A

BRAC UNIVERSITY
Department of Computer Science and Engineering

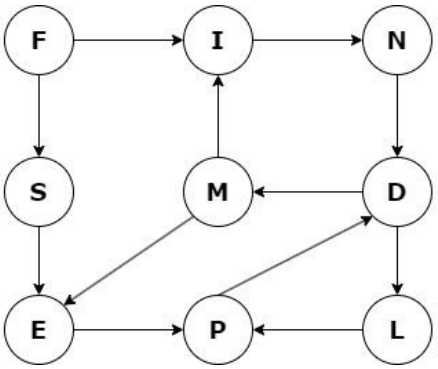
Examination: Final Exam | Semester: Fall 2023
Duration: 1 Hour 40 Minutes | Full Marks: 40

CSE 221: Algorithms

Answer all of the following questions in your script.
Figures in the right margin indicate marks.

Name:	ID:	Section:

1. Sherlock Holmes found himself in a dire situation where James Moriarty imprisoned his assistant Watson in a maze. The maze is made of 9 different rooms. Each room has a part of the key with an alphabet written on it. Moriarty tells Sherlock that only if he can attach all the key chain parts in exactly this order: **"F-I-N-D-M-E-P-L-S"**, then he can unlock the room with that key where Watson is kept. The map of the maze is given below:



a) [CO1] Now you know something about a graph traversal algorithm that travels from the source to as far as possible along each branch (depth-wise traversal). Simulate/Draw the traversal and **fill up** the following table (in your script) with the start and ending time for each node you achieved while doing this task. Keep in mind, wherever you have to choose between different neighbors to explore, follow the sequence that will make the key chain parts in the sequence that Sherlock needs. **06**

Node	D	E	F	I	M	S	L	P	N
Start									
End									

b) [CO1] When Sherlock tries to save his friend, water suddenly fills the rooms. He needs to find a way to contain the water somehow so that the water doesn't overflow from one place to another. So, he created different isolated closed groups of rooms where he can travel anywhere in that particular group, meaning they have a path from any room to any other room inside the group. After doing so, he finds out that he can make a few of these groups of rooms inside this maze. A group can contain 1 to N number of rooms where N is the total number of vertices, and no two groups can share the same room.

i) **Mention** the name of a suitable algorithm for this task. **01**
ii) **Show** simulation for the graph given above. **03**

You can reuse/refer to your answer from question (a) to reduce work.

2. a) [CO2] "Dijkstra's Algorithm can find the single source shortest paths for any graph." **02**
Do you think this statement is true? If yes, **state** your reasoning. Else, **validate** it to be false with an example graph.

b) [CO1] Imagine a daring explorer on a mission to uncover hidden treasures scattered across a cluster of islands. The listed islands have a bidirectional bridge between them: **03**

Aqua	Blaze
Aqua	Coral
Blaze	Coral
Blaze	Dune

Coral	Dune
Blaze	Ember
Dune	Ember
Dune	Flare

Coral	Flare
Ember	Flare
Ember	Glimmer
Flare	Glimmer

If each bridge has a travel time of 5 minutes, give an O(V+E) algorithm that can find the shortest travel time from Aqua to every other island.
Explain your idea in an executable code/pseudocode/flow-chart format.

c) [CO1] Continuing on the previous question, suppose the travel time of each bridge is not equal anymore. The updated travel times are as follows:

Bridge between		**Travel time**
Aqua	Blaze	5
Aqua	Coral	13
Blaze	Coral	4
Blaze	Dune	6
Coral	Dune	5
Blaze	Ember	7

Bridge between		**Travel time**
Dune	Ember	6
Dune	Flare	6
Coral	Flare	16
Ember	Flare	3
Ember	Glimmer	11
Flare	Glimmer	2

Now, the explorer wants to find a set of bridges that connects all the islands with the **minimum total travel time**. However, the bridge between Ember and Flare and the bridge between Flare and Glimmer suddenly become vulnerable and may break down anytime. So, the explorer decides to avoid these two bridges while constructing the subgraph that connects all islands.

