Homework 2
CS 181, Fall 2019

Out: Sep. 28
Due: Oct. 5, 5:00 PM

Please print your answers and place them in the CS1810 homework bin on the 2nd floor of the CIT. You can use \LaTeX or a word document to write up your answers, but we prefer you use \LaTeX. You may scan hand-written work or images for parts of solutions only if they are extremely clean and legible. Please put your Banner ID on the top of each page of your homework. Please ensure that your name does not appear anywhere on the paper you hand in.

Problem 1: Motifs

Up to this point, we have mostly represented DNA strands as single strings of nucleotides, such as AATCGAGG. However, DNA molecules actually consist of two strands of complementary nucleotides, as shown below:

\begin{align*}
\text{AATCGAGG} \\
\text{TTAGCTCC}
\end{align*}

Furthermore, each strand of DNA in a DNA molecule has a polarity that determines the direction in which it can bind to other DNA strands. We label the two ends of a DNA strand with 5’ and 3’ to denote the direction in which the DNA strand is written. When two complementary DNA strands bind to each other to form a DNA molecule, the two strands must have opposite polarities:

\begin{align*}
5’ \text{AATCGAGG} & 3’ \\
3’ \text{TTAGCTCC} & 5’
\end{align*}

By convention, we typically assume that DNA strands are written in the 5’ to 3’ direction unless their polarities are explicitly given. However, when working with complete DNA molecules, it is often important to specify the direction of each strand, as the polarities of the DNA strands can affect biological behaviors. In this problem, we will explore how motif identification is affected by the polarities of DNA strands.

In class you will soon be learning about motifs. Wikipedia defines a motif as “a nucleotide or amino-acid sequence pattern that is widespread and has, or is conjectured to have, biological significance”. Combinatorial pattern matching algorithms are often used to find motifs, but in this homework we’ll see how alignment algorithms can be used for the same task.

For this problem, we’ll be exploring a very important motif, the restriction site. Restriction sites are palindromic sequences (sequences which are identical to their reverse complements, e.g. CCCGGG) that serve as recognition sites for restriction enzymes. Restriction enzymes are proteins that look for these restriction sites and cut them like a pair of molecular scissors, breaking the DNA into fragments. Because the restriction sites are palindromic, the enzymes bind onto both sides of the DNA and cleave the strands...
at a specific position on the site. For example, the restriction enzyme *TaqI* recognizes the palindromic restriction site *TCGA*, and cleaves it between the *T* and *C* position. Thus, we will say that *TaqI* has a recognition site of *T/CGA*, to show the restriction site and where the site is cleaved. Here is an example of how a DNA strand might be cleaved using *TaqI*:

```
5’ AATCGAGG 3’
3’ TTAGCTCC 5’
↓
5’ AAT CGAGG 3’
3’ TTAGC TCC 5’
```

Importantly, *TaqI* can only cleave the restriction sequence *TCGA* if the sequence appears in the DNA molecule from 5’ to 3’. Consequently, if the polarities of the strands in the above example were switched, *TaqI* would no longer be able to cleave the DNA molecule:

```
3’ AATCGAGG 5’
5’ TTAGCTCC 3’
```

Task: For the following sequence, list all the DNA fragments that would be produced if it was exposed to the restriction enzyme *EcoRI*, which has the recognition site *G/AATTC*. How would reversing the polarity of the following sequence change the DNA fragments produced after exposure to *EcoRI*?

```
5’ AATGGCTTAAGTTGGAATTTCGAGAATTCGGGAATTCATCTTAAGGGGAATTGGGAATTCAA 3’
```

Sometimes *dot plots* are used to better visualize how two sequences are aligned. Dot plots make use of a table that is similar to an alignment table. We use the characters of two different strings to label the rows and columns of the table. Each cell is then filled with a dot if its row and column correspond to a matching character, and left empty otherwise. Here is an example:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>T</th>
<th>C</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
</tbody>
</table>

Task: Create a dot plot for the long sequence above and the recognition site of *EcoRI*. Because we are very kind, we have provided you with the table so you can just print it and fill it out. Alternatively, you can obtain the \LaTeX source by running `cs181_setup hw2`.

```
AAATTGGCTTTAAGTTGGAATTTCGAGAATTCGGAATTTCGATCTCATTTAAGGGGAATTGGGAATTTCGAA
```
What do you notice about your dot plot? Task: Describe the relationship between the dot plot and the DNA fragments you previously identified. Specifically, explain how the dot plot can be used to identify restriction sites for both possible polarities of the DNA sequence given.

