Final Exam Instructions

Please thoroughly read the instructions below before you take the Final Exam.

Write a Python program that takes as input a file containing DNA sequences in multi-FASTA format, and computes the answers to the following questions. You can choose to write one program with multiple functions to answer these questions, or you can write several programs to address them. We will provide a multi-FASTA file for you, and you will run your program to answer the exam questions.

While developing your program(s), please use the following example file to test your work: dna.example.fasta

You'll be given a different input file to launch the exam itself.

Here are the questions your program needs to answer. The quiz itself contains the specific multiple-choice questions you need to answer for the file you will be provided.

(1) How many records are in the file? A record in a FASTA file is defined as a single-line header, followed by lines of sequence data. The header line is distinguished from the sequence data by a greater-than (">") symbol in the first column. The word following the ">" symbol is the identifier of the sequence, and the rest of the line is an optional description of the entry. There should be no space between the ">" and the first letter of the identifier.

(2) What are the lengths of the sequences in the file? What is the longest sequence and what is the shortest sequence? Is there more than one longest or shortest sequence? What are their identifiers?

(3) In molecular biology, a reading frame is a way of dividing the DNA sequence of nucleotides into a set of consecutive, non-overlapping triplets (or codons). Depending on where we start, there are six possible reading frames: three in the forward (5' to 3') direction and three in the reverse (3' to 5'). For instance, the three possible forward reading frames for the sequence AGGTGACACCGCAAGCCTTATATTAGC are:

AGG TGA CAC CGC AAG CCT TAT ATT AGC

A GGT GAC ACC GCA AGC CTT ATA TTA GC

AG GTG ACA CCG CAA GCC TTA TAT TAG C

These are called reading frames 1, 2, and 3 respectively. An open reading frame (ORF) is the part of a reading frame that has the potential to encode a protein. It starts with a start codon (ATG), and ends with a stop codon (TAA, TAG or TGA). For instance, ATGAAATAG is an ORF of length 9.

Given an input reading frame on the forward strand (1, 2, or 3) your program should be able to identify all ORFs present in each sequence of the FASTA file, and answer the following questions: what is the length of the longest ORF in the file? What is the identifier of the sequence containing the longest ORF? For a given sequence identifier, what is the longest ORF contained in the sequence represented by that identifier? What is the starting position of the longest ORF in the sequence that contains it? The position should indicate the character number in the sequence. For instance, the following ORF in reading frame 1:

>sequence1

ATGCCCTAG

starts at position 1.

Note that because the following sequence:

>sequence2

ATGAAAAAA

does not have any stop codon in reading frame 1, we do not consider it to be an ORF in reading frame 1.

(4) A repeat is a substring of a DNA sequence that occurs in multiple copies (more than one) somewhere in the sequence. Although repeats can occur on both the forward and reverse strands of the DNA sequence, we will only consider repeats on the forward strand here. Also we will allow repeats to overlap themselves. For example, the sequence ACACA contains two copies of the sequence ACA - once at position 1 (index 0 in Python), and once at position 3. Given a length n, your program should be able to identify all repeats of length n in all sequences in the FASTA file. Your program should also determine how many times each repeat occurs in the file, and which is the most frequent repeat of a given length.