C343 Project - DNA Sequence Alignment

1 Assignment Description

Sequence Alignment is an important technique in understanding how *similar* two DNA sequences are. Applications of DNA sequence alignment range from determining gene function to finding similarity between species. Informally, alignment can be understood as writing the two sequences in rows, where the two characters in the same column are said to be aligned. If the two aligned characters are the same, we have a *match*; if the two characters are not the same, we have a *mismatch*. To try and maximize the number of matches, we can also insert gaps in either sequence. Mismatches can be interpreted as point mutations and gaps as insertion or deletion mutations. We disallow columns that consist of gaps only.

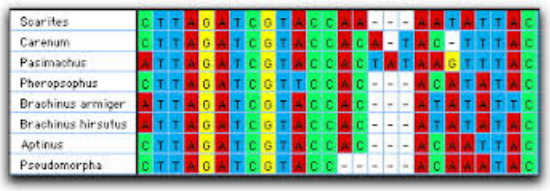

Figure 1: Alignment Example

There can be many alignments of two sequences, but the scores assigned to matches, mismatches, gaps determine which are the best alignments. An optimal solution can be found using a dynamic programming approach. Dynamic programming algorithms can be visualized by a board whose cells keep track of the best solution till that point. In bioinformatics, these are called *dot boards*.

The algorithms fills in the cells of this board with the best scores for every possible prefix of the alignment.

		A	C	A	C	A	C	T	A
	0	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	2	1	0	-1	-2	-3	-4	-5
G	-2	1	0	-1	-2	-3	-4	-5	-6
C	-3	0	3	2	1	0	-1	-2	-3
A	-4	-1	2	5	4	3	2	1	0
C	-5	-2	1	4	7	6	5	4	3
A	-6	-3	0	3	6	9	8	7	6
C	-7	-4	-1	2	5	8	11	10	9
A	-8	-5	-2	1	4	7	10	9	12

Figure 2: Dot Board

2 Your Task

Your task is to complete the `align` function in the `AlignmentFunction` class. This function takes four parameters:

- `s1` - the first sequence.
- `s2` - the second sequence.
- `s1_aligned` - put the result of aligning the fist sequence here.
- `s2_aligned` - put the result of aligning the second sequence here.

`s1_aligned` and `s2_aligned` are of the `StringBuilder` type, which provides an efficient way to build strings. See the documentation of Java for more details.

The return type of `align` is a two-dimentional array of results, that is, objects of the `Result` class. Each of these objects includes a score and a mutation (`I` for insert, `D` for delete, `M` for match or mismatch) that indicates which choice was made. Your `align` function must fill up this two-dimentional array with all the best partial solutions.

Use the function named `Driver.match` in your `align` function to calculate the score of each step. It takes two characters and returns the score.

Suppose that your algorithm returns the sequences s'_1 and s'_2 of length n. The total score is as follows:

$$\sum_{k=0\ldots n-1} s(s'_1[k], s'_2[k])$$

Your algorithm must find the alignment (the s'_1 and s'_2) that maximizes the total score.

You need to setup the recurrence equations for this dynamic programming algorithm, initialize the recurrence, compute the scores using the recurrence equations and finally trace the solution to produce the two aligned sequences. All of this is done in the `align` function.

Here is an example of the completed dot board for aligning $X = CA$ and $Y = AC$. Here the `SPACE_PENALTY` is -1, a match is 2, and a mismatch is -2. To record our choice in each cell, we write D for delete (a gap in Y), I for insert (a gap in X), and M for either match or mismatch.

		j=0	1	2
		Y⇒	A	C
i=0	X⇓	0	I=-1	I=-2
1	C	D=-1	M=-2	M=1
2	A	D=-2	M=1	I=0

Therefore, a best alignment (not unique) for CA and AC is $(CA\_, \_AC)$.

3 Deliverables

Your repo's `seq_align` folder should contain all the files from the zip. These are the ones you need to modify:

- `AlignmentFunction.java` - containing your solution
- `README.md` - where you explain your code

4 Testing

We generated the `tests.txt` file for you to test the correctness of the program. We also encourage you to write your own unit tests.

5 Running Your Code

You can run the program in two modes:

- interactive - invoke with *java Driver gui*
- test - invoke with *java Driver test*. Make sure that the `tests.txt` appears in the current directory.