CS466: Introduction to Bioinformatics Name: Payal Mantri(payaljm2)

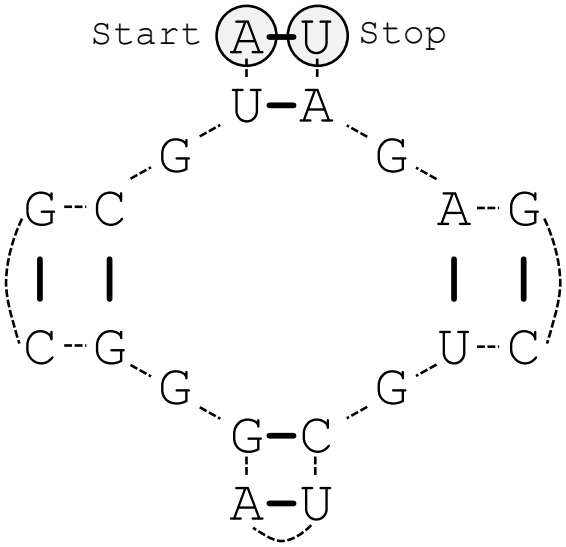
Problem Set 3

Handed out: October 26, 2022 *Due: November 3, 2022*

Instructions: This homework assignment consists of three questions worth a total of 50 points. In addition, Q2d and Q2e are worth an additional 10 bonus points. **Do not forget to write your name at the top!**

1. **RNA Secondary Structure** [15 points]

   Consider the following RNA secondary structure.



   This is the optimal pseudoknot-free secondary structure for the RNA sequence $\mathbf{v}$ = AUGCGCGGGAUCGUCGAGAU, where only base pairings in

$$\Gamma = \{(A, U), (U, A), (G, C), (C, G)\}$$

   receive a score of 1 (and the other base pairings have a score of 0). We will represent this solution in two different ways.

   a. Use the dot-parenthesis format represent the structure. [5 points]

      Hint: Number the bases from 1 to 20 in the figure.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A	U	G	C	G	C	G	G	G	A	U	C	G	U	C	G	A	G	A	U
(	(	-	(	(	)	)	-	(	(	)	)	-	(	(	)	)	-	)	)

b. Recall the Nussinov algorithm, which was based on the following recurrence, where $s[i, j]$ indicates the maximum number of complementary base pairings in the sequence $v_i, \ldots, v_j$.

