Biological Data Analysis (CSE 182) : Assignment 1

Logistics. Please use Gradescope to submit.

Python/Scripting language basics You are free to use any programming language to do your assignment, but we will only support Python. It is the language of choice for data analysis. The goal of this assignment is to get comfortable with simple tasks that will come in handy later on. Please send email to your TA or instructor if you have trouble getting started.

1. Choose a Platform, Scripting Language and Editor. Write a program that outputs "Hello Bioinformatics" when run (5pt).

2. Complete the 'First Day #FinAid survey' on Canvas by navigating to the 'Quizzes' tab. The survey is not graded, but participation is mandatory. The sharing question is free-form. For example, you can provide your standing in the program, department, preferred pronouns, career goals, hobbies, or any thing you would like to share (5 pt).

3. The Fasta format is a standard format used to represent biological sequence data. Google 'fasta format' to understand the format. Download the multi-fasta sequence db from the course home page. Write a program *cat* that reads each line and prints the header line of each sequence in the database followed by the length of that sequence (20pt).

4. Write a program *filter* that extracts all of the mouse and rat sequences from this file. The output format should be multi-fasta, similar to the input, but containing exactly 60 characters on each line, except when the sequence ends. Note that different IDs in the header might represent the same species (20 pt).

5. **Database index creation** Read the data file and create two *index files.*

   (a) The first file *data.seq* contains the concatenation of all of the sequences from each file with no headers, and no newline symbols. Insert the special symbol '@' between any two sequences (14pt).

   (b) The second file *data.in* contains a line with two terms for each sequence. The first term is the 'gi number' for the sequence, and the second term is the offset in "data.seq" where the sequence starts (14pt).

6. Write a program *getSeq* that takes a short sequence as query, and, using the index files you created in the previous problem, returns the gi number of the database sequence containing the query string. What do you get when you query for MHIQITDFGTAKVLSPDS (20pt)?
   (**Non-credit**. Think of using binary search for this problem, and report your thoughts on why this indexing is useful.)

7. How much time did you spend on this assignment? Who did you ask for help (2pt)?