United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Final Exam Total Marks: **40** Spring 2024

Course Code: CSE 2217 Course Title: Data Structure and Algorithms II

Time: 2 hours

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

There are **six** questions. **Answer all of them**. Show full simulation/tabulations wherever necessary. Figures in the right-hand margin indicate full marks.

First determine X and Y correctly for your student ID and write it down. Use these values in Questions 1 and 2.

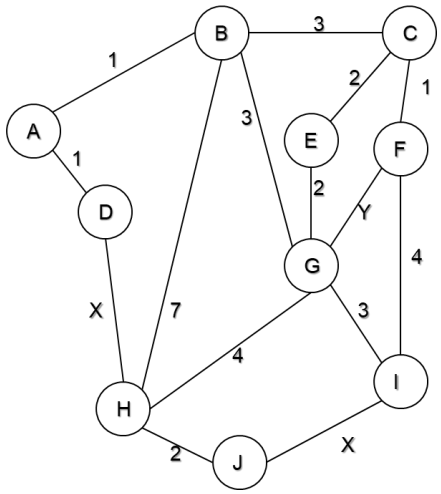
For example, a student with ID: **0111 142 111**

A B

A=142, **B**=111

X = 2 + (142 mod 5) = 2+2 = 4

Y = 2 + (111 mod 6) = 2+3 = 5

1.	**(a)** Find the Minimum Spanning Tree (MST) for the given graph in the following figure using Kruskal's algorithm. Show the details of your calculation.  Figure 1 : A weighted undirected graph	**[3]**
	(b) Given a weighted graph where multiple edges have the same weight, you are asked to apply Prim's algorithm to find MST. "*Depending on the choice of edges (from edges with same weight), you may have different MSTs*". Is the statement True or False? Justify your answer with an example.	**[2]**
	(c) How do you detect a cycle in a graph while applying Prim's algorithm for finding the MST of that graph?	**[1]**
2.	**(a)** The following table shows the parent array of a Disjoint set (Rooted tree implementation). Perform the following operations sequentially using **path compression and union-by-rank heuristic:**	

i. Draw the disjoint set forest **[2]**

ii. What will be returned by Find-Set(X), and Find-Set(5)? **[1]**

iii. Redraw the forest after Union(6, 9) **[1]**

iv. Redraw the forest after Union(9, Y) **[1]**

Index	0	1	2	3	4	5	6	7	8	9	10	11
Parent	0	0	1	0	3	4	7	8	8	9	X	Y

(b) Suppose you have designed a disjoint set data structure that supports the following operations: **[2]**

find(x) – returns the representative of x

min(x) – returns the minimum element in the set of x

union(x, y) – unifies the sets of x and y to generate a new set z

Now, what will be returned if you run the operation min(z)? Explain the process of finding the minimum element in set z.

3 **(a)** A graph contains the vertices {A, B, C, D, E, F, G} and the shortest path from A to B is A→E→C→D→F→G→B. Is it possible to find the shortest path from E to F from given data? Justify your answer. **[2]**

(b) Consider the graph in Figure 2, which represents a city's layout with various locations and connecting roads. Each edge in the graph is labeled with the distance of the corresponding road and also indicates whether the road is safe or not. Apply Dijkstra's algorithm to determine the **shortest safe paths** starting from Central Park to all other destinations in the city. You must show all the shortest paths as shown in Question 3(a) along with the corresponding distances. Note that a safe path does not contain any unsafe road. **[5]**

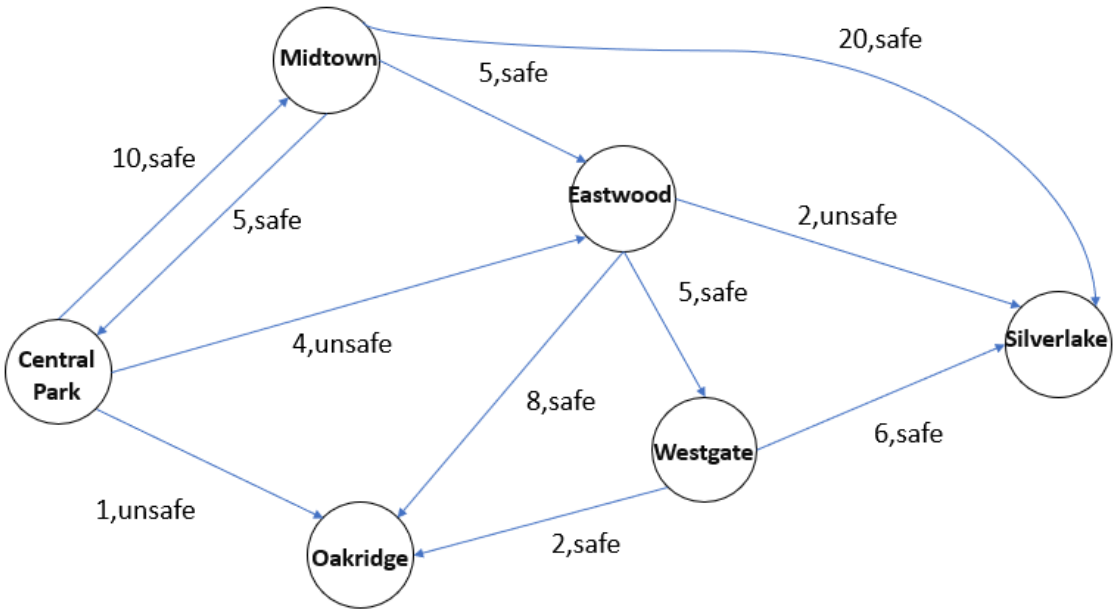


Figure 2: A graph depicting a city's layout

(c) "The BFS algorithm, in certain conditions can be used as a shortest path algorithm as well as a minimum spanning tree algorithm" - justify this statement using appropriate examples. **[2]**

	(d) Explain briefly how we can detect if a graph has any cycle or not using DFS.	**[1]**
4	**(a)** Differentiate between the NP class of problems and the NP Hard class of problems. **(b)** What are the specific conditions under which the classes P and NP could be considered equivalent? Explain briefly.	**[2]** **[3]**
5	**(a)** In the Rabin Karp algorithm, the use of strong hash functions can help in preventing spurious hits - justify the statement using a suitable example. **(b)** Consider the following text T = "237395" and pattern P = "739". Suppose that the alphabet consists of just the d = 10 digits {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}. Using ***modulo q = 13 ,*** find out with detailed steps the valid matches and spurious hits using the Rabin-Karp algorithm.	**[2]** **[3]**
6	**(a)** What are the advantages of open addressing vs chaining for collision resolution? **(b)** Consider the hash function: $h(k,i) = (h_1(k) + i*h_2(k)) \bmod 7$ *where* $h_1(k) = (3k + 3) \bmod 7$ *and* $h_2(k) = (k+3) \bmod 7$ By proper calculations, redraw Table 1 and show the following sequence of operations with proper hash value calculations. i. Insert 60 ii. Insert 54 iii. Delete 14 and replace with NIL iv. Search 7 If your search of 7 fails despite being present at the table, what might the possible reason be? Explain how you can modify the delete operation to prevent this from happening.	**[2]** **[1]** **[1]** **[1]** **[1]** **[1]**

0	1	2	3	4	5	6
9	NIL	7	14	NIL	21	NIL

Table 1: A hash table