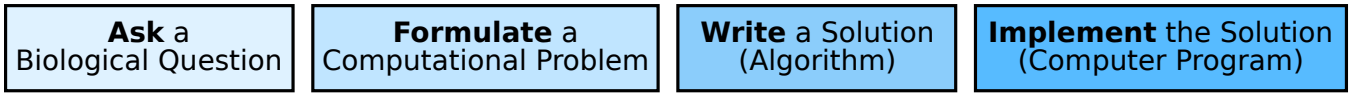
Introduction to Computational Biology

Professor: Anna Ritz (aritz@reed.edu, Bio 200B)

Goals and Objectives

High-throughput technologies produce massive amounts of data, much too large to analyze by hand. The goal of this course is to learn how to analyze DNA, RNA, and protein sequences using computers. Through a combination of foundational examples and current research questions, this course aims to demystify computer science, molecular biology, and some of the ways they intersect.

1. You will know about the properties of DNA, RNA, and proteins, the relationships among these molecules, and some *biological questions* that have puzzled researchers.
2. You will know how to convert a biological question into a *computational problem* that can be solved using computers.
3. You will know how to read and understand solutions to computational problems, which will be formalized as a series of tasks (an *algorithm*). You will learn about general approaches for solving computational problems, and you will be able to apply these approaches to new problems you encounter.
4. You will know how implement the algorithms by writing *computer programs* in Python, which can be run and understood by others.



Using this general framework, you will learn how to to analyze DNA content, identify protein binding patterns, compare sequences, and discover variation within genomes. In the last two weeks, you will formulate your own sequence analysis problem, implement a solution, and present your findings to the class. By the end of this course, you will understand what types of biological questions can be investigated using computers, and what limitations computational methods impose on the understanding of biology.

Course Details

Time & Place: Lecture is MWF 10:00am-10:50am, Physics 240A. There are three lab sections:

- **Lab S1** W 1:10pm-3:00pm, ETC 211
- **Lab S2** Th 1:40pm-3:30pm, ETC 211
- **Lab S3** Th 3:40pm-5:30pm, ETC 211

If you have an occasional conflict with your section, you can attend another lab section. You can also sit in on multiple lab sections if you find them useful.

Textbook: Available at the bookstore and on two-hour reserve at the library.
Bioinformatics Algorithms: An Active Learning Approach.
Edition 2, Volume 1. Phillip Compeau & Pavel Pevzner.
(Other editions are fine, but page numbers may shift.)

Office Hours: Wed 3:10pm-4:10pm & Th 10:30am-11:30am in Bio 200B.
Email Anna to set up an appointment if you cannot make office hour times.

Tutors & TAs: The TAs will be in lab, have posted DoJo drop-in tutoring hours, and are available for one-on-one tutoring.

- Madeline (Maddy) Doak
- Giorlando (Gio) Ramirez
- Sol Taylor-Brill
- James Vesto

Academic Accommodations: Please discuss any documented accommodations with Anna, especially those that include extensions on assignments, extra time on exams, and unexpected absences. Disability Accommodation Notification Letters can be obtained from Disability & Accessibility Resources (DAR). All discussions will remain confidential.

Collaboration Policy:
`https://www.reed.edu/biology/courses/bio131/files/collaboration-policy.pdf`
Note that the Collaboration Policy is different from other biology or math courses.

Moodle: `https://moodle.reed.edu/course/view.php?id=2908`
Used for lecture materials, announcements, assignments, and labs.

Repl.it: `https://repl.it/`
Used for programming labs and assignments.

Rosalind: `http://rosalind.info/problems/locations/`
Used for challenge problems.

Website: `http://reed.edu/biology/courses/bio131/`
Public-facing website for the course, which includes final projects.

The Honor Principle is in effect at all times.

Course Structure

Assignments (60%). Programming assignments are the majority of the points of this class. There are ten graded assignments of varying lengths.

- You may work with others on the assignments (except solo homeworks, see below), within the limits of the Collaboration Policy.
- There will be at least one **checkpoint** midway through each assignment. If you submit your code with a question by the evening of a checkpoint, Anna will give quick feedback the next morning. She might not catch tricky bugs, but this is good if you have a simple question or can't debug an error.
- Anna will provide an expected number of hours for each assignment. Keep track of the number of hours you spend. If it takes you significantly *less* time than expected, then attempt the challenge questions. If it takes you significantly *more* time, check in with a TA or Anna.

