Fall 2015 **CMSC 423: MidTerm 2** H. Corrada Bravo

Time: 1 Hour, 15 Minutes

WAIT FOR INSTRUCTIONS BEFORE BEGINNING

HONOR PLEDGE: "I pledge on my honor that I have not given or received any unauthorized assistance on this examination."

Signature and UID: _____

Print name: _____

- ***Write your answers with enough detail about your approach and concepts used, so that the grader will be able to understand it easily.***
- ***The sum of the grades is 105, but your grades would be out of 100 (thus you get 5 bonus points by solving all the problems).***
- ***Select the best choice for the first 6 problems and mark it by X in the table below.***

Problem	1	2	3	4	5	6
A						
B						
C						
D						
E						

DO NOT WRITE BELOW THIS LINE

Questions 1-6	/ 30	Question 9	/ 20	Total
Question 7	/ 10	Question 10	/ 15	
Question 8	/ 10	Question 11	/ 20	

Multiple-choice Problems (Answer THE BEST CHOICE in the Table of the First Page and NOT HERE):

1. **(3 points**) The BLOSUM62 scoring matrix for pairwise alignment is an example of a log-odds based scoring approach. Which of the following statements is most accurate for this type of scoring approach:

   (a) It is strictly based on the number of matching characters in an alignment.

   (b) It is based on probabilistic models of sequences that could arise through an evolutionary process learned over a training set of closely related sequences.

   (c) It is based on the log-ratio of probabilities computed from two possible explanatory probabilistic models of sequence evolution

   (d) All of the above

   (e) Only (b) and (c)

2. **(10 points**) Consider the recurrence relations for global alignment with affine gap penalties below. There are a number of mistakes in the indices used to define some of the recurrences, and/or in the terms included in some of the terms included in some of the recurrences. How many mistakes are there overall? Circle, explain and correct each mistake. Assume that the cost of a gap of length *k* is $gap(k) = \sigma + (k-1)\epsilon$.

$$X[i,j] = \max \begin{cases} X[i,j-1] - \epsilon \\ M[i,j-1] - \sigma \\ Y[i,j-1] - \sigma \end{cases}$$

$$M[i,j] = \max \begin{cases} X[i,j] \\ M[i,j] + \mathrm{SCORE}(x[i], y[j]) \\ Y[i,j] \end{cases}$$

$$Y[i,j] = \max \begin{cases} Y[i-1,j] - \epsilon \\ M[i-1,j] - \sigma \\ X[i-1,j] - \sigma \end{cases}$$

(a) None (b) One (c) Two (d) Four (e) Nine

3. **(2 points**) Given a directed graph G=(V,E), the Hamiltonian Path problem is to:

   a) Find a path that visits all edges in E exactly once

   b) Find the path that visits the most nodes in V, while visiting every edge in E exactly once

   c) Find a path that visits all nodes in V exactly once

   d) Find the shortest path between every pair of nodes in V

   e) None of the above

4. **(10 points**) Consider the multiple sequence alignment problem for 3 sequences *v*, *w*, and *u* of length *n,* and a proposed recurrence relation to compute a global alignment shown below. What would be the time and space complexity of a dynamic programming solution to this problem. Explain.

$$s_{i,j,k} = \max \begin{cases} s_{i-1,j,k} & + & \text{SCORE}(v_i, -, -) \\ s_{i,j-1,k} & + & \text{SCORE}(-, w_j, -) \\ s_{i,j,k-1} & + & \text{SCORE}(-, -, u_k) \\ s_{i-1,j-1,k} & + & \text{SCORE}(v_i, w_j, -) \\ s_{i-1,j,k-1} & + & \text{SCORE}(v_i, -, u_k) \\ s_{i,j-1,k-1} & + & \text{SCORE}(-, w_j, u_k) \\ s_{i-1,j-1,k-1} & + & \text{SCORE}(v_i, w_j, u_k) \end{cases}$$

   (a) $O(n^2)$

   (b) $O(n^3)$

   (c) $O(n^3 2^3)$

   (d) $O(2^3)$

   (e) $O(3^n)$

5. **(2 points**) Which of these statements are accurate for genome assembly

   (a) The Hamiltonian approach is problematic due to the computational complexity of the Hamiltonian path problem

   (b) The time complexity of constructing a read overlap graph is the same as the time complexity of constructing a DeBruijn graph

   (c) The Eulerian approach is problematic due to the large number of Eulerian paths in a DeBruijn graph

   (d) All of the above

   (e) Only (a) and (c)

6. **(3 points**) Which of these are reasons to use inexact string matching methods to compare biological sequences:

   (a) Exact matching misses string overlaps required for genome assembly assuming sequencing errors

   (b) Exact matching would not sensitively identify protein sequences from different species with potentially the same molecular function

   (c) Genomic variants in sequences from an individual may not match any position of a reference genome when using exact matching

   (d) Only a) and c)

   (e) All of a), b) and c)

Questions (show all derivations as appropriate for full credit):

Problem 7. (10 points) (a) Define the concept of “coverage” as used in genome assembly. (b) The Lander-Waterman statistic provides a mathematical model of the relationship between “coverage” and the number of contiguous pieces (contigs/islands) of sequence that can be assembled from a given genome. Describe roughly the relationship between the two (a sketch illustrating the function given by the Lander-Waterman statistic is sufficient). Mention the rate (linearly, quadratically, exponentially, etc.) at which the expected number of contigs changes as coverage increases.

