Time: 3 hours only

Question 1. [5 + 5 + 5] [CLO-I, C2]

a. You are required to compare **Arrays** and **Linked Lists** in terms of their **memory allocation** and **insertion/deletion complexity**.

b. Using the stack data structure, evaluate the following postfix expression gradually, show the contents of the stack after each operation, and provide the result.

```
12, 7, 3, -, /, 2, 1, 5, +, *, +
```

c. A printer server uses a **Circular Queue of maximum size 07** to manage print jobs. Each print job includes:

- Job ID
- No. of pages

Perform the following operations on the circular queue, ensuring all necessary validation checks for enqueue and dequeue are applied:

1. Add new print jobs to the queue (enqueue).
2. Process (print) jobs (dequeue).

Perform a dry run for the following operations on a circular queue, and illustrate the array showing the positions of `front` and `rear` after each step:

- Enqueue 5 jobs
- Dequeue 3 jobs
- Enqueue 3 more jobs
- Dequeue 2 more jobs
- Display queue
- Simulate **overflow** when a new jobs arrives at a full queue.

Question 2. [10 + 5 + 5][CLO-II, C2]

a. You are required to design and implement selected modules of the ***Muhammad Umar Timber Store*** system using a **Binary Search Tree (BST)** approach. Each BST node represents a timber record and defined by the following class structure:

```
class node{
   node left, right;
   int T_ID;
   int T_Qty;

   void create(node ptr){
```

Complete the following tasks:

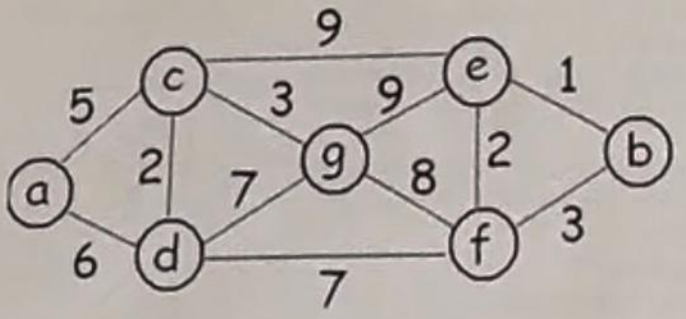
1. Implement the `create(node ptr)` method to construct the BST with proper validation
2. Develop a recursive method `leaf_count(node ptr)` to compute the total number of leaf nodes in the BST.

b. Define the balance factor in an AVL tree and explain how it is calculated. Additionally, briefly explain the following **AVL tree rotation techniques**, illustrating each with a suitable example:

1. Single Right Rotation (LL case)
2. Single Left Rotation (RR case)
3. Left–Right Rotation (LR case)
4. Right–Left Rotation (RL case)

c. A connected, weighted, undirected graph **G (V, E)** is given.

1. Apply **Prim's Algorithm** to find the **Minimum Spanning Tree (MST)** of the graph, starting from a specified vertex.
2. Show the **selection of edges at each step** and compute the **total cost** of the MST
3. Draw the final MST obtained.



Question 3. [5 + 5 + 5] [CLO-III, C3]

a. Explain briefly the two main factors of algorithm complexity and give a brief description of constant, linear, quadratic, and logarithmic complexity classes with suitable examples.

b. Describe the preconditions required for Binary Search to operate correctly. Explain its algorithm and working mechanism. Perform a systematic trace of Binary Search to find 73 in the given sorted array, clearly indicating `low`, `mid`, and `high` at each step. ***(Note: algorithm / code is NOT required)***

`20, 27, 43, 57, 65, 73, 81, 85, 87, 89`

c. Demonstrate the working of Merge Sort, and Quick Sort by applying them to the following array and