CMPSC 300
Bioinformatics
Spring 2021

Lab 2 Assignment: Writing Python code to complete an Analysis of DNA

Submit deliverables through your assignment GitHub repository and complete the Google Form.

Objectives

There are many different kinds of tools of analysis being developed each week for research in Bioinformatics. However, in spite of the diverse types of analyses which these tools are able to address, an investigator must still develop her own tools for research due to the nature of the work. In this case, skill in Python programming for developing tools is often necessary for researchers who are working in areas where no tools currently exist.

To strengthen your understanding of DNA structure, in relation to programming, you are to create code to input a string of DNA for manipulation and to return a complementary sequence of DNA (i.e., the pairing strand of the sequence).

Clone Your Assignment Repository

In this section, we will be using `Git` commands. It is suggested that the reader refer to online searches for help. For example, GitHub provides good documentation at the following link; `https://git.github.io/htmldocs/git.html`.

In many cases, you will be given a new repository containing assignment materials and you will save your files in this assignment repository as you continue to work on them. Copy and paste the assignment repository cloning command into your terminal to create your assignment repositories. Be sure to place your assignment repositories in a directory such as `cs300/` to keep your class materials organized by class.

Today's assignment repository can be found at the below link to a GitHub Classroom repository. Here you will work on your assignment and then push your work to the cloud where the instructor will be able to view your work for grading. Often, there will be files in your assignment repositories which you are to edit before you submit them by using the below commands for `git`.

https://classroom.github.com/a/VnF9bp9U

To use this link, please follow the steps below.

- Click on the link and accept the assignment
- Once the importing task has completed, click on the created assignment link which will take you to your newly created GitHub repository for this lab,
- Clone this repository (bearing your name) and work locally
- As you are working on your lab, you are to commit and push regularly. The commands are the following.
  - `git add -A`
  - `git commit -m ‘‘Your notes about commit here’’`
  - `git push`

After you have pushed your work to your repository, please visit the repository the GitHub website (you may have to log-in) to verify that your files were correctly sent.

DNA and Python Basics

Bioinformatics requires the use of computational techniques to solve biologically-related problems. Therefore, in this course it is essential for you to have some basic understanding of both biological and computer science concepts related to Bioinformatics before we approach Bioinformatics problems.

Seemingly all goals of Bioinformatics research depend on automation for completion. In many cases, bioinformaticians are required to create their own programs and tools in order to complete their cutting-edge work. In this lab, you will also create a basic tool to help in your future work in the field.

Part 1: Activity

Please complete the below activity at the GitHub form shown below.

https://forms.gle/HTm9h5pHtDAo2a6i7

Part 2: Getting to Know Python

Python Help

Before we can start writing Python programs for Bioinformatics solutions, you need to get comfortable with the structure and the syntax of this programming language. Just like with any natural or programming language, you need to learn various syntactical and semantic rules when writing in Python.

If you feel that your skill in Python3 needs some improvement, you are invited to read Chapters 1 and 2 in the "Think Python" textbook which is available at, `https://greenteapress.com/wp/think-python/`. To get started with Python, please complete Sections 3 and 4.1-4.5 in the Python Tutorial, which can be found in: `https://docs.python.org/3/tutorial/index.html`. Note: if you are proficient with Python then please consider lending some help to your fellow classmate, if asked.

DNA Base Counter program

You are to write a Python program that:

Inputs: A DNA string of around 1000 bases.
Returns: Four integers (separated by spaces) counting the respective number of times that the bases,'A', 'C', 'G', and 'T' occur in the sequence. In addition, you are to add the pair-counts of "AT", "TA". "GC" and "CG". Your programming is to simply go though the string and count these combinations. Your program is to output these statistical values for evaluation by the user.

Filename:`baseCounter.py`

Sample Input: A DNA string of arbitrary size. Note; this sequence can be loaded as a file or pasted into the terminal at run-time.

Sample Output:

```
The baseCounter program to count
the number of bases in a sequence of DNA
--------------------------------------------------------------------
The sequence is: AAAACCCGGTTGCGTAAT
 A count: 6
 T count: 4
 C count: 4
 G count: 4
 AT count: 1
 TA count: 1
 GC count: 1
 CG count: 2
----------------------------------------
```

Complementary DNA program

You are to write a Python program that:

Inputs: A DNA string of arbitrary size.
Returns: The complementary sequence of the inputted strand and the reverse of this complement.

Note: the reverse is necessary to account for the opposite reading-direction of the complement strand.

Filename:dnaComp.py

Sample Input: Sequence of DNA. Note this sequence can be loaded as a file or pasted into the terminal at run-time.

Sample Output:

```
A program to return the complementary
sequence for an inputted DNA string
----------------------------------------
Enter the sequence    : AAAACCCGGTTGCGTAAT
Original sequence     : AAAACCCGGTTGCGTAAT
Complementary         : TTTTGGGCCAACGCATTA
Reverse Complementary : ATTACGCAACCGGGTTTT
```

Part 3: Ethical Portion

Locate the article in your repository by Michael Specter entitled, ”Rewriting the code of life” [1].

The reference to the article is,
`https://www.newyorker.com/magazine/2017/01/02/rewriting-the-code-of-life?reload=true`

Please complete the questions found in the file, `writing/reflections.md` You should see the instructor if you have questions about assignment submission.

Required Deliverables

1. **Python code**: Your working Python code to complete the above coding objective.
   - File: `baseCounter.py`
   - File: `dnaComp.py`
2. **Activity**: Although you are not to submit this portion of work to your GitHub repository, you are to complete the activity using the Google Form before the due date of the lab.
3. **Ethical reflection**: You are to complete the ethical reflection which is discussed above. Please locate the Markdown file, `writing/reflection.md` in your cloned assignment directory. Edit this file using Markdown code to reflect your work. If you need help on this formatting, visit the below link for help.
   - **Markdown Tidbits**:
     `https://www.youtube.com/watch?v=s-oSuHFVnR4`

- **Markdown Cheatsheet**:
  `https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet`

4. Do not forget to use the above `git` commands to push your work to the cloud for the instructor to grade your assignment. Please see the instructor if you have any questions about assignment submission.

Grading

The grade that you receive for this lab assignment will be based on the following:

- 40% Complete and correct Python programs.
- 20% Clearly written reflection document that adequately responds to each question in the document.
- 30% Complete and correct responses for the activity using the provided Google Form.
- 10% Complete GitHub Actions CI build-pass corresponding to all the GatorGrader checks passing.

References

[1] M. Specter, "Rewriting the code of life," *The New Yorker*, vol. 2, 2017.