It is time to begin thinking about your final project! You get to pick an area of bioinformatics/computational biology that interests you. You can build on material we have done in class, or you can use it as an opportunity to learn about subject matter we don't have time to cover. **I strongly encourage you to work with a partner.** You may want to use Slack as a way of finding others with similar interests, or talk to me if you want help finding a partner.

## Option 1: Learn about a new algorithm or algorithmic problem

Pick an algorithm or algorithmic problem that extends something we have discussed in class, or that relates to another area of bioinformatics that we haven't covered. You can get ideas from the recommended textbook, the new online textbook, another book, or even a research paper. At the end of this document is a list of potential project ideas that help get you started thinking about what you might want to do.

## Option 2: Learn about a particular biological application or question that relies on computational biology

Pick a particular biological application or question that requires computational methods to be able to study. For example, maybe you're really interested in metagenomics (the study of genetic material from environmental samples) and want to investigate how computational approaches are used in this area. Or perhaps you want to further investigate how reference genomes are built and used. Or perhaps you would like to better understand the coronavirus COVID-19 strain that has interrupted our semester. If you go this route, you will want to narrow down your focus as much as possible. You may find it useful to get ideas from biology classes you have taken, our textbook, or research papers.

## Option 3: Other

If you have another idea for a project that doesn't fit nicely into either of the above categories, let's chat on Slack or setup an appointment for a Zoom meeting to discuss further.

## Deliverables and Timeline

There are several deliverables for the project. Details on each are covered in the sections below.

1. An initial project proposal due by 11:55pm on **Wednesday, April 1st**. This is worth 10% of your final project grade.
2. Short project presentations will occur during our final week of classes; either on **Monday, April 27 or Wednesday, April 29**. This is worth 30% of your final project grade.
3. The completed final project is due by 5:30pm on **Friday, May 8th** (our regularly scheduled final exam time). This is worth 60% of your final project grade.

## Project Proposal

You project proposal is due by 11:55pm on **Wednesday, April 1st**. Submit to Moodle a proposal proposal containing the following information:

1. Your partner's name. If you are working alone, a short description of why you want to work alone (or if you would like a partner but have not found one). Be aware that I may assign some partners after you submit a proposal (e.g. several students want to work on similar projects).
2. A description of what you plan to do. You should provide enough details to convince me that what you plan to do is reasonable given the time that you have. You should talk about any implementation you plan to do or data you will need to find. As a guideline, I expect this to be ∼1 full page in length (single spaced).
3. At least three references you plan to use for researching the project (textbooks, research papers, etc.).
4. Any questions you have for me.
5. Your preferred presentation date. (Monday, April 27 or Wednesday, April 29)

### Proposal Grading

The proposal will count towards 10% of your final project score. The proposal will be graded on: (1) Whether the proposal includes all required pieces and sufficient details for each; and (2) whether the proposal sets out an appropriate plan for completing the project in the given time frame.

## Project Presentation

Each group will give a short (probably 10-15 minutes - I will let you know the exact length once the number of groups is determined) presentation to the class on your project on either Monday, April 27 or Wednesday, April 29. These will be conducted via a Zoom meeting. There are two main reasons that I am having you do project presentations.

1. It's a great opportunity to get some exposure to the wide variety of ideas and projects that your classmates are undertaking. I really hope that you find this exercise to be fun!
2. Talking about technical material is an important and useful skill. This is an opportunity to get more practice at learning how to do this effectively.

**Content: Your presentation should include the following components:**

- A clear overview of your project topic/goal. It is impossible to go into all the details in such a short presentation, so you will need to be strategic in your choice of what to present.
- **If you choose a project from Option 1**, your presentation should include the following specifics:
  - What algorithmic approach are you investigating?
  - What are the high-level important pieces about how it works? (You will not have time to go into all the details.)
  - What types of problems is it useful for solving?
  - How does it fit into what we have been talking about in this class?

  - Some demonstration of your analysis of this algorithmic approach. This could take many forms. A few examples are: (1) A plot that shows how the runtime changes with more data, or how accuracy is affected by a particular parameter; (2) Some overview of results found by running the approach on a particular dataset; (3) Comparison of this approach to others we have learned about. Etc.

- **If you choose a project from Option 2**, your presentation should include the following specifics:
  - What biological application or question are you investigating?
  - What computational approaches are used to investigate this application or question? You should describe at least one of these approaches at a high level that gives the main idea of how the approach works. Why are computational approaches necessary for solving this problem?
  - What are the main challenges (biological or computational) to solving this problem or answering this question?
  - How does it fit into what we have been talking about in this class?
  - You should present some results of computational approaches applied to your question or application (you do not have to have implemented or run the computational approach yourself).

