Assignment No. 4

Aim: A business house has several offices in different countries, they want to lease phone lines to connect them with each other and the phone company charges different rent to connect different pairs of cities. Business house want to connect all its offices with a minimum total cost. Solve the problem by suggesting appropriate data structures in C++.

Objectives:
1. To understand minimum spanning tree of a Graph.
2. To understand how Prim‘s algorithm works.

Theory:

- **Spanning Tree:**

A Spanning Tree of a graph G = (V, E) is a sub graph of G having all vertices of G and no cycles in it.

- **Minimal Spanning Tree**: The cost of a graph is the sum of the costs of the edges in the weighted graph. A spanning tree of a graph G= (V, E) is called minimal cost spanning tree or simply the minimal spanning tree of G if its cost is minimum.
    - When a graph G is connected, depth first or breadth first search starting at any vertex visits all the vertices in G.
    - The edges of G are partitioned into two sets i.e. T for the tree edges & B for back edges. T is the set of tree edges and B for back edges. T is the set of edges used or traversed during the search & B is the set of remaining edges.
    - The edges of G in T form a tree which includes all the vertices of graph G and this tree is called as spanning tree.

- **Definition:** Any tree, which consists solely of edges in graph G and includes all the vertices in G, is called as spanning tree. Thus for a given connected graph there are multiple spanning trees possible. For maximal connected graph having _n‘ vertices the number of different possible spanning trees is equal to n.
- **Cycle:** If any edge from set B of graph G is introduced into the corresponding spanning tree T of graph G then cycle is formed. This cycle consists of edge (v, w) from the set B and all edges on the path from w to v in T.

- **Prim's algorithm:** Prim‘s algorithm is an algorithm in graph theory that finds a minimum spanning tree for a connected weighted graph. This means it finds a subset of the edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized.

  An arbitrary node is chosen initially as the tree root (any node can be considered as root node for a graph). The nodes of the graph are then appended to the tree one at a time until all nodes of the graph are included. The node of the graph added to the tree at each point is that node adjacent to a node of the tree by an arc of minimum weight. The arc of minimum weight becomes a tree arc containing the new node to tree. When all the nodes of the graph have been added to the tree, a minimum spanning tree has been constructed for the graph.

Algorithm / Pseudo code:

- **Prim's Algorithm:**

All vertices of a connected graph are included in the minimum spanning tree. Prim's algorithm starts from one vertex and grows the rest of tree by adding one vertex at a time by adding associated edge in T. This algorithm iteratively adds edges until all vertices are visited.

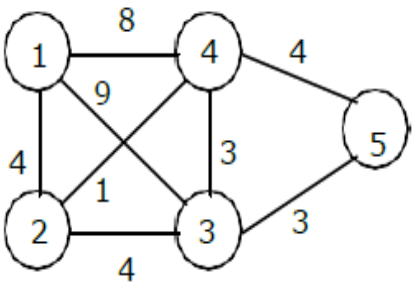
```
void prims (vertex i)
1. Start
2. Initialize visited [ ] to 0
        for (i=0;i<n; i++)
                visited [ i ] = 0;
3. Find minimum edge from i
        for (j=0;j<n; j++)
        {
        if (min > a [i] [j])
                {
                        min = a[i] [j]
                        x = i;
                        y = j;
                }
        }
4. Print the edge between i and j with weight.
5. Make visit [i++] = x
   visit [j++] = y

6. Find next minimum edge starting from nodes of visit array.
7. Repeat step 6 until all the nodes are visited.
8. End.
```

Program:

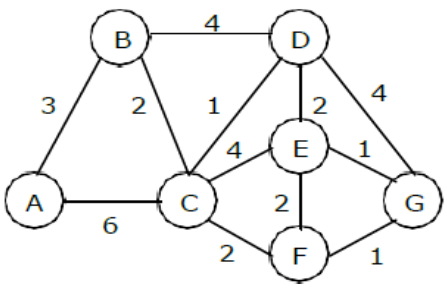
Output:

Solve the examples:

1.



2.



Conclusion: