# United International University (UIU)

Dept. of Computer Science & Engineering (CSE)

Final Exam Total Marks: **40** Fall 2023

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.**

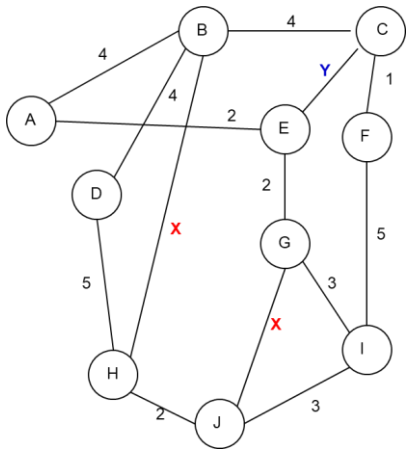
For example, a student with ID: **0111 142 001**

**A B**

**A**=142, **B**=1

**X** = 2 + (142 mod 6) = 2+4 = 6

**Y** = 2 + (1 mod 5) = 2+1 = 3

| | | |
|---|---|---|
| **1.** | **(a)** Find the Minimum Spanning Tree (MST) for the given graph in the following figure using Prim's algorithm. Show the details of your calculation.<br>  | **[3]** |
| | **(b)** 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 structure would help you to perform the following operations:<br>i) joining two trees in the forest<br>ii) detecting a cycle | **[3]** |
| **2.** | **(a)** In the Rabin-Karp Algorithm, why does it convert the strings into numeric values? | **[1]** |
| | **(b)** "There is no difference between the Rabin-Karp Algorithm and the Naive Approach of Pattern Matching in terms of Worst Case Scenario." - Do you agree with the statement? Give proper justification. | **[1]** |
| | **(c)** Imagine you are working on a database search engine that aims to efficiently retrieve information from a large dataset of product codes. Each product code is a combination of letters A through J, representing different product categories. Your task is to implement the Rabin-Karp | **[3]** |

algorithm to find occurrences of a specific product code pattern within the dataset.

The numeric values of the letters of the product code:

| A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

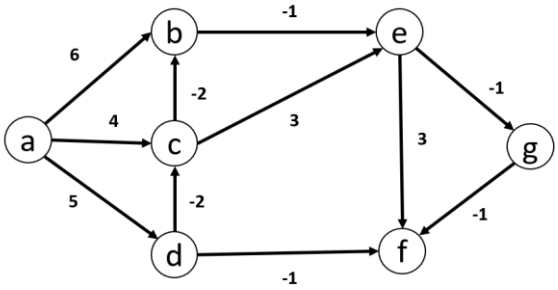
The hash function is defined as follows:
hash(s) = s[0] * 10^(n-1) + s[1] * 10^(n-2) + …… + s[n-1]*10^(n-n)
Where,

hash(s) is the hash value of string
n is the length of the string
10 is the base of the hash function.

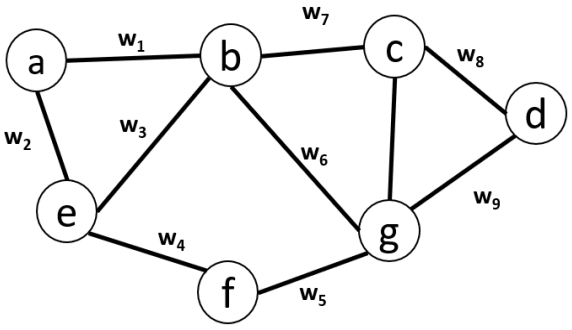
You are given a dataset of product codes, for example, "HGJABCDFGH," and you need to efficiently identify occurrences of a specific product code pattern within it. The pattern you are looking for is "CDFG". Clearly show the step-by-step calculations for the hash values, modulo operations, and the final results.

**3.** **(a)** "The Disjkstra's Algorithm might fail on graphs with negative edge weights" - do you agree with the statement? Justify your answer using an example. **[2]**

**(b)** Does the following graph contain any negative weight cycles? Justify by applying the Bellman Ford Algorithm. **[5]**




**(c)** Consider the following graph. Redraw the graph and assign the values of $w_1$, $w_2$, $w_3$, $w_4$, $w_5$, $w_6$, $w_7$, $w_8$ and $w_9$ in such a way that the BFS algorithm can be used on the graph to determine the shortest path from vertex A to all other vertices. **[1]**




**(d)** Apply BFS on the graph you generated in Question 3c considering the vertex *a* as the source vertex. Show the parent values of each vertex and draw the minimum spanning tree that is obtained based on the parent values. **[2]**

| | | |
|---|---|---|
| **4.** | **(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:**<br>i. Draw the disjoint set forest<br>ii. What will be returned by Find-Set(6), and Find-Set(5)?<br>iii. Redraw the forest after Union(0, 7)<br>iv. Redraw the forest after Union(2, 9)<br><br>**Index** 0 1 2 3 4 5 6 7 8 9 10 11<br>**Parent** 6 4 3 4 4 8 3 7 10 8 10 8<br><br>**(b)** What is the time complexity of Make-Set(x), Union(x,y) and Find-Set(x) operations in Disjoint Sets? Can you name one application where we use Disjoint Sets? | **[2]**<br>**[1]**<br>**[1]**<br>**[1]**<br><br>**[2]** |
| **5.** | **(a)** Draw the 11-item hash table that results from using the hash function<br>$h(k, i) = ( h'(k) + 2i^2 )$ ***mod*** 11, where $h'(k) = k$ ***mod*** 11, to hash the keys 50, 3, 6, 17, and 61. Assume that collisions are handled by open addressing. What kind of clustering did you encounter?<br><br>**(b)** State the difference between Direct-address Tables and Hash Tables. How do we deal with collisions in Hash tables? Describe four collision resolution techniques with proper examples. | **[4]**<br><br>**[3]** |
| **6.** | **(a)** Explain the concept of polynomial-time reduction. How is it used to establish the relationship between different complexity classes?<br><br>**(b)** Explain the concept of NP-Completeness. What does it mean for a problem to be NP-complete? | **[2]**<br><br>**[3]** |