|  |  |  |
| --- | --- | --- |
| Fall 2014 | **CMSC 423: Final** | H. Corrada Bravo |

Time: 2 Hours

**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 8 problems and mark it by X in the table below.***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Problem** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **A** | **X** | **X** |  |  |  |  |  |  |
| **B** |  |  |  |  |  |  | **X** |  |
| **C** |  |  | **X** | **X** |  |  |  | **X** |
| **D** |  |  |  |  | **X** | **X** |  |  |
| **E** |  |  |  |  |  |  |  |  |

DO NOT WRITE BELOW THIS LINE

|  |  |  |  |
| --- | --- | --- | --- |
| Problems 1-8: | /30 | Problem 12: | /12 |
| Problem 9: | /8 | Problem 13: | /15 |
| Problem 10: | /8 | Problem 14: | /20 |
| Problem 11: | /12 | **Total:** |  |

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

1. **(2 points**) Refer to the RNA translation code in Problem 9. Which of the following is a *non-synonymous* mutation?

a) AGA->AGC b) GGT->GGA c) CGG->AGG

d) (a) and (b) e) none of the above

1. **(3 points)** Given a directed graph G=(V,E), the Eulerian Path problem is:

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

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

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

d) Find the shortest path between a pair of specific nodes u and v in V

e) None of the above

1. **(6 points)** Consider the Sequence Profile (Position Specific Scoring Matrix, or PSSM) for some DNA motif shown below. In which position of string ATTCAGGA is the highest probability substring of length 4 found? (Note: assume 1-indexing so the first character is in position 1. Show your work.)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Pos 1 | Pos 2 | Pos 3 | Pos 4 |
| A | .8 | .6 | .4 | .6 |
| C | .2 | .3 | .5 | .1 |
| G | 0 | .3 | .1 | .1 |
| T | 0 | 0 | 0 | .2 |

a) 1 b) 3 c) 4 d) 5 e) 6

1. **(2 points**) Which of the following resources contain high-throughput sequencing data:

a) KEGG Pathway Database b) Pubmed

c) Short Read Archive d) (b) and (c) e) all of the above

1. **(3 points)** Which of the following statements are true of the Burrows-Wheeler transform BWT(S) of string S

a) there is an algorithm using BWT(S) with time complexity O(|P|) to determine if string P occurs in S

b) the *i*th occurrence of character *c* in BWT(S) corresponds to the *i*th occurrence of c in S

c) BWT(S) is sufficient to reconstruct the first column of the rotation matrix of S

d) (a) and (c) e) None of the above

1. **(6 points**) I used k-means clustering with k=2 and obtained the two cluster centers given below (Table 1). What would be the cluster assignments for the three samples given below (Table 2)? Show your work (a drawing is sufficient).

Table . Centers

|  |  |  |
| --- | --- | --- |
|  | Cluster 1 | Cluster 2 |
| Gene 1 | 1 | -1 |
| Gene 2 | 0 | 0 |
| Gene 3 | 1 | -1 |

Table . Samples

|  |  |  |  |
| --- | --- | --- | --- |
|  | Samp. A | Samp. B | Samp. C |
| Gene 1 | 2 | -1 | 2 |
| Gene 2 | 0 | 0 | 0 |
| Gene 3 | 2 | 0 | -1 |

a) A: 1, B: 1, C: 2

b) A: 2, B: 2, C: 1

c) A: 1, B: 2, C: 2

d) A: 1, B: 2, C: 1

e) A: 2, B: 1, C: 2

1. **(4 points)**  Which of the following statements best represents complexity of using suffix arrays for exact string matching of query string P to target string T

a) preprocessing: O(|P|), space: O(|P|), search: O(|T|)

b) preprocessing: O(|T|), space: O(|T|), search: O(|P| \* log |T|)

c) preprocessing: O(|T| \* log |T|), space O(|T|), search O(|P| \* log |T|)

d) preprocessing: O(|T| + |P|), space O(|T|), search O(|P| \* |T|)

e) None of the above

1. **(4 points**) The cyclo-spectrum of peptide W-S-P-I contains the following masses. Show your work

a) 283 b) 71 c) 299 d) (a) and (c) e) None of the above



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

1. **(8 points)** Identify the longest open reading frame in the following DNA sequence and translate it into an aminoacid sequence (start codon is ATG):

TGCGTATGTATGTCAGACGGTGAGACGCTTGCGGGCTAAGCGACG

****

TGCGTATGTATG TCA GAC GGT GAG ACG CTT GCG GGC TAAGCGACG

Met Ser Asp Gly Glu Thr Leu Ala Gly

1. **(8 points)** Given a specific DNA sequence *S* of length *n,* how many DNA sequences of length *n* would match *S* with at most *d* mismatches?