### Presentation Grading

Your project presentation will be worth 30% of your final project grade. Specifically, your presentation will be graded on two axis: (1) whether you include the required information for your project; and (2) the clarity of your presentation. I highly suggest you practice your presentation in front of others beforehand. I also suggest that you practice presenting using Zoom by sharing your screen. Pro tip: you can create your own Zoom meeting (with only you in it), and practice your presentation!

## Final Project

You will need to submit your final project to Moodle by 5:30pm on Friday, May 8th. You must submit to Moodle the following:

1. For projects involving implementation, your working code and a detailed README. Please also include a small sample dataset that the code can be run on.

2. A short paper (no more than 5 pages single spaced, 12pt font) describing what problem you were working on, what you explored, your results (including figures), and some discussion. The goal of this paper is both to help me understand your project and also give you practice communicating your work to others. Your paper should also contain references to any relevant literature (these do not count towards your 5 page limit). If you think it is important to your project, you may include additional figures as part of an appendix that is not included in your 5 page limit.

### Final Project Grading

Your final project submission will be worth 60% of your final project grade. Specifically, your project will be graded on the following three categories: (1) **Demonstration of understanding of your topic.** This is assessed through both the writing and any submitted code. For

full credit, you should include how your topic ties back to class material (or explicit notion that it doesn't). (2) **Complexity and Scope.** For full credit you should set and achieve ambitious goals. (3) **Effective Technical Writing** - Your written paper should exhibit excellent organization and clarity. You should certainly spend time revising your writing before submitting your final product. For full credit your paper must be easily understood by a reader with a similar background, but no expertise in your topic area. Furthermore, your paper must be virtually free from grammatical and spelling errors.

## Optional Check-In

I won't require this, but I strongly encourage you to schedule an appointment with me to chat about your progress sometime before the due date. You can use office hours for this as well.

## Project Ideas

Here are a few suggestions of possible project ideas. If you want more info or extra resources on these, talk to me.

- Zucker Algorithm (or a different algorithm) that uses free energy to predict RNA secondary structure.
- Explore a technique for multiple sequence alignment (i.e. Carrillo-Lipman, MUSCLE).
- Explore other methods for constructing Phylogenetic trees or other related problems. Things you could look at are Probabilistic/Bayesian methods, Coalescent trees (e.g. Kingman's coalescent), or consensus phylogenetic trees (Strict, Majority-rule, Adams, etc.)
- Explore a variation on genome assembly. For example, double stranded genome assembly. Here is a paper that talks about a method to do this.
- Structural variation detection is the process of identifying portions of DNA that have been deleted/duplicated/inverted compared to a reference genome. Explore a technique for structural variation detection. Here is a review paper that talks about the overall problem and lists a bunch of different algorithms/programs.
- We are also able to sequence RNA. The computational problems here are slightly different. For example, if you want to align RNA sequenced reads to a genome, you would need a special tool (e.g. TopHat).
- One of the big Computational Biology conferences is called RECOMB. Here is a site with the names of the papers from last year, you could pick one to dig into: `https://recomb2019.org/materials/#Acc` (you will have to search for each paper to find its text).
- The journal *Bioinformatics* often has good and interesting work. Look through the papers on their website here to see if anything catches your eye: `https://academic.oup.com/bioinformatics`.
- Find a tool and work backwards to find the corresponding paper in order to understand the tool. This website contains a great list of computational biology tools that are out there, organized by category: `https://omictools.com/`. (Note: This site appears to require you to create an account first).

- Find an interesting paper somewhere else? Dig into figuring out what they are doing and why? Is code available? Great - try it out!

## Available Data

There are lots of different sources for biological data and types of biological data, if you need that for your project. Be wary though. If your project requires specific data, you will want to make sure you can get access to the right kind when you submit your proposal.

This blog overviews all sorts of different types of sequencing data (you could even just learn about one of these in your project). Here are a few places you could look to find interesting datasets, if that is something you need for your project.

- You can download DNA, RNA and Protein sequences for different species from here. This is also just a good site to poke around and see what is there.
- The UCSC Genome Browser allows you to query and download different kinds of data about genomes in a nice tab delimited format.
- Here are a few resources for data related to cancer: (1) Broad Institute, (2) Genomic Data Commons
- Look for a paper that describes some interesting data. Many times the paper will have a spreadsheet as part of the supplemental material that contains some of the raw data.
- Often times software packages released along with a paper will contain small sample datasets that you can use when running the code too. The paper should contain a link to the code if it exists.