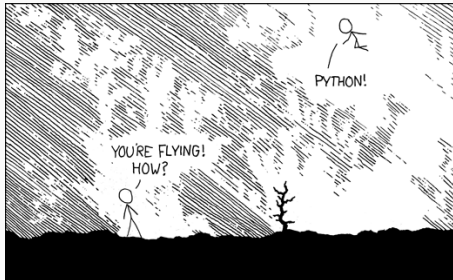
Why Python?

1. Everybody's doing it. It has a large (and growing) user base among scientists.
2. Python is a general purpose, high level, programming language – you should be able to do anything you want to do using Python, and it should be relatively easy to accomplish.
3. It's free, well-documented, and runs on all operating systems.

Why Python?

4. It's easier for novices to pick up than most other languages and it's also used by many professional programmers. This makes collaborating with both novices and experts easier.
5. A dynamic language like Python allows us to write small programs quickly and to also manage the complexity of larger ones.
6. If we want to squeeze every last ounce of performance out of our hardware, then a compiled language like C++ is better, but if we want to quickly answer a research question, which is usually the case in computational biology, Python is often easier/faster.

Why Python?



<http://xkcd.com/353/>

For this class, I recommend Python version 2.7 rather than Python 3.x

- ROSALIND assumes 2.7
- BioPython for Python 3.x is not as stable, but improving all the time
- If you're used to Python 3, feel free to use it, just note some of the ROSALIND hints may require slightly different syntax

Python 2.7 Strings

```
print "Hello, World!"  
a = "Hello"  
b = "World"  
type(a)  
print a + ", " + b + "!"*3  
a[0:4]
```

```
## Hello, World!  
## <type 'str'>  
## Hello, World!!!  
## 'Hell'
```

Python 2.7 Numbers

```
a = 12
b = 2.5
c = a + b
type(a)
type(b)
print str(a) + " + " + str(b) + " = " + str(c)
```

```
## <type 'int'>
## <type 'float'>
## 12 + 2.5 = 14.5
```

Python 2.7 Division

```
18/5
```

```
18/5.0
```

```
18/float(5)
```

```
## 3
```

```
## 3.6
```

```
## 3.6
```

Or place this import at the beginning of the file:

```
from __future__ import division
```

```
18/5
```

```
## 3.6
```

Start first assignment in ROSALIND

<http://hwheeler01.github.io/CompBio/assignments/>

The screenshot shows the ROSALIND website interface. At the top, there's a navigation bar with the ROSALIND logo, links for About, Problems, Statistics, and Glossary, a search bar, social media icons for Facebook and Twitter, and links for My Classes, hewlat, and Log out. Below the navigation bar, the main heading is "COMP 383 Computational Biology" followed by "solved 2 of 30". There are buttons for "Print all problems", "Announcements", "All classes", and a red "Leave" button. The text "by Heather Wheeler at Loyola University Chicago" is displayed. A paragraph explains the submission rules: "Each problem is due by 11:59 PM on the date listed. You will be required to upload your commented code with your solution to each problem. Your code must be commented in **your own words** and turned in independently. Code should be commented sufficiently so someone learning Python can understand it. **If you had a partner for a problem, note her/his name in your code comments.** Cheating includes submitting as your own work something that has been written by an outside person (or web site). If you are working with a partner, an "outside person" refers to someone other than your partner." Below this is a table with 7 columns: Num, Title, XP, Cost, Due Date, Questions, and Solutions. The table lists 6 problems, with the first two marked as solved.

Num	Title	XP	Cost	Due Date	Questions	Solutions
1	Installing Python	1	1	Jan. 21, 2016	☞	☞
2	Variables and Some Arithmetic	1	1	Jan. 21, 2016	☞	☞
3	Strings and Lists	0	1	Jan. 21, 2016	☞	
4	Conditions and Loops	0	1	Jan. 21, 2016	☞	
5	Working with Files	0	1	Jan. 21, 2016	☞	
6	Dictionaries	0	1	Jan. 21, 2016	☞	