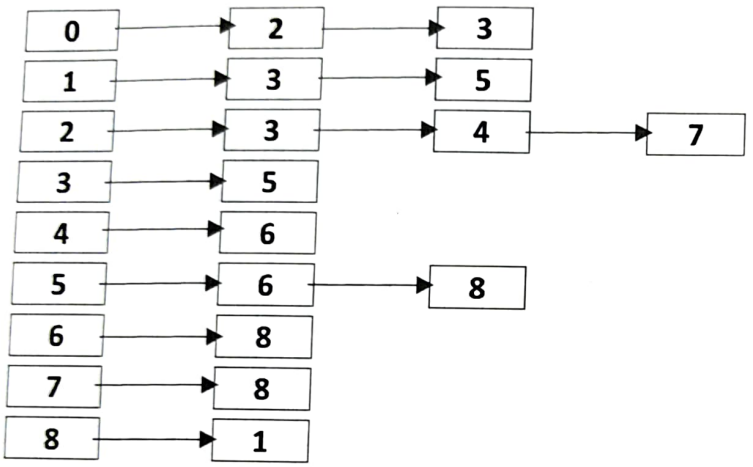
End Semester Examination
Total Marks: 64

Duration: 2 Hours

You must use pseudocode to write algorithms.

Q1. [6 Marks] Insert **38, 16, 27, 34, 31, 30, 18** one by one in an AVL tree, starting from an empty tree. After inserting all elements delete **38**. Draw the tree after every insertion and deletion. Clearly mention which rotations are performed (if applicable) during every insertion and deletion.

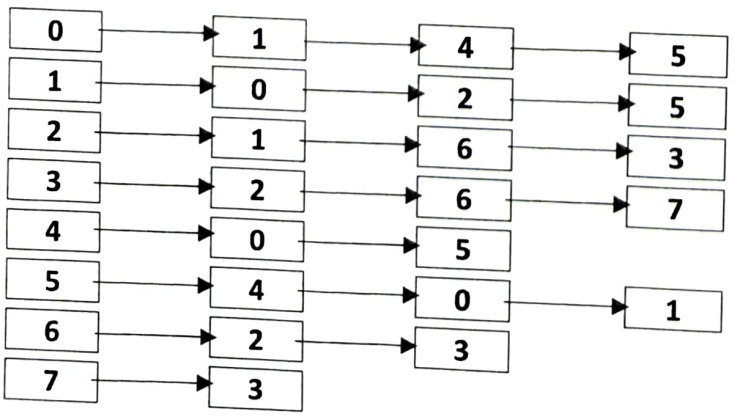
Q2. [2+5 Marks] Write an algorithm for DFS that also computes the discovery (or start) and finish time for each vertex. Derive a topologically sorted order using DFS for an acyclic directed graph represented using the following adjacency list. You need to start from vertex zero. Draw the DFS forest with discovery (or start) and finish time for each vertex. Write the corresponding topological sorted order.



Q3. [7 Marks] Write an **O(1)** algorithm to find the **third highest element** from a max-heap of integers. For example, the third highest number among numbers **1, 2, 3 , 4, 5, 6, 7** is **5**.

Q4. [8 Marks] Write an algorithm to print all **cross edges** encountered during the DFS of a directed graph.

Q5. [2+3 Marks] Write an algorithm for BFS. Compute the shortest distance to each vertex from vertex zero using BFS for an undirected graph represented using the following adjacency list. Show all intermediate enqueue and dequeue operations in the order they take place. Also, draw the final BFS tree with the shortest distance of each vertex.



Q6. [5 Marks] Prove that the time complexity of building a max heap from a given array of **n** elements using the bottom-up approach is **O(n)**.

Q7. [7 Marks] Prove that the height of an AVL tree is **O(log n)**.

Q8. [4 Marks] Give an example to demonstrate that the Dijkstra algorithm for single-source shortest-paths doesn't work for an acyclic directed graph that may also contain negative weight edges.

Q9. [6 Marks] Insert the following keys one by one in a hash table using the double hashing scheme starting from an empty hash table. For each insertion, write the number probes that are required and the index at which the element is inserted in the hash table. For example, the number of probes required to insert 7 is zero. Both the functions used for the double hashing are the same and defined as follows. The size of the hash table is 13.

```
int HashFunction(int key) {
    return key % 13;
}
```

Keys: **7 33 8 14 46 27 21**

Q10. [4 Marks] Generate the Huffman codes for the characters **a, b, c, d, e, f**. The frequency of their occurrences is given in the table below. Draw the tree and mark each character with its corresponding encoding.

Character	Frequency
A	41
B	63
C	13
D	86
E	5
F	2

Q11. [5 Marks] Give an example to demonstrate that the minimum spanning tree of a connected undirected graph with more than **n+1** edges, where **n** is the number of vertices, may contain the edge with the maximum weight. Each edge has a distinct weight. Show the graph and the corresponding minimum spanning tree.