Given a vector of d distinct positions, there are 4^d possible strings I can generate from S that would match with at most d mismatched positions. There are n choose d possible vectors. So choose(n,d) \* 4^d

**Problem 11 (12 points)** We showed in class that the worst-case space complexity of suffix *tries* is , where *n* is the size of the string. Discuss the two tricks used to turn a suffix trie into a suffix tree so that only linear space is used in the worst case. Sketch a proof that only linear space is required.

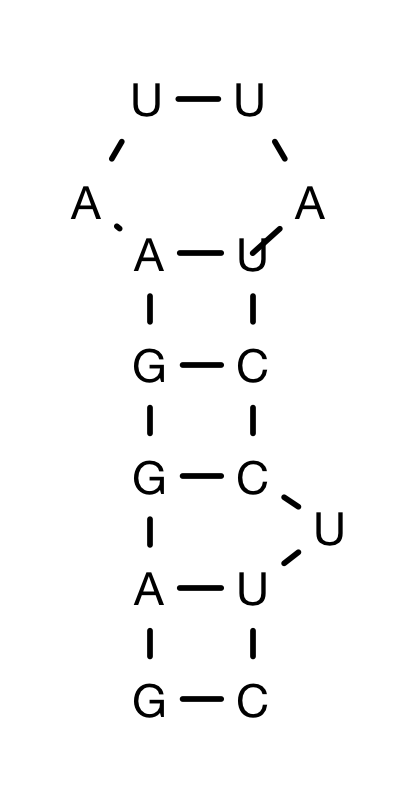
Collapse all paths where no nodes split and label resulting edge with start and stop indices of substring. Each edge label takes O(1) space. Since every node now has at least a binary split, #nodes .

1. **(12 points) Prefix-Suffix matching.** Give an algorithm that takes two strings and , and uses a suffix tree to find the longest suffix of that matches a prefix of . The algorithm should run in time O(|| + ||).

Construct string and build the suffix tree for S. Follow the path given by beta and find the deepest node that has outgoing edge labeled with $2.

***Long Questions (you should always PROVE THE CORRECTNESS of your solutions)***

1. **(15 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 GAGGAAUUAUCCUUC is given by base-pairings of non-consecutive bases (e.g., G1-C15, A2-U14, etc., where C15 is the C occurring in the 15th position of the sequence). Note that not all bases are paired (for example U13 is unpaired):



The *RNA secondary structure prediction* problem is to determine the secondary structure of an RNA molecule, given it’s nucleotide sequence. One solution for this problem is given by finding the structure that maximizes the number of base-pairings 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 would be given by *M(1,n)* where *n* is the length of the RNA nucleotide sequence.

1. Write a recurrence relation for *M(i,j).* Explain *how* you derived it.
2. Draw a dynamic programming table/graph to solve this problem for sequence GAUUC
3. What is the maximum number of base-pairings for the sequence in part (b)



2

**Problem 14 (20 points)** We’ve seen in class two algorithms that use probability estimates as part of an optimization problem: (a) in the Gibbs sampling algorithm for motif finding, we used the ‘profile probability’ of a k-mer to sample positions in DNA sequences containing a protein binding site, and (b) in the EM algorithm used in fuzzy k-means, we used ‘assignment probability’ to calculate cluster centers using weighted averages. Design an EM algorithm to solve the motif finding problem.

1. In fuzzy k-means, the parameters of interest were the k centers. What is the estimate of interest in motif finding?
2. In fuzzy k-means, *HiddenMatrix* was a matrix with a row for each point (e.g., expression from one gene across multiple timepoints) and a column for each center. What does *HiddenMatrixij* correspond to in fuzzy k-means?
3. What should the dimensions of *HiddenMatrix* be for your motif finding EM algorithm? What does *HiddenMatrixij* correspond to in this case?
4. How did you compute *HiddenMatrixij* in fuzzy k-means? Write the mathematical expression. How would you compute *HiddenMatrixij* for motif finding? Write a mathematical expression. Note: this is the *E-step* in your algorithm.
5. How was *HiddenMatrix* used to calculate centers in fuzzy k-means? Write the mathematical expression to calculate the *jth* cluster center. Note: this is the *M-step* in fuzzy k-means.
6. Given *HiddenMatrix,* how would you use it to calculate a motif profile? Write a mathematical expression for entry *pcl* corresponding to nucleotide *c* and position *l* of the profile. Note: this is the *M-step* of your algorithm.

1. the profile, 2. The probability point  *i* belongs to cluster *j*, 3. Num sequences by num. starting positions, the probability that motif starts in position *j* of string *i,*

4.

For motif finding we use profile probability of a given k-mer and normalize to make it a probability. *Sij* is the k-mer at position *j* in the *i*th string in dataset, *p* is current profile:

5.

6. For motif finding we count the number of times a nucleotide occurs at a given position of the motif, now we do ‘weighted counts’

where is 1 if the (j+l-1)’th character of the *i-*th string in the dataset is *c* and 0 otherwise.