Answer the following questions based on this scenario.
i) Which bridges will the explorer choose? **03**
ii) What will be the total travel time? **02**
Show your works on the answer script.

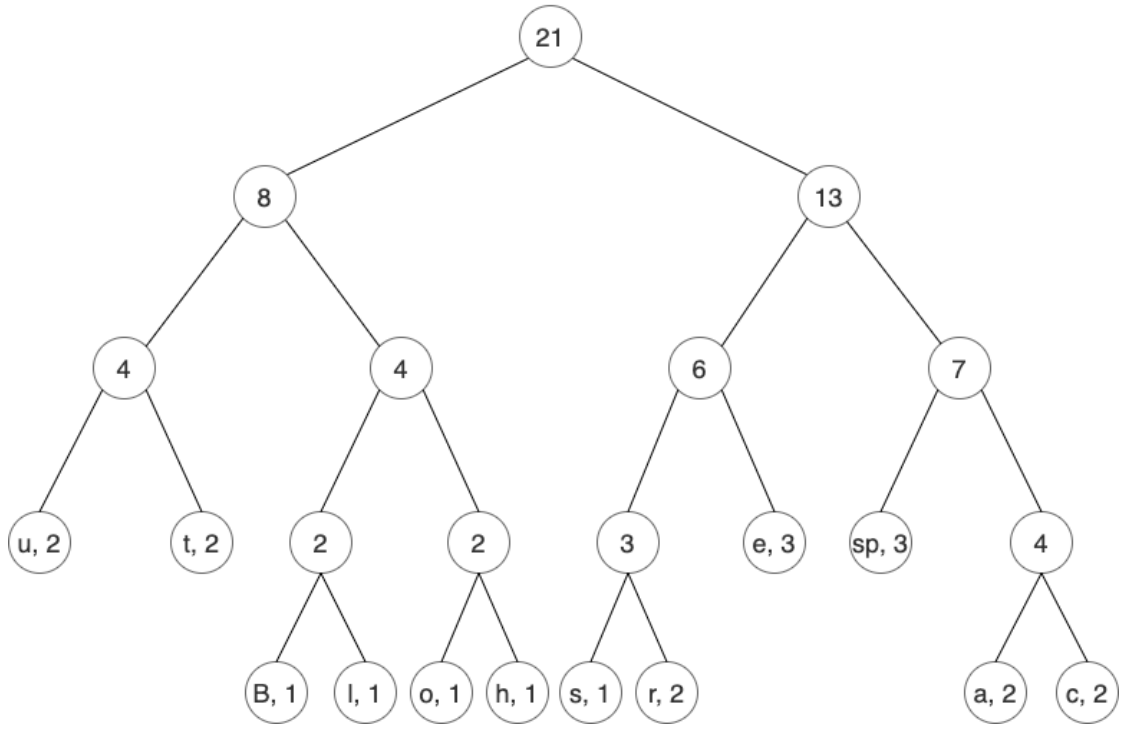
3. a) [CO1] Welcome to the exciting realm of Bracula Gaming World, where you can earn lots of money by playing different games. But there's a twist – time is limited to just 20 hours! Each game has its own time limit and prize money if played for the full duration. However, you can play for a **fraction of the time** and still earn money in proportion. Check the list of games below.
Your task is to make the most of your 20 hours and collect **as much money as possible**.

Game Name	Total Time Limit	Prize Money For Full Duration
Road Rush	15 hours	75$
GTA Vice City	20 hours	90$
Virtual Cop	3 hours	30$
House of the Dead	6 hours	55$
Far Cry	2 hours	17$
Devil May Cry	10 hours	47$

i) **Name** the algorithm tailored for this situation. **01**

ii) **Write** the names of the games and their respective playing times that will lead to maximum prize money. Show a simulation of your algorithm. **04**

b) [CO1] You are given the following tree generated by simulating the Huffman encoding algorithm on a text message.



i) **Write** the encoding for each of the characters. Consider left to be 0, and right to be 1 when traversing the tree. **04**

ii) Decode the following encoded message using the character codes you wrote on question (i). **01**
0100100111101111000010111101100010110110001011110111010011011000111100010 1

4. DNAs contain genetic and hereditary information of almost every living organism. DNA Sequence Matching is a procedure to compare two DNA sequences and find similarity between them. Given below is the recurrence relation of a dynamic programming algorithm to perform sequence matching of two DNAs.

$$seq(i,j) = \max \begin{cases} seq(i, j-1) - 1 \ , \\ seq(i-1, j) + p(i, j) \ , \\ seq(i-1, j) - 1 \end{cases}$$

where,

$$seq(i,j) = 0, \quad if\ i = 0\ or\ j = 0$$
$$p(i,j) = 1, \quad if\ DNA1[i] = DNA2[j]$$
$$p(i,j) = -2, \quad otherwise$$

As you can see, the formula takes two indices, $i\ and\ j$ of two DNA sequences given as input. Then it produces a similarity score as output. We want to apply this formula to compare the following DNA sequences.

index	1	2	3	4	5
DNA1	G	C	G	T	A
DNA2	C	T	G	A	G

Now answer the following questions:

a) [CO1] Put appropriate values to **fill the gaps** in the line below (answer in your script): **02**
$seq(\_\_, \_\_)\ represents\ the\ final\ similarity\ score\ of\ the\ given\ DNAs.$

b) [CO1] **Apply** the formula to calculate a similarity score between the DNA sequences given above. Show your work with either a recursion tree or a memory table. **05**

c) [CO2] **State** the time complexity of this algorithm with proper reasoning. **02**
You can assume that, the lengths of the DNA sequences are $N\ and\ M$ respectively.

d) [CO3] Mriaslun, a talented algorithmist, claims that he can implement the algorithm in an optimized way so that it only takes $O(min(N, M))$ space. Can you do it too? **Explain** your idea. **01**

B

BRAC UNIVERSITY
Department of Computer Science and Engineering

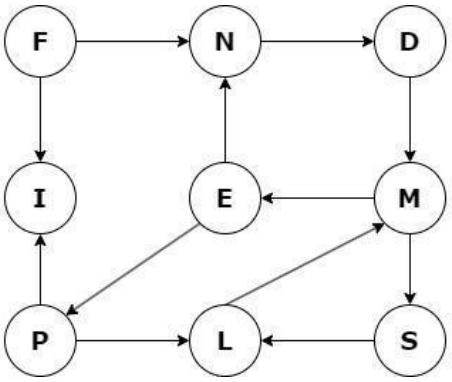
Examination: Final Exam | Semester: Fall 2023
Duration: 1 Hour 40 Minutes | Full Marks: 40

CSE 221: Algorithms

Answer all of the following questions in your script.
Figures in the right margin indicate marks.

Name:	ID:	Section:

1. Sherlock Holmes found himself in a dire situation where James Moriarty imprisoned his assistant Watson in a maze. The maze is made of 9 different rooms. Each room has a part of the key with an alphabet written on it. Moriarty tells Sherlock that only if he can attach all the key chain parts in exactly this order: **"F-I-N-D-M-E-P-L-S"**, then he can unlock the room with that key where Watson is kept. The map of the maze is given below:



a) [CO1] Now you know something about a graph traversal algorithm that travels from the source to as far as possible along each branch (depth-wise traversal). Simulate/Draw the traversal and **fill up** the following table (in your script) with the start and ending time for each node you achieved while doing this task. Keep in mind, wherever you have to choose between different neighbors to explore, follow the sequence that will make the key chain parts in the sequence that Sherlock needs. **06**

Node	D	E	F	I	M	S	L	P	N
Start									
End									

b) [CO1] When Sherlock tries to save his friend, water suddenly fills the rooms. He needs to find a way to contain the water somehow so that the water doesn't overflow from one place to another. So, he created different isolated closed groups of rooms where he can travel anywhere in that particular group, meaning they have a path from any room to any other room inside the group. After doing so, he finds out that he can make a few of these groups of rooms inside this maze. A group can contain 1 to N number of rooms where N is the total number of vertices, and no two groups can share the same room.

i) **Mention** the name of a suitable algorithm for this task. **01**
ii) **Show** simulation for the graph given above. **03**

You can reuse/refer to your answer from question (a) to reduce work.

