# **Experiment No 7**

#### Title: You have a business with several offices; you want to lease phone lines to connect them up with each other; and the phone company charges different amounts of money to connect different pairs of cities. You want a set of lines that connects all your offices with a minimum total cost. Solve the problem by suggesting appropriate data structures.

#### Objectives:

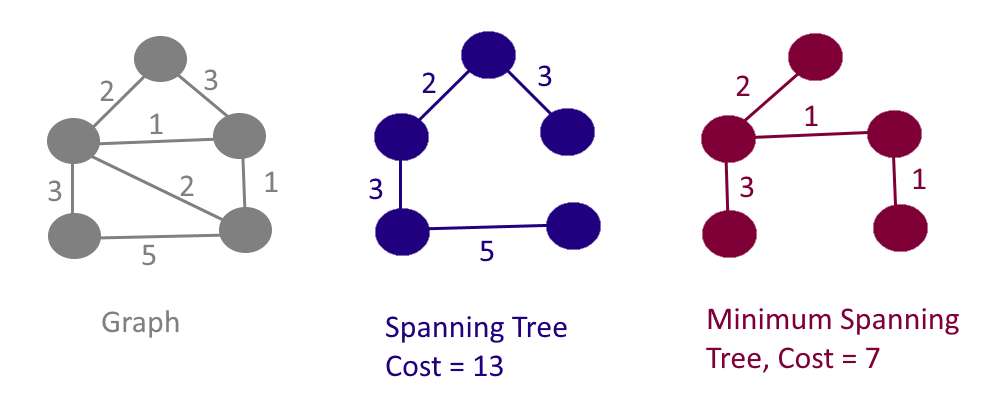
1. To understand concept of Minimum Spanning tree (MST).
2. To understand various algorithm to find MST.

#### Outcomes:

* To find MST for given graph.
* To implement various algorithms to find MST.

#### Theory:

Consider given a connected and undirected graph, a *spanning tree* of that graph is a subgraph that is a tree and connects all the vertices together. A single graph can have many different spanning trees. A *minimum spanning tree (MST)* or minimum weight spanning tree for a weighted, connected, undirected graph is a spanning tree with a weight less than or equal to the weight of every other spanning tree. The weight of a spanning tree is the sum of weights given to each edge of the spanning tree.  
*How many edges does a minimum spanning tree has?*   
A minimum spanning tree has (V – 1) edges where V is the number of vertices in the given graph.



**There are two algorithms to find MST.**

1. **Prims Algorithm**
2. **Kruskal Algorithm**

**Prims Algorithm:**

It is a greedy algorithm. It starts with an empty spanning tree. The idea is to maintain two sets of vertices:

* Contain vertices already included in MST.
* Contain vertices not yet included.

At every step, it considers all the edges and picks the minimum weight edge. After picking the edge, it moves the other endpoint of edge to set containing MST.

### **Steps for finding MST using Prim's Algorithm:**

1. Create MST set that keeps track of vertices already included in MST.
2. Assign key values to all vertices in the input graph. Initialize all key values as INFINITE (∞). Assign key values like 0 for the first vertex so that it is picked first.
3. While MST set doesn't include all vertices.
   1. Pick vertex u which is not is MST set and has minimum key value. Include 'u'to MST set.
   2. Update the key value of all adjacent vertices of u. To update, iterate through all adjacent vertices. For every adjacent vertex v, if the weight of edge u.v less than the previous key value of v, update key value as a weight of u.v.

**Time Complexity of Prims Algorithm:**

The time Complexity of the above program is O(V^2). If the input graph is represented using adjacency list, then the time complexity of Prim’s algorithm can be reduced to O(E log V) with the help of binary heap.

## **Kruskal's Algorithm:**

An algorithm to construct a Minimum Spanning Tree for a connected weighted graph. It is a Greedy Algorithm. The Greedy Choice is to put the smallest weight edge that does not because a cycle in the MST constructed so far.

**If the graph is not linked, then it finds a Minimum Spanning Tree.**

**Steps for finding MST using Kruskal's Algorithm:**

**1.** Sort all the edges in non-decreasing order of their weight.   
**2.** Pick the smallest edge. Check if it forms a cycle with the spanning tree formed so far. If cycle is not formed, include this edge. Else, discard it.   
**3.** Repeat step#2 until there are (V-1) edges in the spanning tree.

**Software Required:** g++ / gcc compiler- / 64 bit Fedora, eclipse IDE

**Input**:

1. Number of nodes in a graph.

2. Edge value for each nod.

**Output:** Threaded Binary tree.

**Conclusion:** This program gives us the knowledge of Threaded Binary tree and all its operations.

**OUTCOME**

**Upon completion Students will be able to:**

* **ELO1:** To effectively find MST of given graph.
* **ELO2:** To implement various algorithms to find MST.
* **ELO3:** To use MST in various real world application.

**Questions.**

1. What is Minimum spanning tree?
2. What are different algorithms to find MST?
3. Write and explain prims algorithm to find MST of given graph.