CS 490/CS 590 Project I: Detecting a Disease Allele for Cystic Fibrosis

Description: This project is based off of the Guided Programming Project of Chapter 2 whose purpose is to detect missense mutations in an allele of the CFTR gene. It is a fairly straightforward sequence manipulation project, in which you need to implement **transcription** and **translation** for two given DNA sequences in FASTA format.

Detailed Specifications: Your program should allow any two DNA sequences saved in FASTA format files, allowing the user to input the filenames from the prompt. You should clear out the header line and any white space and return characters from the sequences so that they are pure nucleotide sequences, namely strings over alphabet {A,C,G,T} or {a,c,g,t}. Because you allow lowercase and uppercase inputs, it is useful to convert both to uppercase immediately. Due to simplifications involved in this first project, however, you may immediately reject the inputs if the nucleotide sequences are of different lengths. Otherwise, you are to continue with transcription and translation.

As the motivating biological problem involves detection of a possibly harmful mutation for a gene, you may assume that one of the sequences (say the first one) represents the "wildtype" exon sequence of the gene responsible for an important protein while the other sequence (say the second one) represents someone's allele for that gene. In particular, based on Chapter 2, the gene in question codes for the ***Cystic Fibrosis conductance Transmembrane Regulator*** protein, and the allele you are testing is Mary's, to see if she is a carrier for the cystic fibrosis disease. Although both are exon sequences representing the same chromosomal regions, there are four possibilities for the strand and orientation of each string: **template** versus **non-template** strands, and **5' to 3'** versus **3' to 5'** orientations. Remember that a template strand is the complement of the non-template (coding) strand, and the different orientations represent reversals. RNA is the interface between the DNA and protein, and the RNA corresponding to a DNA region is identical to the **non-template** strand in the **5' to 3'** direction except for every T being replaced with a U. **Transcription** itself is the process of making that RNA from the DNA. Recall the cases for transcription of a DNA sequence **S** into its corresponding RNA sequence **R**, letting **S'** denote the string that is identical to **S** except that it has every T of string **S** replaced with a U:

- If **S** is a coding (non-template) strand in the 5' to 3' direction: **R** is identical to **S**'.
- If **S** is a non-template strand in the 3' to 5' direction: **R** is the **reverse** of **S**'.
- If **S** is a template strand in the 3' to 5' direction: **R** is the complement of **S'**.
- If **S** is a template strand in the 5' to 3' direction: **R** is the **reverse** of the **complement** of **S'.**

Certainly, if the RNA of the wildtype and the RNA of the person's allele are identical, then there is no mutation. But it isn't necessary for the RNA sequences to be identical in order for there to be an absence of **missense mutations** that are the potentially harmful kinds**.** it is the differences in the **amino acid sequences** (and proteins are amino acid sequences) resulting from each RNA sequence that creates a phenotypic difference. Because there are 64 possible codons (three consecutive nucleotides) and only 20 amino acids, sometimes different codons encode the same amino acid, resulting in different RNA sequences that also encode the same protein. This process of RNA encoding amino acid sequences is **translation**. And, you will have to hard-code the translation mapping according to Figure 2.4 on page 27

of your book, using the same letters representing each amino acid. Finally, to detect a missense mutation, you will need to also translate each RNA sequence into the corresponding amino acid sequence. You are allowed some simplifying assumptions in this project in that you can start your translation at the first nucleotide and end at the last nucleotide, thus not needing to consider different frames. So, the translation is fairly straightforward, as you proceed through the consecutive, non-overlapping codons, concatenating the amino acid it encodes into your amino acid sequence. The result of the translation of each RNA sequence is two amino acid sequences, which you may now compare for a missense mutation!

There are no further "algorithmic" specifications for this project as it is algorithmically straightforward. Nor am I enforcing the use of any particular data structures, although hashes and maps are handy.

What to turn in: You must turn in a single zipped file containing your source code, a Makefile if needed for compilation, and a README file indicating how to execute your program.

Your program must be written in C/C++ or Java and compile using an open source compiler such as g++, gcc, or javac.

I will demo a Perl implementation of this project in class/lab so you will understand I/O and other issues. I am also uploading sample inputs in addition to the CFTR reference sequences.

This assignment is due by MIDNIGHT of Tuesday, September 3. Late submissions carry a -33 % per day penalty.

CS 490 Students: The assignment is worth 6% of your total grade.

CS 590 Students: The assignment is worth 4% of your total grade.