National University of Singapore
School of Computing
CS1010S: Programming Methodology
Semester I, 2024/2025

Mission 05
DNA Translation

Release date: 21st October 2024
Due: 3rd November 2024, 23:59

Required Files

- codon_mapping.csv
- dna.txt
- mission05-template.py

General Restrictions

- No importing additional packages that have not been provided for you.

Information

In this mission, we will investigate how dictionaries can help us act as a look-up table to handle a series of DNA sequencing tasks. We will study some simple DNA translation techniques and how using dictionaries can help speed-up certain operations.

We start by introducing some terminology in Molecular Biology, and then proceed to attempt to replicate DNA, transcribe DNA to RNA, and translate RNA into protein.

This mission consists of **four** tasks.

Note: If you find this mission challenging, Recitation 8 will give you more familiarity with various terminology via the problem of identifying DNA transcription region. You may find the higher order functions `map` and `filter` to be useful in solving some of these tasks.

Background

The central dogma of molecular biology states that DNA encodes genetic information to produce protein. This information is first transcribed into RNA. RNA is then translated into a protein. In short, the flow of information is DNA → RNA → protein (Figure 1).

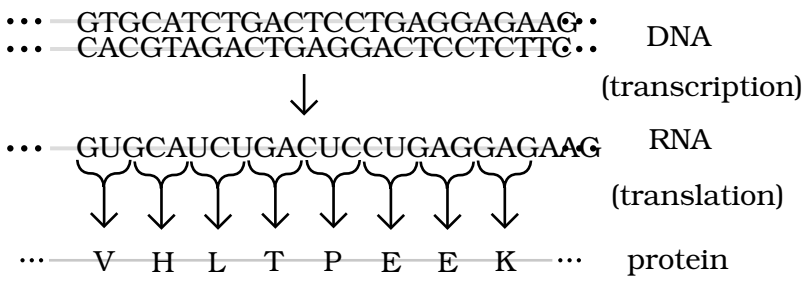

Figure 1: Central Dogma of molecular Biology
`commons.wikimedia.org/wiki/File:Genetic_code.svg`

DNA is made from FOUR different chemical building blocks called nucleotides. Each nucleotide consists of a sugar molecule attached to a phosphate group and a nitrogen-containing base. The bases used in DNA are: adenine (A), cytosine (C), guanine (G), and thymine (T). To form a strand of DNA, nucleotides are linked into chains. A strand of DNA contains genetic instructions to produce proteins in living things.

RNA also contains FOUR different bases. Three of these are the same as in DNA: adenine, guanine, and cytosine. RNA contains uracil (U) instead of thymine (T).

Protein is made out of TWENTY amino acids, forming a polymer chain. Each amino acid is encoded by a sequence of THREE RNA bases, also called a codon. This leads to 64 possible combinations of codons. Of these 64 codons, 61 represent amino acids, and the remaining 3 represent stop signals, which trigger the end of protein synthesis. The full list of codons and their respective amino acids is given in the file `codon_mapping.csv`.

Task 1: DNA Replication (3 marks)

Two DNA strands forms a double helix structure, and each nucleotide within each strand of DNA is paired with a complementary nucleotide. Complementary bases attach to one another (A-T and C-G). This means a single strand contains the information required to synthesize a new complementary strand during replication.

One point to note is that each strand of DNA has two ends, a 5' end and a 3' end. Paired DNA strands have an antiparallel structure—one strand of the helix runs in the 5' to 3' direction while the other complementary strand runs in the 3' to 5' direction. (Figure. 2)

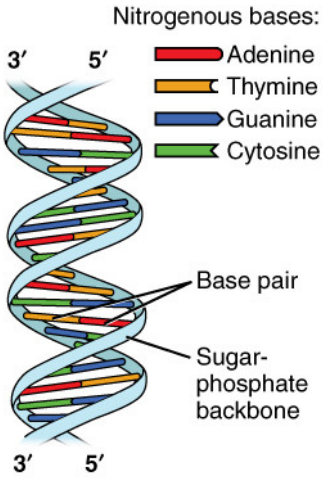

Figure 2: Anti-parallel DNA strands form a double helix. *Image from wikimedia commons.*

Write the function `replicate`, that takes as input an uppercase string representing a DNA strand in the 5' to 3' direction, and returns the complementary DNA strand, also represented in the 5' to 3' direction.