2. a) [CO2] "Dijkstra's Algorithm can find the single source shortest paths for any graph." **02**
Do you think this statement is true? If yes, **state** your reasoning. Else, **validate** it to be false with an example graph.

b) [CO1] Imagine a daring explorer on a mission to uncover hidden treasures scattered across a cluster of islands. The listed islands have a bidirectional bridge between them: **03**

Aqua	Blaze
Aqua	Coral
Blaze	Coral
Blaze	Dune

Coral	Dune
Blaze	Ember
Dune	Ember
Dune	Flare

Coral	Flare
Ember	Flare
Ember	Glimmer
Flare	Glimmer

If each bridge has a travel time of 5 minutes, give an O(V+E) algorithm that can find the shortest travel time from Aqua to every other island.
Explain your idea in an executable code/pseudocode/flow-chart format.

c) [CO1] Continuing on the previous question, suppose the travel time of each bridge is not equal anymore. The updated travel times are as follows:

Bridge between		**Travel time**
Aqua	Blaze	5
Aqua	Coral	13
Blaze	Coral	4
Blaze	Dune	3
Coral	Dune	5
Blaze	Ember	2

Bridge between		**Travel time**
Dune	Ember	6
Dune	Flare	6
Coral	Flare	1
Ember	Flare	3
Ember	Glimmer	9
Flare	Glimmer	2

Now, the explorer wants to find a set of bridges that connects all the islands with the **minimum total travel time**. However, the bridge between Ember and Flare and the bridge between Flare and Glimmer suddenly become vulnerable and may break down anytime. So, the explorer decides to avoid these two bridges while constructing the subgraph that connects all islands.

Answer the following questions based on this scenario.
i) Which bridges will the explorer choose? **03**
ii) What will be the total travel time? **02**
Show your works on the answer script.

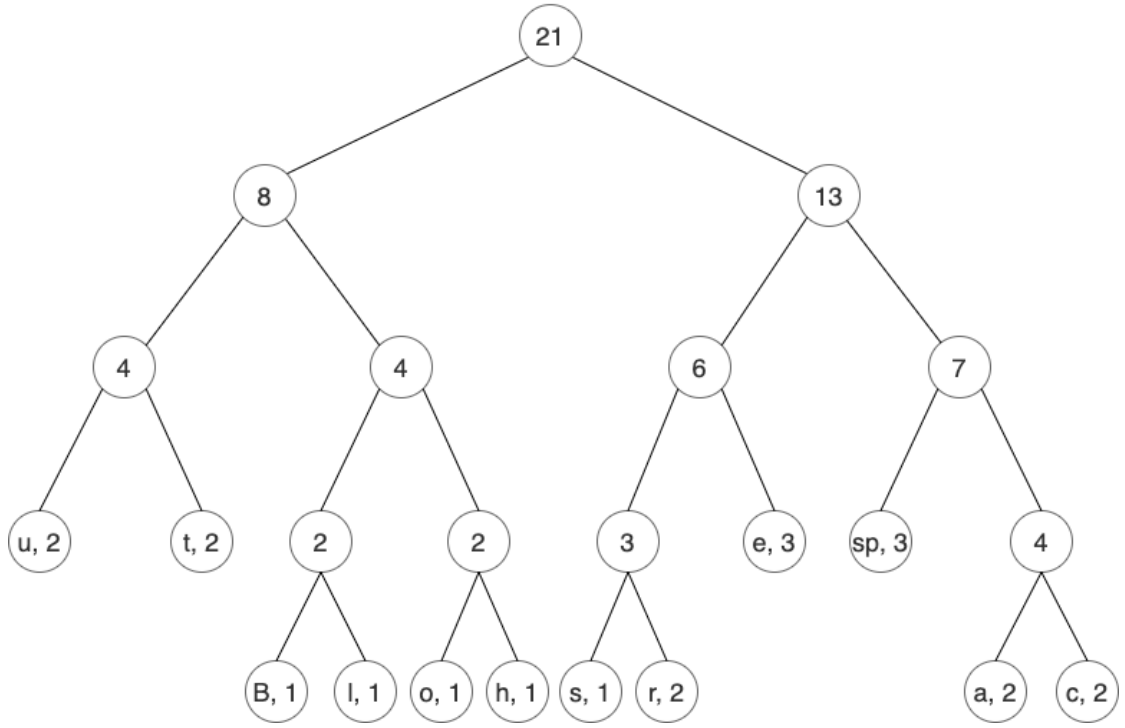
3. a) [CO1] Welcome to the exciting realm of Bracula Gaming World, where you can earn lots of money by playing different games. But there's a twist – time is limited to just 20 hours! Each game has its own time limit and prize money if played for the full duration. However, you can play for a **fraction of the time** and still earn money in proportion. Check the list of games below.
Your task is to make the most of your 20 hours and collect **as much money as possible**.