Note: Restriction enzymes are the basis for the polymerase chain reaction (PCR), which is used to amplify DNA samples and earned its inventor Kary Mullis the Nobel Prize in 1993. It is no exaggeration to say that most of modern DNA biotechnology would not be possible without PCR. Indeed, there is a long tradition in biology of using tools and mechanisms devised by nature to suit the needs of science and society. A more recent example of this is the CRISPR/Cas system, which is a prokaryotic immune mechanism that is now being used for precise genome editing.

Problem 2: Free End Gap Alignment

Now that we have an intuition for what motifs might look like in a sequence, let’s design an alignment algorithm to find them for us. Global alignment won’t work for the motif problem because it will align our short motif to a large region of the longer string with many gaps. Since we expect our motif to match on a small area, we might guess that local alignment is appropriate. But local alignment will also have trouble because we want to align the entirety of our motif, and local alignment may pick out a subsequence of the motif. To sidestep these problems, we introduce a new kind of alignment: Free End Gap Alignment. We want to align the entire motif to a subsequence of the longer sequence, with as few gaps in the aligned section as possible. To this end, we will discount any gaps that occur on the ends of a sequence. For example, the optimal global alignment of GTAGGCTTAAGGTTA and TAGATA is

```
GTAGGCTTAAGGTTA
-TAG----A----T-A
```

with a score of $-3$ (assuming our usual scoring scheme of $+1$ for match, $-1$ for gap and mismatch). An optimal Free End Gap Alignment, however, would be

```
GTAGGCTTAAGGTTA
-TAGA---TA------
```

with a higher score of $2$. Indeed, this latter alignment better captures the essence of what it means to find a motif in a longer sequence. We formalize the Free End Gap Alignment problem as follows:

<table>
<thead>
<tr>
<th>FREE END GAP ALIGNMENT</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> Strings $u$, $v$, and a scoring function $\delta$ that takes in two characters as arguments.</td>
</tr>
<tr>
<td><strong>Output:</strong> An alignment of $u$, $v$, for which the score is maximal (as defined by $\delta$) and for which end gaps are free (i.e. not penalized).</td>
</tr>
</tbody>
</table>
Task: Design a dynamic programming solution to the Free End Gap Alignment problem. Your solution must include

- a new recurrence relation, if necessary
- a description of how your table is initialized
- how to backtrack through your table, including where to start and where to end

If you wish, you may also include diagrams, pseudocode, mathematical expressions, and plain English text in your answer. *Hint: It might help to think about how to appropriately combine the ideas from global and local alignment.*

Now to see the algorithm at work. We’ve implemented Free End Gap Alignment for you already. Run `cs181_setup hw2` to get the code. The implementation takes in two arguments, a long sequence and a short sequence to align it against (the scoring scheme is the standard scheme mentioned prior). The code will then output all optimal alignments. Here’s an example of how to run our implementation:

```
> sh FEGAlignment.sh GTAGGCTTAAGGTTA TAGATA
GTAGGCTTAAGGTTA
-TAGA--TA------
```

Task: Use our implementation to count the number of DNA fragments that are created by exposing the following sequence (all three lines make up one contiguous sequence, written 5’ to 3’) to the restriction enzyme *BamHI*, which has the recognition site *G/GATCC*. How would your answer change if the polarity of the following sequence was reversed?

```
AAGGAGTCGATGCAATGGGACCGATCCGAAATTTGGCCTAGGGGTAAAGCCTAGGGCTAGGGTGAAATTGGCCTAGGGGTAAGGCCTAGGGG
GGAGAGAGTCGATGAGGATCCAGGGTCATCCCCCTAGGGGGATATATATGGGGATTCGATGATGGACTT
ATTACCCTAGGATAGGGCGATTGACGTAGGGATCCCGATGGGCCTAGGTATATAGGGTATACCCGATCCC
```

**Problem 3: Homology**

Finding conserved patterns across different species is important for evolutionary biology. Consider the following sequences:

- **A**: TACTAC CTTTCACCAAT AAGTCAGGATGAAAACGA
- **B**: TGCTAG TTGGACACTTAG AAATCAGGGGGA GAGCAA
- **C**: GGGAAA CTACTGCAG GAGACGATCGAA AACAAAAGG

We wish to find whether B or C is a homologous protein to A, that is, a protein that originates from some ancestral species. The optimal alignment for A and B is

- **A**: TACTACCTTTCACCAATA-AG-TCAGGGATGAAAACGA
- **B**: TGCTAG-TTGG-AC-ATTAGAAATCAGGGGGAGAGCAA
with a score of 6. The optimal alignment between A and C is

A: T--ACTACCTTCTCACCATAAGTCAGGAT--GAAAACGA----
C: GGGAA--AC--TACTGC--AGGA--GAC--G--ATCGAAAACAAAGG

with a score of -1. After splicing the DNA into cells, you find that the DNA actually transcribes the following protein sequences (using the one-letter amino acid code)

A: LLTNKR
B: LDTYEQ
C: LLQNKR

Notice how similar A and C are as proteins despite being much more dissimilar than A and B as DNA sequences. Task: Look up the terms intron and exon and give a biological reason for this observation. Why might it be a better idea to align by amino acid sequence rather than by DNA?

