Advanced Programming 2023/2024

Project Specification

Andrea Giovanni Nuzzolese

Input

The input consists of the complete genome sequence of WHCV has been deposited in GenBank under accession MN908947. More specifically we focus on a FASTA file, i.e. `GCA_000864885.1_ViralProj15500_genomic.fna`, which looks like the following:

```
>AY274119.3 SARS coronavirus Tor2 complete genome
ATATTAGGTTTTTACCTACCCAGGAAAAGCCAACCAACCTCGATCTCTTGTAGATCTGTTCTCTAAACGAACTTTAAAAT
CTGTGTAGCTGTCGCTCGGCTGCATGCCTAGTGCACCTACGCAGTATAAACAATAATAAATTTTACTGTCGTTGACAAGA
...
GGAGTACGATCGAGGGTACAGTGAATAATGCTAGGGAGAGCTGCCTATATGGAAGAGCCCTAATGTGTAAAATTAATTTT
AGTAGTGCTATCCCCATGTGATTTTAATAGCTTCTTAGGAGAATGACAAAAAAAAAAAAAAAAAAAAAAA
```

The FASTA format is a text-based format for representing either nucleotide sequences or amino acid (protein) sequences, in which nucleotides or amino acids are represented using single-letter codes. A sequence begins with a greater-than character (">") followed by a description of the sequence (all in a single line). The lines immediately following the description line are the sequence representation, with one letter per amino acid or nucleic acid.

General Goal

Write a program able to:

- parse the FASTA file and generates a vector representation of the sequence;
- generates a DNA object from the vector representation of the sequence;
- allows transcriptions, translations, amino acid chaining, and protein identification;
- present the user with relevant outcomes.

The program must be designed and implemented in Python by using the Object-Oriented paradigm and by applying its associated fundamental concepts (i.e. encapsulation, inheritance, data abstraction, and polymorphism) properly.
The design must be carried out by drawing CRD cards and class diagrams.
Design and implementation choices must be explained into the final project document.
Additionally, the dataset management and presentation must rely on Pandas (and NumPy) and Flask, respectively.

Detailed Instructions

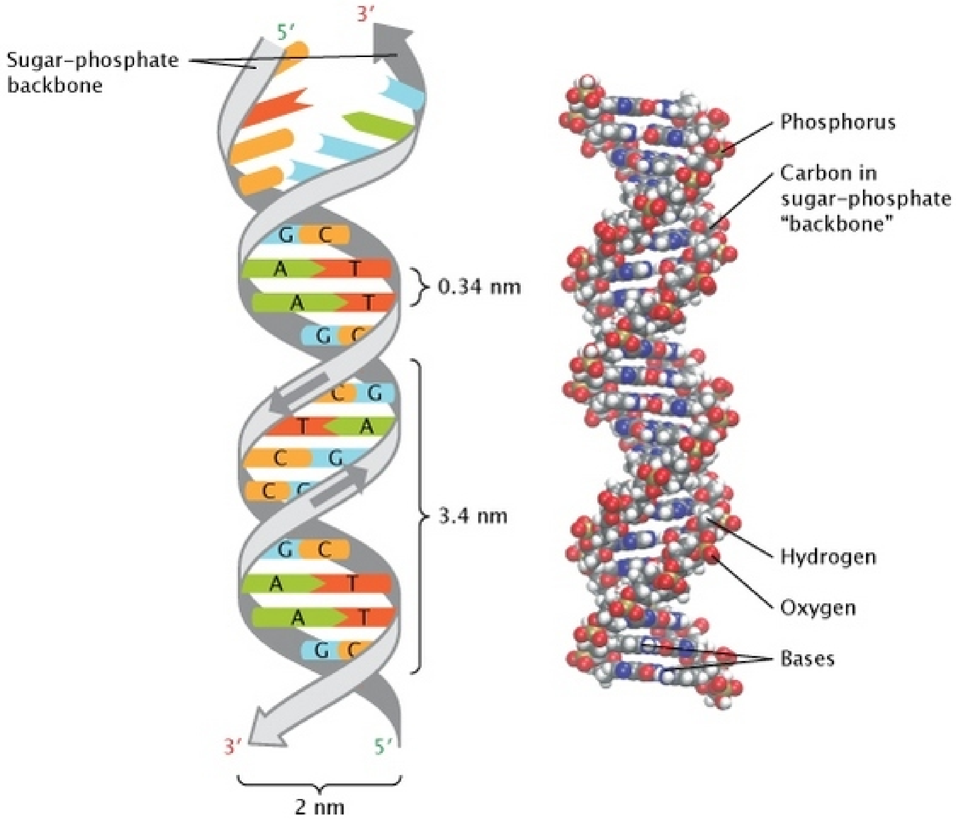
The program is composed of three main parts, consisting of:

- **[Part 1]** the part with classes and their associated methods forreading the FASTA file and generating a Dataset that contains the sequence. The FASTA file cannot be accessed with any library (e.g. Biopython) but Pandas.