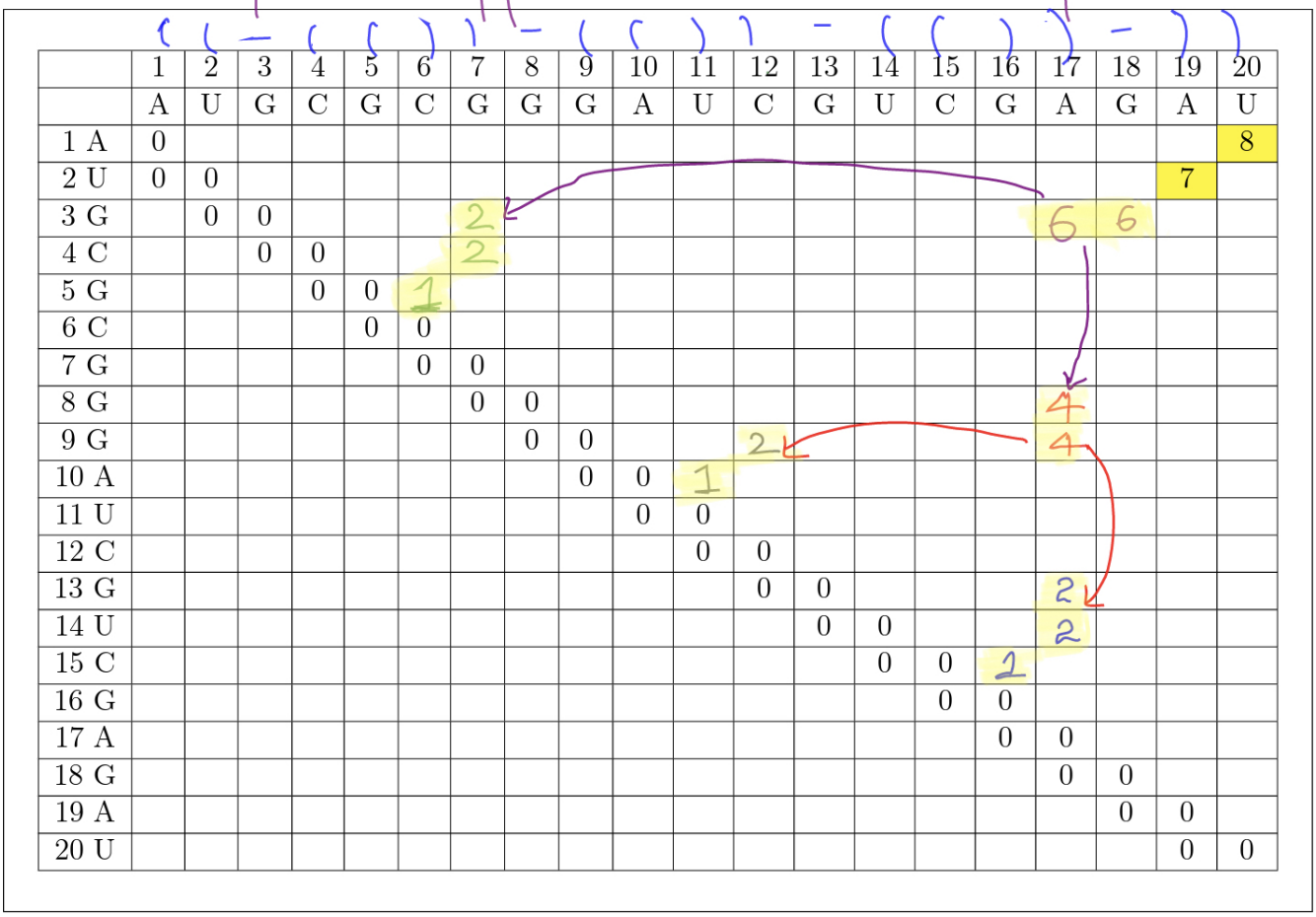
$$s[i,j] = \max \begin{cases} 0, & \text{if } i \geq j, \\ s[i+1, j-1] + 1, & \text{if } i < j \text{ and } (v_i, v_j) \in \Gamma, \\ s[i+1, j-1], & \text{if } i < j \text{ and } (v_i, v_j) \notin \Gamma, \\ s[i+1, j], & \text{if } i < j, \\ s[i, j-1], & \text{if } i < j, \\ \max_{i<k<j}\{s[i,k] + s[k+1, j]\}, & \text{if } i < j. \end{cases} \tag{1}$$

Represent the given optimal structure by performing a backtrace from cell $(1, n) = (1, 20)$ in the following table. **Only fill out the cells that correspond to the given structure.** [10 points]

Hint: Cell $(1, 20)$ has score $s[1, 20] = 8$, corresponding to the eight complementary base pairings. Since v_1 and v_{20} are paired up in the given structure, we move from $(1, 20)$ to $(2, 19)$. The latter cell has a score of $s[2, 19] = 7$ complementary base pairings. These two cells have already been filled out in the table.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	A	U	G	C	G	C	G	G	G	A	U	C	G	U	C	G	A	G	A	U
1 A	0																			8
2 U	0	0																	7	
3 G		0	0				2										6	6		
4 C			0	0			2													
5 G				0	0	1														
6 C					0	0														
7 G						0	0													
8 G							0	0									4			
9 G								0	0			2					4			
10 A									0	0	1									
11 U										0	0									
12 C											0	0								
13 G												0	0				2			
14 U													0	0			2			
15 C														0	0	1				
16 G															0	0				
17 A																0	0			
18 G																	0	0		
19 A																		0	0	
20 U																			0	0

pairings. These two cells have already been filled out in the table.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	A	U	G	C	G	C	G	G	G	A	U	C	G	U	C	G	A	G	A	U
1 A	0																			8
2 U	0	0																	7	
3 G		0	0				2										6	6		
4 C			0	0			2													
5 G				0	0	1														
6 C					0	0														
7 G						0	0													
8 G							0	0									4			
9 G								0	0			2					4			
10 A									0	0	1									
11 U										0	0									
12 C											0	0								
13 G												0	0				2			
14 U													0	0			2			
15 C														0	0	1				
16 G															0	0				
17 A																0	0			
18 G																	0	0		
19 A																		0	0	
20 U																			0	0