Problem 8. (10 points) You are a clinical genomicist and have sequenced a patient's genome. To find possible disease causing mutations you are going to compare the millions of reads generated by the sequencing instrument to a reference human genome: you want to find the placement of sequence queries (reads) of length 200 along the human genome (3 Gbp), allowing for at most 4 mis-matches in a 200 bp read.

However, using a dynamic programming solution to the fitting alignment problem is not efficient. Instead you will use a much more efficient exact matching algorithm to find candidate positions in the genome. (Recall that the exact matching problem: given *query* and *target* strings, find all **exact** occurrences of *query* in *target*).

The proposed algorithm is: divide each 200 bp query read into *non-overlapping* k-mers and find all exact occurrences of each k-mer in the reference genome. Note that the larger k is, the more efficient this procedure since the number of string comparisons is smaller, but k must be less than 200 since you would miss occurrences with at most 4 mis-matches. What is the maximum value of k you can use that guarantees that no occurrences with at most 4 mis-matches of the query read are missed? Explain.

Problem 9. (20 points) Given a set of reads from a sequencing run of a human tissue sample, we might want to align each read to the human genome. Since the former are roughly 200bp long, and the latter is roughly 3Bbp long, global alignment will not be appropriate. Instead, we would use a *fitting* alignment, in which every character in the 200bp read must be aligned, but gaps added before the first character in the read and gaps added after the last character in the read are unpenalized. Describe how you would modify the global alignment dynamic programming algorithm to compute fitting alignments. Please specifically address the following points:

(a) How do you define a fitting alignment (drawing indicating which string is the 200bp read and which is the reference genome is sufficient)?

(b) What are the initial conditions in the DP table?

(c) In which cell of the DP table will the score of the optimal fitting alignment be found (i.e., where will you start backtracking)?

(d) What recurrence relations will you use: global alignment or local alignment (adding 0 to “start over” an alignment)? Write out the recurrence relation?

(e) Where will you stop backtracking to construct the fitting alignment?

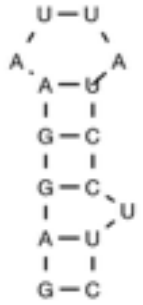
Note: You can assume linear (not affine) gap penalties.

Problem 10. (15 points). Write down an expression for the probability that the length *k* prefix of a randomly generated DNA string of length *n* is equal to the reverse complement of its length *k* suffix. E.g., ACGTATTAACGT is one such string for n=12 and k=4.

(a) Solve assuming $2k \leq n$

(b) Solve assuming $k \leq n < 2\mathrm{k}$

Problem 11 (20 points) The secondary structure of RNA molecules, given by intra-molecular complementary base pairings, is commonly referred to as 'hairpin' or 'stem and loop' structures based on two-dimensional pictorial representation. For example, the secondary structure of RNA sequence `x=GAGGAAUUAUCCUUC` is given by base-pairings of non-consecutive bases (e.g., x_1-x_{15}, x_2-x_{14}, etc., where x_{15} is the C occurring in the 15th position of the sequence). Note that not all bases are paired (for example x_{13} is unpaired):



The RNA secondary structure prediction problem is to determine the secondary structure of an RNA molecule, given its nucleotide sequence. One solution for this problem is given by finding the structure that maximizes the number of complementary base-pairings between the prefix and suffix in the RNA sequence using dynamic programming (for example the number of base-pairings in the above example is 5). Let M(i,j) be the maximum number of base-pairings for the subsequence starting at position i and ending at position j. Then the solution to the RNA prediction problem would be given by M(1,n) where n is the length of the RNA sequence.

(a) Complete this recurrence relation below for M(i,j). Explain how you derived it.

$$M(i,j) = \max \begin{cases} M(\quad,\quad) + (1 \text{ if bases } x_i \text{ and } x_j \text{ are complementary, } 0 \text{ o.w.}) \\ M(i,\quad) \\ M(\quad,j) \end{cases}$$

(b) Explain what would be initial conditions for the reccurence in these cases:

- M(i,i)

- M(i,j) for i<j

- M(i,j) for i > j

(c) Draw and fill-in a dynamic programming table/graph to solve this problem for RNA sequence GAUUC.

(d) What is the maximum number of base-pairings for the sequence in part (c)?

(e) What is the time complexity of a DP solution to this problem?