Game Name	Total Time Limit	Prize Money For Full Duration
Road Rush	16 hours	85$
GTA Vice City	20 hours	95$
Virtual Cop	4 hours	30$
House of the Dead	7 hours	45$
Far Cry	3 hours	18$
Devil May Cry	11 hours	50$

i) **Name** the algorithm tailored for this situation. **01**

ii) **Write** the names of the games and their respective playing times that will lead to maximum prize money. Show a simulation of your algorithm. **04**

b) [CO1] You are given the following tree generated by simulating the Huffman encoding algorithm on a text message.



i) **Write** the encoding for each of the characters. Consider left to be 0, and right to be 1 when traversing the tree. **04**

ii) Decode the following encoded message using the character codes you wrote on question (i). **01**
0100100111101111000010111101100010110110001011110111010011011000111100010 1

4. DNAs contain genetic and hereditary information of almost every living organism. DNA Sequence Matching is a procedure to compare two DNA sequences and find similarity between them. Given below is the recurrence relation of a dynamic programming algorithm to perform sequence matching of two DNAs.

$$seq(i,j) = \max \begin{cases} seq(i\,,j-1) - 1\,, \\ seq(i-1\,,j) + p(i\,,j)\,, \\ seq(i-1\,,j) - 1 \end{cases}$$

where,

$$seq(i,j) = 0, \quad if\ i = 0\ or\ j = 0$$
$$p(i,j) = 1, \quad if\ DNA1[i] = DNA2[j]$$
$$p(i,j) = -2, \quad otherwise$$

As you can see, the formula takes two indices, $i\ and\ j$ of two DNA sequences given as input. Then it produces a similarity score as output. We want to apply this formula to compare the following DNA sequences.

index	1	2	3	4	5	6
DNA1	G	A	T	C		
DNA2	C	T	G	A	G	C

Now answer the following questions:

a) [CO1] Put appropriate values to **fill the gaps** in the line below (answer in your script): **02**
$seq(\_\_\,,\_\_)\ represents\ the\ final\ similarity\ score\ of\ the\ given\ DNAs.$

b) [CO1] **Apply** the formula to calculate a similarity score between the DNA sequences given above. Show your work with either a recursion tree or a memory table. **05**

c) [CO2] **State** the time complexity of this algorithm with proper reasoning. **02**
You can assume that, the lengths of the DNA sequences are $N\ and\ M$ respectively.

d) [CO3] Mriaslun, a talented algorithmist, claims that he can implement the algorithm in an optimized way so that it only takes $O(min(N,M))$ space. Can you do it too? **Explain** your idea. **01**