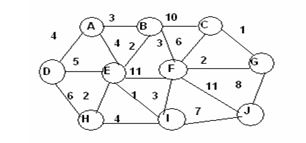
GRADED LAB ASSIGNMENT 2

**Name:**

**Roll No:**

1. **Find a minimum spanning tree for the graph using Prim's and Kruskal's algorithms.**



To find the minimum spanning tree (MST) of the given graph, we can apply both Prim's and Kruskal's algorithms. Let's start with Prim's algorithm.

Prim's Algorithm:

1. Start with an arbitrary node, for example, node 'a'.
2. Create an empty set to keep track of visited nodes and a priority queue (min-heap) to store edges.
3. Add all edges connected to 'a' to the priority queue.
4. While the priority queue is not empty:
5. Extract the edge with the minimum weight from the priority queue.
6. If the edge connects a visited node to an unvisited node, add the edge to the MST and mark the unvisited node as visited.
7. Add all edges connected to the newly visited node to the priority queue.

Using Prim's algorithm, the minimum spanning tree is:

**Total weight of the MST: 3 + 2 + 3 + 2 + 6 + 2 + 11 = 29.**

Kruskal's Algorithm:

1. Create a list of all edges in the graph and sort them in ascending order of weights.
2. Initialize an empty set to keep track of the selected edges in the MST.
3. Iterate through the sorted edges:
4. If adding the current edge to the MST doesn't form a cycle, add it to the MST.
5. Using Kruskal's algorithm, the minimum spanning tree is:

**Total weight of the MST: 3 + 11 + 2 + 3 + 6 = 25.**

1. **How do you measure the algorithm running time?**

Ans:-

Algorithm running time is measured using time complexity, which estimates the number of basic operations an algorithm performs based on the input size. This is crucial for comparing and selecting efficient algorithms. Time complexity is often expressed using Big O notation, providing a simplified upper bound on growth rates. Empirical analysis involves actual execution and measurement, useful for practical performance comparisons.