Two of the assignments will be **solo homeworks**, where you may work with TAs and Anna but not collaborate with other students. Solo homeworks will check your progress in acquiring the programming skills required for the course.

Exams (20%). There are two in-class exams that will assess knowledge about the biology topics and computational thinking. You will not be quizzed on Python programming concepts. If you have accommodations that involve extra exam time, be sure to make arrangements with Anna before the first exam. There is no final exam (though we will use the final exam slot).

Final Project (10%). There is a two-week Final Project at the end of the semester that is based on biological sequence analysis. It will consist of four deliverables: a project proposal, a project update, a final report, and the code. You will give a presentation during finals week.

Participation (10%). Participation will be assessed in three ways:

- **Labs.** Lab hand-ins will not be graded, but will be used for participation points.
- **Code Conferences.** There will be at least one Code Conference during the semester to discuss assignments to-date. No preparation beyond your assignment solutions are required.
- **Engagement with Material.** Participation in the form of discussion, questions, worksheets, and group work will be considered.

Late Policy. You are allowed two "grace periods" that give you a 24-hour extension for two of the ten programming assignments. You must email Anna to tell her you are using the grace period before the last deadline of the homework. After 24 hours, late assignments are no longer accepted.

Keys for Success

Start the assignments early. The time you spend on each assignment might vary drastically, depending on how much you debug your programs.

- Have a plan for the work you need to do for each component of the assignment.
- Work through problems on paper if you are confused with a concept.
- It is better to have something working that is not quite complete than something that does not work but "has all the parts."

This course is cumulative – topics build upon each other throughout the semester, and attendance is critical. Let Anna know if you miss a lecture or a lab to get caught up. While there is no penalty for missing lecture, lab attendance is expected unless explicitly stated or accommodations have been arranged.

When in doubt, ask. Asking for help is sometimes unintuitive, but a little clarification goes a long way. If you have no idea how to start a problem (which happens!), stop by office hours or the DoJo. Also take advantage of the one-on-one tutoring available.

Working collaboratively is fun! If possible, try to work with others on the assignments within the limits outlined in the Collaboration Policy.

Start the assignments early. Really.

Topics and Deadlines

The course content is flexible; refer to the Google Calendar on the course website and Moodle for up-to-date information. Assignments may have multiple deadlines.

	Topic	**Homework/Exams**	**Lab**
Week 1 01/27–01/31	**Intro to Intro** Python as a Calculator Data Types	HW1: Warm-Up	L1: Repl.it and Python Intro
Week 2 02/03–02/07	**Transcription** Iteration	HW2: Python Practice	L2: Pattern Maker
Week 3 02/10–02/14	**Translation** Conditionals and Functions	HW3: Central Dogma I	L3: Compute Reverse Complements
Week 4 02/17–02/21	**Origin of Replication** Formulating Computational Problems	HW4: Central Dogma II	L4: Hamming Distance Functions
Week 5 02/24–02/28	**DNA Motifs** String Manipulation	Exam 1	L5: Asymmetry of Replication in *E. Coli*
Week 6 03/02–03/06	**Summarizing Motifs** Lists of Lists	HW5: Motif Finders I (solo HW)	L6: Motif Scores
Week 7 03/09–03/13	**Finding Motifs** While Loops	HW6: Motif Finders II	L7: Greedy Motif Finder
Week 8 03/16–03/20	**Genome Assembly** Python Dictionaries	HW7: Overlap & De Bruijn Graphs	L8: Drawing Graphs
SPRING BREAK!!!			
Week 9 03/30–04/03	**Sequence Alignment** Recursion	HW8: Alignment I	L9: Longest Common Subsequence
Week 10 04/06–04/10	**Sequence Alignment** Recursion, cont'd.	HW9: Alignment II	L10: Alignment Backtracking
Week 11 04/13–04/17	**Genome Rearrangements**	HW10: Rearrangements (solo HW)	L11: Dot Plots
Week 12 04/20–04/24	**Machine Learning in Bioinformatics**	Exam 2	Final Project Time
Week 13 04/27–05/01	**Final Projects**		Final Project Time
Project Presentations during the Final Exam Period			