20.9 Program 4b: DNAtoProtein Converter

Objectives

* File input and output
* Rethrowing exceptions
* Code in an IDE and upload to grading program
* Use documentation and code structure according to class guidelines

Program Description

Template Provided

**DNA to Protein**

*Translation theory*: DNA → RNA → Protein

**DNA**: DNA is a discrete code physically present in almost every cell of an organism. We can think of DNA as a character with four possibilities to choose from. These characters are A, C, G, and T. They stand for the first letters with the four nucleotides used to construct DNA. The full names of these nucleotides are Adenine, Cytosine, Guanine, and Thymine.

**RNA**: Messenger RNA (mRNA) is a temporary copy of the sequence of the gene that codes for the protein. mRNA transcribes the antisense strand of DNA (converts T to A, A to U, C to G, G to C).

**Protein**: Consists on amino acids linked by amide bonds. Each triplet codon of mRNA translates to protein (amino acid).

This is analogous to how our Java program works. We have the code in a ".java" file (DNA), which compiles into a ".class" (mRNA) and running the ".class" file gives us our output (Protein).

For this assignment, you need to name your java file as “DNAConverter.java”. Inside your Java class, you need to write 2 functions. Please name your function as described in the problem descriptions below.

* **DNA2RNA**

DNA2RNA() takes two parameters, String DNAFile, and String RNAFile. String DNAFile is the path of DNA.txt (provided), and String RNAFile is the path of RNA.txt (which should be generated in this method). This method shall read DNA.txt and transcribe this into RNA.txt. You should throw a NoSuchElementException("Not a DNA character") if you encountered a character that doesn’t belong to DNA.

* **RNA2Protein**

void RNA2Protein() takes three parameters, String RNAFile, String ProteinFile, and String RNA2ProteinTable. String RNAFile is the path of RNA.txt that you generated. String ProteinFile is the path of Protein.txt which should be generated in this method. RNA2ProteinTable is the path of RNA2Protein.txt, which is a reference for genetic code table. This method shall read RNA.txt and translate this into Protein.txt with help from RNA2Protein.txt. If the length of a line in RNA.txt is not a multiple of 3, throw a RuntimeException("Invalid RNA length").

Additionally, you are allowed to create helper methods.

Example:

Antisense strand of DNA: TACCGGACCTGAAGT

mRNA: AUGGCCUGGACUUCA

Note: (converts T to A, A to U, C to G, G to C)

Protein (amino acid): MKACS

Note: (converts using the shorter version of RNA2ProteinTable, longer table is present in the file which will be used for testing)

RNA2ProteinTable

AUG M

GCC K

UGG A

GCA T

ACU C

UCA S

GGG S

Some useful methods:

* Scanner: nextLine(), hasNextLine(), next(), hasNext()
* PrintWriter: print(), println(), flush()
* StringBuilder: append(), toString()
* String: charAt(), split(), substring()
* HashMap: put(), containsKey(), get()

**Note: If you fail the (a) part of the test case, you will also miss the (b) part of the test case. The (a) part of the test case converts the DNA to RNA, and the (b) part uses the RNA generated and converts it to a Protein.**

**Note: I cannot emphasize how much important it is to flush() your PrintWriter before the end of the program. I have seen a lot of students encountering the issue of not seeing any output in a file for P3 and a buffer flush() solved that.**

Basically, when you print using PrintWriter, it stores it in a buffer in memory. Only when the buffer is full, the data will be written to the file. This is to prevent multiple small data writes to the disk, which is very inefficient. For most cases in CS 108, we do not work with huge test cases, which means we only output a few bytes to the file. The flush() function would send the data in the buffer to the file **immediately**, however small the size of the buffer is at present. When you inspect the PrintWriter documentation provided by Java, calling PrintWriter(File) (or PrintWriter(String)) *"Creates a new PrintWriter, without automatic line flushing, with the specified file (file name)."* This means that if the program ends, it would not auto flush the data in the buffer that you thought would be printed out.

[Source](https://docs.oracle.com/javase/7/docs/api/java/io/PrintWriter.html#constructor_summary)

Turning In Procedure

* You are required to submit the DNAConverter.java in Zybooks which will be autograded.
* The automatic grading program is very specific. If you feel you have the correct solution but are not receiving full credit, please
  + Carefully review the output -- you might need to scroll all the way to the right to find what is wrong with a particular output.
  + Verify you have the correct names for the program itself and all methods.
  + Check your calculations by hand: was there a logic error?
  + Review the requirements: did you miss a step? misinterpret a requirement?
  + If all these check out, contact the T.A. for assistance.