BCB 567 Fall 2017

Assignment 1: Comparison of DNA Sequences

Due at 11:50 pm on Wednesday, September 20

Requirements

This assignment gives you an opportunity to implement an algorithm for computing an optimal local alignment between two DNA sequences. The algorithm is described in one of the tutorials in the Week 1 folder on Blackboard Learn (Bb). You can write your program in any of the following programming languages: C, C++, Java, Perl and Python. You should include a README file in your submission that describes the components and structure of your program and shows the instructions for compiling and running your program at command line.

Your program should take the following arguments at command line: seq1 seq2 mismatch gap_open gap_extend, where seq1 is a file of one DNA sequence in FASTA format, seq2 is a file of another DNA sequence in FASTA format, mismatch is the base mismatch score (a negative integer), gap_open is the gap open penalty (a non-negative integer), and gap_extend is the gap extension penalty(a positive integer). Each base match score is always 10.

Your program reads the two files of sequences in fasta format and the other parameters, computes the dynamic programming matrices, produces an optimal local alignment, and prints out the alignment. Your program should take time and space in $O(mn)$, where m and n are the lengths of two DNA sequences. Your program should be able to handle sequences of lengths less than $10,000$.

A DNA sequence in FASTA format begins with a single-line description, followed by lines of DNA letters in the sequence. The description line begins with the symbol '>' and continues with the name of the sequence. There is no space between the '>' and the first letter of the name, and no space within the name. The rest of the line is called a description (optional). In other words, if there is a description, then the name and the

description is separated by one or more space. In this project, it is fine to assume that there is no description after the name.

Your program reports, on the stdout, a summary of sequence and alignment information and the alignment. The summary includes the values of the four scoring parameters, the name and length of each sequence, the similarity score of the alignment, the length of the alignment, the percent identity, the number of matches, the number of mismatches, the total length of gaps. The alignment is reported in sections of 70 characters, with each section consisting of three rows. The sequence positions of the first DNA letters in each section are reported in the left margin of 10 spaces.

Submission

You are required to include, in your submission, each source code file. You should make your code efficient and non-redundant and include as many checks as possible to catch errors and avoid segmentation faults in execution. Note that the specification is subject to change. Please check on Bb for the latest clarifications.

Be sure to put down your name after the @author tag in each source file. Your zip file should be named Firstname_Lastname_hw1.zip, You may submit a draft version of your code early to see if you have any submission problem with Blackboard Learn. We will grade only your latest submission.