United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

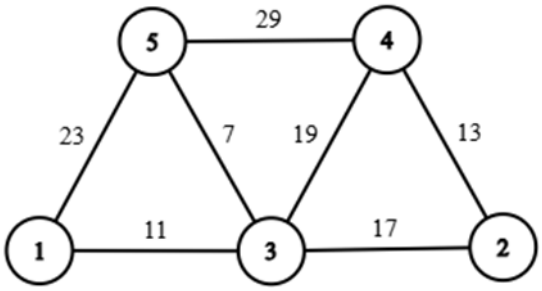
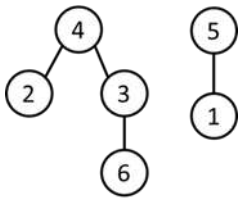
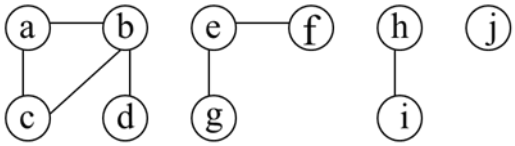
Final Exam Total Marks: **40** Fall 2024

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

1.	**(a)** Apply Prim's Algorithm to find the Maximum Spanning Tree (MST) on the following graph and calculate its total weight.  	**[4]**
	(b) Given an undirected graph with 50 vertices and 200 edges, which algorithm would you choose between Prim's and Kruskal's algorithms to find the MST? Justify your answer with proper mathematical reasoning.	**[2]**
	(c) Given a connected undirected graph with 6 vertices and 10 edges, what is the maximum number of edges that can be included in a Minimum Spanning Tree (MST)?	**[1]**
2.	**(a)** Draw the resultant forest after calling **UNION(5, 6)** and after that draw the resultant forest again after calling **FIND-SET(1)** on the disjoint-sets of the following figure. You must use the *union-by-rank* and the *path-compression* heuristics.  	**[3]**
	(b) Using disjoint-set data structures, determine whether **node c** and **node f** are in the same connected component.  	**[2]**
	(c) Suppose you are asked to utilize a disjoint set data structure to implement **Kruskal's algorithm** to find MST from a given graph. Explain, with an example, how disjoint set data	**[3]**

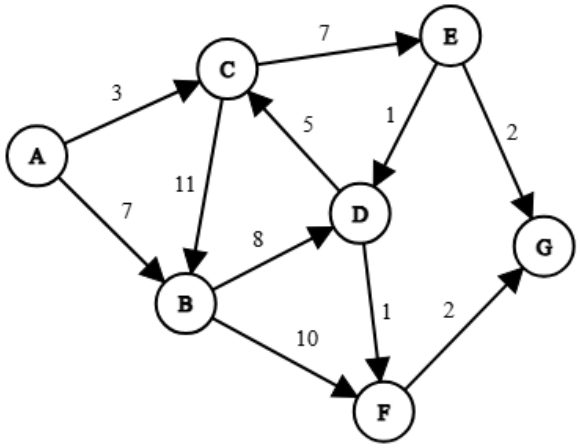
structure would help you to perform the following operations:

i) joining two trees in the forest

ii) detecting a cycle

3 **(a)** If we increase the weights of every edge in a graph by the same amount does the shortest path change? Justify your answer by showing an example. **[2]**

(b) Apply Dijkstra's algorithm to find the shortest path from source node A to every node in the following directed graph. **[4]**



(c) "Every connected directed acyclic graph (DAG) has exactly one topological ordering" - true or false? Design a graph G with exactly 4 vertices that justifies your reasoning. **[1+2]**

(d) The Bellman-Ford algorithm tries to relax every edge in each iteration. If there are n vertices, explain why the algorithm requires (n-1) iterations to correctly calculate the lengths of all shortest paths in the graph. **[2]**

4 **(a)** What is a spurious hit in the Rabin-Karp algorithm? How does the occurrence of spurious hits impact the time complexity of the Rabin-Karp algorithm? **[2]**

(b) You are tasked with building a plagiarism detection system to check if short patterns from a reference document are present in a student's submission. Suppose the reference document contains the text "academy" and you want to verify if the pattern "emy" appears in it. **[4]**

You need to apply ***Rabin-Karp's algorithm (using the rolling hash technique)*** to find whether the pattern exists in the text or not. The hash code for each character is given as follows:

.

a	c	d	e	m	y
1	2	3	4	5	6

The hash function is defined as:

$$h(xyz) = (x + y + z) \% 7$$

Where, x, y and z represent the hash codes for each character.

You must calculate the hash values for the pattern and for all substrings in the text **"academy"** using rolling hash technique. Find the valid hits and spurious hits(if any).

5	**(a)** In Chaining, what is the average cost of a successful search? (the load factor is defined as α)	**[1]**
	(b) Given that memory is not an issue to be concerned about, what is the best possible hashing scheme and why?	**[2]**
	(c) Consider an open-addressing hash table as shown below. The table already contains some data items and other empty slots. Assume that collisions are handled by Quadratic probing using the hash function: $h(k, i) = (h'(k) + 2i^2)\ mod\ 13$, where $h'(k) = (k + 7)\ mod\ 13$	
	By showing detailed calculations, redraw the table after	
	(i) Insert 47	**[1]**
	(ii) Insert 64	**[1]**
	(iii) Delete 12 (replace with NIL)	**[1]**
	(iv) Search 38.	**[1]**

0	1	2	3	4	5	6	7	8	9	10	11	12
	44					12		38				70

Although 38 had been present at the hash table, your search would fail here. Explain how one can modify the operations such that this will be prevented. **[1]**