2. **Additive Phylogeny** [20 points]

(a) Suppose you are working with a biological collaborator and they have obtained the sequences of the same gene from 5 different, but closely related, species. They have done some initial analysis to compute pairwise distances between between the different species (maybe using something as simple as edit distance, or perhaps something more sophisticated). These distances are shown in the table below.

	a	b	c	d	e
a	0	8	15	9	13
b	8	0	5	10	11
c	15	5	0	11	17
d	9	10	11	0	3
e	13	11	17	3	0

Is the above distance matrix additive? Justify your answer. [5 points]

As per Four point Condition theorem
Every four leaves (quartet) can be labeled as (i,j, k, l) such that
$d_{i,j} + d_{k,l} \leq d_{i,k} + d_{j,l} = d_{i,l} + d_{j,k}$
Let D be an n $\times$ n distance matrix. Matrix D is additive if and only if the four point condition holds for every quartet $i, j, k, l \in n^4$

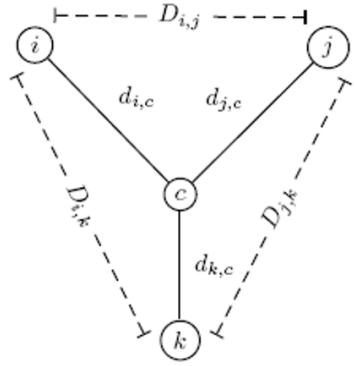
We prove that given matrix D is **not additive** by using contradiction.

Let us consider quartet a,b,c,d
$d_{a,b} + d_{c,d}$ = 8+11= 19
$d_{a,c} + d_{b,d}$ = 15+11 = 26
$d_{a,d} + d_{b,c}$ = 9+5 = 14

For this quartet, the four point condition doesn't hold true , as there are no two sums whose value is equal to each other and greater than the third sum.

Since the condition doesn't hold true for atleast one of of the quartet, by method of contradiction, we can say that **D is not additive**.

(b) In class we considered the large additive distance phylogeny problem with $n = 3$ sequences.



We derived $d_{i,c} = \frac{D_{i,j} - D_{j,k} + D_{i,k}}{2}$. **Derive** $d_{j,c}$ **and** $d_{k,c}$ in terms of $D_{i,j}$, $D_{i,k}$ and $D_{j,k}$. Provide all intermediate steps. [5 points]

We have

$$D_{i,j} = d_{i,c} + d_{j,c} \tag{2}$$
$$D_{i,k} = d_{i,c} + d_{k,c} \tag{3}$$
$$D_{j,k} = d_{j,c} + d_{k,c} \tag{4}$$

i. To derive $d_{j,c}$

Adding equation 2 and 4

$$\begin{aligned} D_{i,j} + D_{j,k} &= d_{i,c} + d_{j,c} + d_{j,c} + d_{k,c} \\ D_{i,j} + D_{j,k} &= 2d_{j,c} + (d_{i,c} + d_{k,c}) \\ D_{i,j} + D_{j,k} &= 2d_{j,c} + D_{i,k} \qquad \text{Using (3)} \end{aligned}$$

rearranging the terms above , we have

$$d_{j,c} = \frac{D_{i,j} + D_{j,k} - D_{i,k}}{2}$$

ii. To derive $d_{k,c}$

Adding equation 3 and 4

$$\begin{aligned} D_{i,k} + D_{j,k} &= d_{i,c} + d_{k,c} + d_{j,c} + d_{k,c} \\ D_{i,k} + D_{j,k} &= 2d_{k,c} + (d_{i,c} + d_{j,c}) \\ D_{i,k} + D_{j,k} &= 2d_{k,c} + D_{i,j} \qquad \text{Using (2)} \end{aligned}$$

rearranging the terms above , we have

$$d_{k,c} = \frac{D_{i,k} + D_{j,k} - D_{i,j}}{2}$$

(c) How do you calculate the trimming parameter δ when constructing an additive phylogeny? Explain using words and mathematical expressions, or provide pseudocode. As usual, pseudocode should contain initializations of variables and an explanation of what they represent. [10 points]

Hint: A **hanging edge** of the triple (A, B, C) of leaves is the edge (B, P) connecting leaf B and the first common vertex P on the paths from A to C and from B to C. The trimming parameter δ equals the length of the **shortest** hanging edge.