There are 20 different amino acids and only 4 different nucleotides (used in DNA). Task: Give a probabilistic reason why aligning by amino acid sequence might be better than by DNA sequence.

**Problem 4: A Fowl Virus**

After going to the Brown career fair and talking with a farm recruiter, you’ve decided to become a chicken farmer. Your farm thrives. Your chickens lay eggs at unprecedented rates and you are known around the world as the best farmer that ever was. But then suddenly, your chickens start dying at alarming rates! You realize that there is a viral epidemic ravaging your coops. Luckily for your chickens, you took computational biology and happen to live next to a sequencing center. You manage to isolate some of the virus and get it sequenced. Now, with the sequence in your flash drive, it is up to you to save your chickens.

In the support directory (obtained by running `cs181_setup hw2`) you’ll find the sequence for the terrible virus plaguing your fowl. You know that the virus is a retrovirus and is inserting genes into your poor hens’ DNA. It’s up to you to find out what gene or genes are causing this farmhouse mayhem. Fortunately, you know about BLAST (Basic Local Alignment Search Tool). Navigate to the BLAST website. You’re only interested in your chicken’s genome so type “chickens” into the species search box and click “search.” Now, to find out what gene the virus is infecting your chickens with, copy and paste the viral genome into the box asking for a FASTA sequence and click “BLAST.” It will take a short time (1-2 minutes) before NCBI returns your query. Once it does, investigate the alignments it returns and try to get to the bottom of this plague. Task: figure out what gene your chickens are being infected with and what disease is killing them as a result. (Hint: the “Features” section of each alignment lists genes or proteins coded within the alignment.). Be sure to provide justification for your answers. The virus itself has an interesting history; if you’re interested, try to figure out what virus it is exactly.

**Bonus:** The general structure of a retrovirus genome is composed of coding regions for the gag-pol-env polypeptides (NOT proteins) and other proteins that assemble the polypeptides into proteins. Using this information, find a closely related virus for the virus above. Give a short description of how you found the related protein.
Problem 5: Game Show

You are a contestant on a game show with the following rules. There are three doors, labeled 1, 2, 3. A single prize has been hidden behind one of them. You get to select one door. Initially your chosen door will not be opened. Instead, the game show host will open one of the other two doors at random, doing so in such a way as not to reveal the prize. For example, if you first choose door 1, the host will then open one of doors 2 and 3, and it is guaranteed that the host will choose which one to open so that the prize will not be revealed.

At this point you will be given a fresh choice of door: you can either stick with your first choice, or you can switch to the other closed door. All the doors will then be opened and you will receive whatever is behind your final choice of door.

Suppose you choose door 1 first; then the game show host opens door 3, revealing nothing behind the door, as promised. Task: Determine whether you should stick with door 1, switch to door 2, or whether it makes no difference what you do. Give a detailed justification for your answer and state the probability of winning using either strategy.

Bonus Problem: Earthquake

Imagine that the game happens again and just as the game show host is about to open one of the doors, a violent earthquake rattles the building and one of the three doors flies open at random. It happens to be door 3, and it happens to not have the prize behind it. You had initially chosen door 1.

After recovering from the jolt, the host suggests, “Okay, since you chose door 1 initially, door 3 is a valid door for me to open according to the rules of the game; I’ll let door 3 stay open. Let’s carry on as if nothing happened.”

Task: Should the contestant stick with door 1, switch to door 2, or does it make no difference? What is the difference between this scenario and the no-earthquake scenario, if any? Assume that the prize was placed randomly, that the game show host does not know where it is, and that the door flew open because its latch was broken by the earthquake.

Problem 6: Regex Introduction

The next chapter in CS181 is combinatorial string pattern matching. An understanding of pattern matching is helped by knowledge of regular expressions, or regexes. This is a way for programmers to specify a sequence of characters that define a search pattern. For a guide to regex, see the resources page on the website.
Here, we present problems with 3 strings. The first two strings should be returned by a written regex, while the last one should be ignored. For instance, a valid regex expression for the strings $a, b, c$ would be $[ab]$. All of these regex strings can be written in very compact forms.

a. James Bond, Jack Ryan, Jason Bourne
b. Casino Royale, Spectre, Skyfall
c. CIA, FBI, NSA