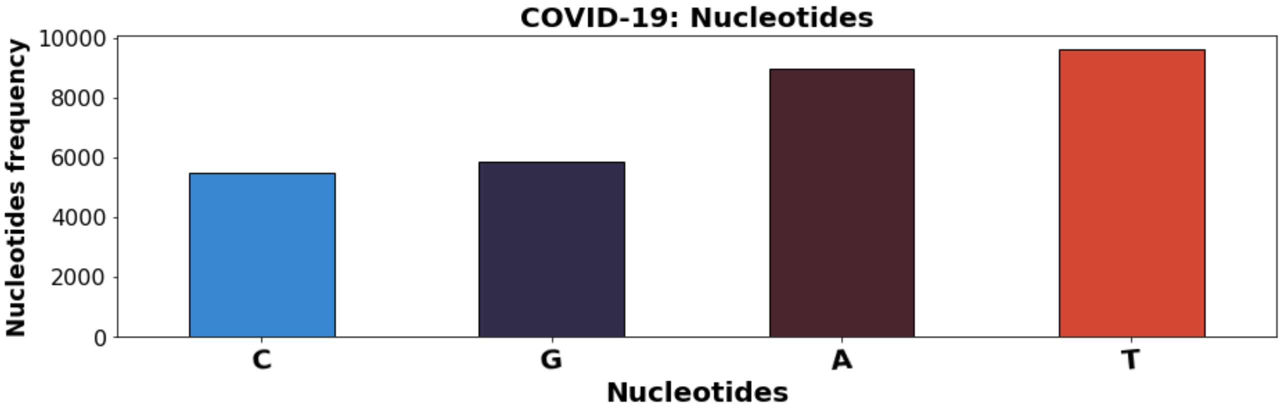
- **[Part 2]** the Dataset can be used for generating different Genetic Entities. Some of those Genetic Entities are Sequences, whist other are Organic Elements.

  Possibile Sequences are the DNA, mRNA, Amino Acid Chains and Proteins. Instead Organic Elements can be Nucleotids, which are the basic building blocks of nucleic acids (mRNA and DNA), or Amino Acids, which are the building blocks of protein. A nucleotide consists of a sugar molecule (either ribose in RNA or deoxyribose in DNA) attached to a phosphate group and a nitrogen-containing base.

  The information in DNA is stored as a code made up of four chemical bases: adenine A, guanine G, cytosine C, and thymine T (figure below). The order, or sequence, of these bases determines the information available for building and maintaining an organism, similar to the way in which letters of the alphabet appear in a certain order to form words and sentences.



  A specific type of Sequence, which is the DNA, can be directly derived from the Dataset. More specifically the DNA is a sequence of Nucleotides. A sequence of Nucleotides is something like a NumPy array that contains Nucleotides as items. DNA sequences, like any other sequence, allows to get the frequency of its components. An example of the frequency of the Nucelotides in a DNA sequence is reported in the figure below.



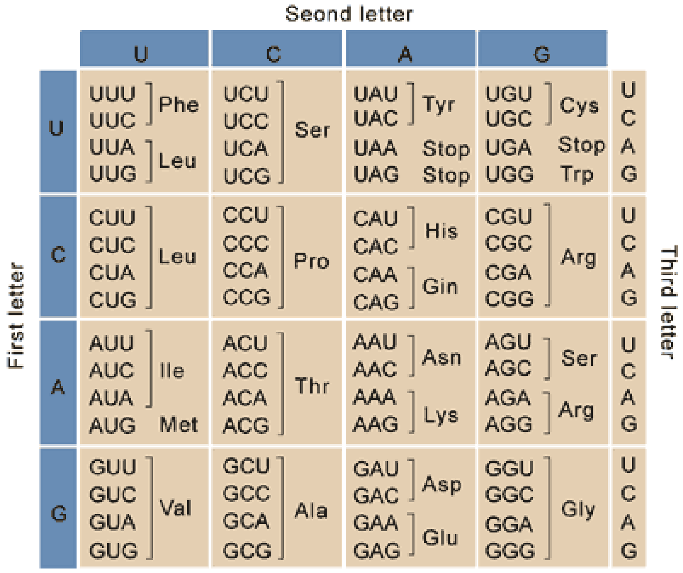
  Additionally, all Sequences provides transcriptions even if a transcription is not effective for the specific Sequence type. For example, a DNA sequence is transcribed to a mRNA sequence, but a mRNA when transcribed returns the same mRNA sequence. Amino Acid Chains and Proteins allow transcriptions that do not affect theis sequence as well.