Given n × n distance matrix D = $[D_{i,j}]$, where $D_{i,j}$ represents distance between nodes i and j
For every three leaf nodes triple $(i, j, k) \in D$,
assume c as center point and calculate distance from each of node to c using following equations

$$d_{i,c} = \frac{D_{i,j} - D_{j,k} + D_{i,k}}{2}$$

$$d_{j,c} = \frac{D_{i,j} + D_{j,k} - D_{i,k}}{2}$$

$$d_{k,c} = \frac{D_{i,k} + D_{j,k} - D_{i,j}}{2}$$

Now find minimum value $min_{i,j,k} = min\{d_{i,c}, d_{j,c}, d_{k,c}\}$
This $min_{i,j,k}$ gives minimum value for triplet (i,j,k).
Calculate a minimum value from all the minimum of all possible triplets in D to get delta.

It can be represented using following way

$$\delta = \min_{0 \le i < j < k < n} \{\min(d_{(i,c)}, d_{(j,c)}, d_{(k,c)})\}$$

For a n*n matrix , There $\binom{n}{3}$ combinations of i,j,k . And we need to calculate 3 sums for each of these triple .
And to get minimum of all minimums ,we will run through array of size $\binom{n}{3}$
Hence total running time of algorithm is $O\left(\binom{n}{3}\right)$ which is $O(n^3)$

(d) Bonus question: Give a 3×3 matrix such that the matrix (i) is symmetric, (ii) has diagonal entries of 0 and (iii) is NOT additive. If you think this is impossible, explain why. Otherwise, does your matrix satisfy the triangle equality? [5 points]

D= $\begin{bmatrix} 0 & 2 & 5 \\ 2 & 0 & 1 \\ 5 & 1 & 0 \end{bmatrix}$

D (i) is symmetric, (ii) has diagonal entries of 0 and (iii) is NOT additive. However D doesn't satisfy the triangle inequality rule.

(e) Bonus question: Give a 3×3 matrix such that the matrix (i) is symmetric, (ii) has diagonal entries of 0, (iii) satisfies the triangle inequality and (iv) is NOT additive. If you think this is impossible, explain why. [5 points]

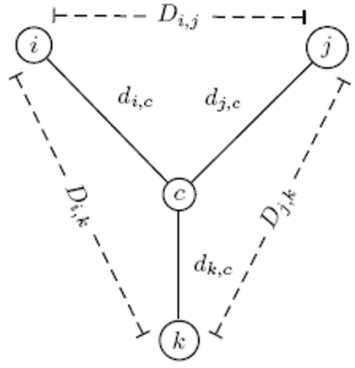
.It is impossible to construct a 3x3 matrix with the given conditions which is not additive .

Let i,j,k be nodes of matrix D = $\begin{bmatrix} 0 & D_{i,j} & D_{i,k} \\ D_{i,j} & 0 & D_{j,k} \\ D_{i,k} & D_{j,k} & 0 \end{bmatrix}$

We can see D (i) is symmetric, (ii) has diagonal entries of 0.
Now assume D satisfies triangle inequality(the sum of any two distances is greater than third one) i.e

$$D_{i,j} + D_{i,k} > D_{j,k} \tag{5}$$
$$D_{i,k} + D_{j,k} > D_{i,j} \tag{6}$$
$$D_{i,j} + D_{j,k} > D_{i,k} \tag{7}$$

Now we prove that it is impossible for D to NOT be Additive matrix by contradiction.
That is we prove that D is additive
We create a tree with c as center for the points i,j, k