RESTRICTIONS: You must use the dictionary `dna_base_pairings` in your solution.

Sample execution:

```
>>> replicate("AAATGC")
    'GCATTT'
>>> replicate("ATTGGGCCCC")
    'GGGGCCCAAT'
```

Task 2: DNA to RNA transcription (4 marks)

Two DNA strands forms a double helix structure, and each nucleotide within each strand of DNA is paired with a complementary nucleotide. Complementary bases attach to one another (A-T and C-G). This means a single strand contains the information required to synthesize a new complementary strand during replication.

RNA is transcribed from DNA. During the transcription process, an RNA polymerase binds to a sequence of DNA. The bound RNA polymerase separates the DNA strands, providing the single-stranded template needed for transcription. The transcription process reads the template DNA strand from 3' to 5' direction. This results in a new RNA molecule that grows from 5' to 3' direction. The RNA transcript carries the same information as the complementary strand of DNA, but contains the base U instead of T.

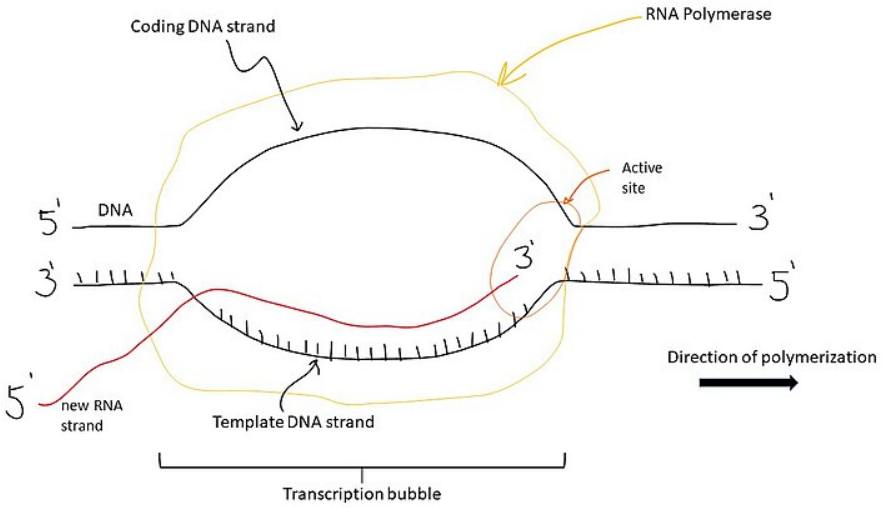

Figure 3: Simplistic view of a template DNA strand undergoing transcription to RNA
commons.wikimedia.org/wiki/File:Transcription_bubble.jpg

Write the function `transcribe`, that takes in an uppercase string representing a DNA template strand in the 5' to 3' direction, and returns a new RNA strand produced by the transcription process, also in the 5' to 3' direction.

Then, write the function `reverse_transcribe` which reverses this process.

Sample Execution:

```
>>> rna = transcribe(dna) # dna is read from dna.txt
>>> rna[0:10:1]
    'AAUAGUUUCU'
>>> transcribe("AAATGC")
    'GCAUUU'
>>> transcribe("ATTGGGCCCC")
    'GGGGCCCAAU'
>>> reverse_transcribe(transcribe("AAATGC"))
    'AAATGC'
>>> reverse_transcribe("GGGGCCCAAU")
    'ATTGGGCCCC'
```

Task 3: Codon-Amino acid mapping (3 marks)

Now that we have the transcribed RNA strand, we can begin translation. The message encoded in RNA is read in non-overlapping three-character groups called codons. This gives us a total of 64 possible codon combinations.

Since there are only 20 different amino acids but 64 possible codons, most amino acids are indicated by more than one codon. This phenomenon is known as redundancy or degeneracy, and it is important to the genetic code because it minimizes the harmful effects that incorrectly placed nucleotides can have on protein synthesis.

Each codon maps to a specific amino acid, which then forms the building blocks of proteins. Therefore, by translating the RNA strand, we can obtain the amino acid sequence of the protein being produced.

The first 5 codons and their mappings are shown here:

codon	amino acid	3-letter abbreviation	1-letter abbreviation
AAA	Lysine	Lys	K
AAC	Asparagine	Asn	N
AAG	Lysine	Lys	K
AAU	Asparagine	Asn	N
ACA	Threonine	Thr	T
...	...	...	...