The transcription is the first step in gene expression. In the case of DNA it involves copying a gene’s DNA sequence to make an RNA molecule.

Basically the mRNA is a copy of our DNA. However, in RNA, a base called uracil (U) replaces thymine (T) as the complementary nucleotide to adenine (that’s the only difference, T is replaced by U).

Instead, translation is the process that takes the information available in mRNA and turns it into an Amino Acid Chain. Translations are possible for mRNA only.

It is essentially a translation from one code (nucleotide A T C G sequence) to another code (amino acid sequence). How does this translation happen? As in any language, we need a dictionary for translation, in this case the amino acid dictionary is the table below. The nucleotides are read in groups of three “AUG GCC CAG UUA …”. Each triplet is called a codon and codes for a specific Amino Acid.



Seond letter

First letter	U		C		A		G		Third letter
U	UUU UUC UUA UUG	Phe Leu	UCU UCC UCA UCG	Ser	UAU UAC UAA UAG	Tyr Stop Stop	UGU UGC UGA UGG	Cys Stop Trp	U C A G
C	CUU CUC CUA CUG	Leu	CCU CCC CCA CCG	Pro	CAU CAC CAA CAG	His Gln	CGU CGC CGA CGG	Arg	U C A G
A	AUU AUC AUA AUG	Ile Met	ACU ACC ACA ACG	Thr	AAU AAC AAA AAG	Asn Lys	AGU AGC AGA AGG	Ser Arg	U C A G
G	GUU GUC GUA GUG	Val	GCU GCC GCA GCG	Ala	GAU GAC GAA GAG	Asp Glu	GGU GGC GGA GGG	Gly	U C A G

There are 61 codons for 20 amino acids, and each of them is “read” to specify a certain amino acid out of the 20 commonly found in proteins.

One codon, AUG, specifies the amino acid methionine and also acts as a start codon to signal the start of protein construction.

There are three more codons that do not specify amino acids. These stop codons, UAA, UAG, and UGA, tell the cell when a polypeptide is complete. All together, this collection of codon-amino acid relationships is called the genetic code, because it lets cells “decode” an mRNA into a chain of amino acids.

It is worth to mention that not all the amino acids sequences are proteins. Only the sequences with more than 20 amino acids code for functional proteins. The short amino acid sequences are Oligopeptides and have other functionalities. Here, we will focus on the chains with more than 20 amino acid chains and we call them Proteins.

Hence, Amino Acid Chains has a special operation that allows them to get a matrix in which each chain is associated with a length. From this matrix is then possible to distinguish between Proteins and Oligopeptides.

- **[Part 3]** the part that implements the Web-based user interface (UI). Such a UI provides a list of choices, where each choice enables an analytical objective (cf. Part 2).

  Namely, it provides the following interactions:

  - Generate a DNA from the Dataset;
  - Visualis the statistics of a Sequence, i.e. its different Organic Elements and their count, frequency, max and min;
  - Transribe DNA to mRNA;
  - Translate mRNA to Amino Acid Chain;

    - Get the list of Oligopeptides sorted either ascending or descending with respect to the chain length;
    - Get the list of Proteins sorted either ascending or descending with respect to the chain length;
    - Visualise a single Oligopeptide;
    - Visualise a single Protein;

The three parts should be implemented as three separate components, i.e. three Python modules consisting in three separate files.

Project Document

The software **must be extensively described** into a project document with:

- text in English;
- CRC cards;
- UML diagrams in order to point out what the structure of classes is;
- Design and implementation choices.

  The UML diagram must be described in the text, as **images are not self-explaining**.

Additional libraries can be used. However, **no Python library for processing genomic sequences, such as Biopython, can be used.**

Project Delivery

The delivery of the project must be done on a GitHub repository owned by the project team and shared with the teacher.