And we have

$$d_{i,c} = \frac{D_{i,j} - D_{j,k} + D_{i,k}}{2} \tag{8}$$
$$d_{j,c} = \frac{D_{i,j} + D_{j,k} - D_{i,k}}{2} \tag{9}$$
$$d_{k,c} = \frac{D_{i,k} + D_{j,k} - D_{i,j}}{2} \tag{10}$$

Now for $d_{i,c}$ is always positive as from (5) $D_{i,j} + D_{i,k} > D_{j,k}$.
Similarly $d_{j,c}$ *and* $d_{k,c}$ are also positive . And we have a edge-weighted tree T with i,j,k as leaves that best fits matrix D. **Thus D is additive.**

Thus we can say that it is impossible to create a 3x3 matrix which satisfies all 4 conditions mentioned

3. **Two-state Perfect Phylogeny** [15 points]

In this question we will consider a variant on the two-state perfect phylogeny problem called the *Incomplete Directed Perfect Phylogeny Problem*. In this problem instead of observing a binary matrix M, where the m rows are samples and the n columns are characters (that are either expressed (1) or not expressed (0), you observe an incomplete matrix $M \in \{0, 1, *\}^{m \times n}$. In this matrix, 0 and 1 mean the same thing a previously, but an entry of $*$ indicates that the information was incomplete and we were unable to be measure that character in the corresponding sample. The "Directed" part of this problem means that mutations are only ever gained (i.e. we only ever go from a 0 to a 1). The following is an example of such a directed incomplete matrix M:

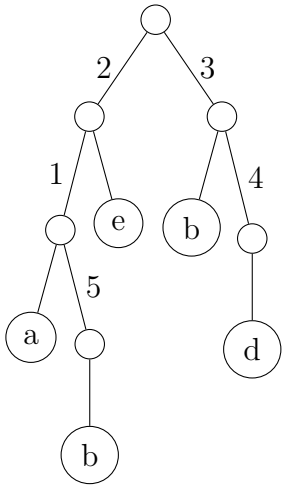
	1	2	3	4	5
a	1	1	0	0	*
b	0	*	1	0	*
c	1	*	0	0	1
d	0	0	1	1	*
e	0	1	0	0	*

(a) Fill in values to the above matrix such that it is compatible with a two-state perfect phylogeny (and construct the corresponding tree) or provide an argument as to why no such completion exists. [5 points]

The matrix that will be compatible with two-state perfect phylogeny

	1	2	3	4	5
a	1	1	0	0	0
b	0	0	1	0	0
c	1	1	0	0	1
d	0	0	1	1	0
e	0	1	0	0	0

The tree for corresponding matrix is



(b) Suppose you are given an incomplete matrix $M \in \{0, 1, *\}^{m \times n}$. Give pseudocode for a brute force algorithm to determine whether or not M admits a perfect phylogeny. Let k be the number of missing entries. Give the running time of your algorithm. [10 points]

Given there are k missing entries in the matrix. Each entry can take two possible values 0 or 1.
The idea is to generate all possible permutations of 0 and 1 of length k and use each permutation to check whether the matrix is perfect phylogeny
Label each missing entry a value i such that $0 \leq i \leq k-1$. Generate all possible permutations C of 1s and 0s of length k in following manner

$$P_j = \{s_i | s_i = 0 \; or \; 1 \; \& \; 0 \leq i \leq k-1\} \quad | \; 0 \leq j \leq 2^k - 1$$

For each permutations P_j, generate a new matrix $D_{P,j}$ by filling missing entry labelled i as i^{th} value in C_j

$$D_{P,j}[a,b] = \begin{cases} D[a][b] & a \leq m, b \leq n \; and \; D[a][b] \in \{0,1\} \\ P_j[i] & \text{where i is label of missing entry} \end{cases}$$

Check if $D_{P,j}$ is a perfect phylogeny by using the Two-State Perfect Phylogeny algorithm as discussed in class.
If for any $j \in 0 \leq j \leq a^k - 1$ Matrix $D_{P,j}$ satisfies the problem, Then we can say that given Matrix M admits a perfect phylogeny .

Running time there 2^k possible permutations for k missing entries. for each combination, to check whether a matrix$M \in \{0,1\}^{m,n}$ admits perfect phylogeny takes O(m*n) time .

Hence total running time is $O(2^k mn)$