The full list of 64 codons and their respective amino acids is found in `codon_mapping.csv`.

Write the function `get_mapping`, that takes in a CSV filename. The associated file will contain a header row and 4 columns of data. The function should return a dictionary with the first column as the dictionary-keys, and the last column as the corresponding values.

RESTRICTIONS: Your function must work for **any** arbitary CSV file of the same format. You may wish to use the `read_csv` function provided to you.

Sample execution:

```
>>> codon2amino = get_mapping("codon_mapping.csv")
>>> codon2amino["ACA"]
    'T'
>>> codon2amino["AUU"]
    'I'
>>> codon2amino["CUC"]
    'L'
>>> codon2amino["ACU"]
    'T'
>>> codon2amino["UAG"]
    '_'
>>> codon2amino["UGA"]
    '_'
```

Task 4: RNA to Protein translation (6 marks)

An RNA strand is a template to create a chain of amino acids that form a polymer chain, the protein. During translation, the RNA strand is read in non-overlapping groups of 3 bases. Each group of 3 bases corresponds to a codon, and each codon maps to a particular amino acid.

There are FOUR special triplets to take note of: the start codon `"AUG"`, and the stop codons `"UAA"`, `"UAG"` and `"UGA"`. A protein begins translation from the first start codon encountered (Figure 4) and only stops when the first stop codon is reached (Figure 5), in the 5' to 3' direction.

Write the function `translate`, that takes in an uppercase string representing a RNA strand in the 5' to 3' direction, and returns a new protein (represented as the string of 1-letter amino acid abbreviations) produced by the translation process. All valid proteins start with `"M"` and end with `"_"`. If no valid protein exists, then return `None`.

Sample execution:

```
>>> translate("AUGUAA")
'M_'
>>> translate("AGAGAUGCCCUGAGGG")
'MP_'
>>> translate(rna)
'MANLT...HLTTY_'
```

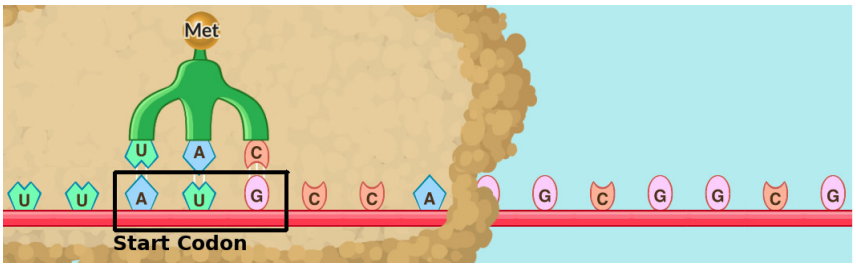

Figure 4: Translation begins at the first `"AUG"` start codon encountered.

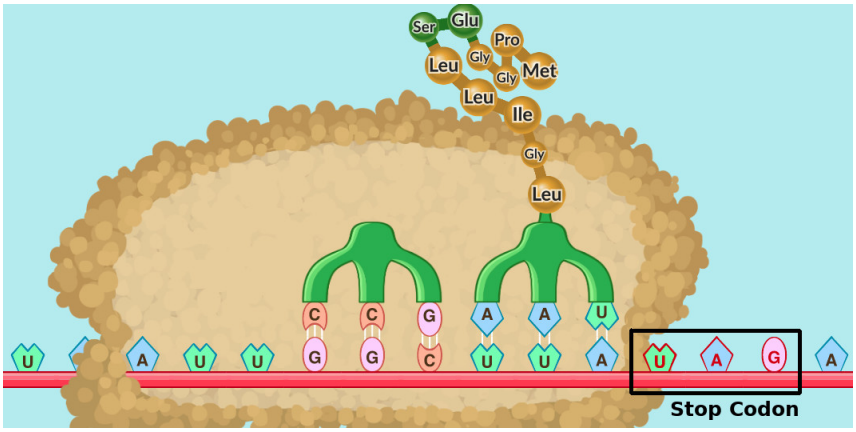

Figure 5: Translation ends at the first stop codon (`"UAA"`, `"UAG"` or `"UGA"`) reached.

To see the entire process from DNA to protein, refer to this link by **Concord Consortium**:
`lab.concord.org/embeddable.html#interactives/sam/DNA-to-proteins/1-dna-to-protein.json`