Qs Explain kruskal's algorithm to find minimum spanning tree with the help of an example.

1. In kruskal's algorithm, the n vertices of the graph are considered as 'n' distinct partial trees.
2. At each step of the algorithm, two distinct partial trees are connected into a single partial tree by an edge of the graph, which should be of ***minimum*** weight.
3. After (n-1) steps, only one tree exists and this is the minimum spanning tree.
4. Consider the following graph.

   The following are the stages of kruskal's algorithm:

5. Algorithm

```
1) Start
2) Let G be a connected weighted graph having v vertices and e edges.
3) Let X be the set of all edges of G arranged in increasing order of weights.
4) Let T be the minimum spanning tree which is currently empty.
5) while( the number of edges in T is not equal to V-1   AND   X is not empty )
   {
         Let w be the next edge of minimum weight in X.

         Remove w from X

         if(w does not create a cycle in T)
               Add w to T.
         else
               Discard      w.
   }
6) Stop
```

6. Working

7. The efficiency of kruskal's algorithm is O(e log v) where e is the number of edges and v is the number of vertices.

Qs Explain Prim's algorithm to find minimum spanning tree with the help of an example.

1. In prim's algorithm, any vertex of the graph is initially chosen as the root of minimum spanning tree
2. The vertices of the graph are then added to the tree one by one as long as all the vertices are not included in the tree
3. The vertex of the graph, which is selected to be added to the tree, is that vertex which connects to the tree by one edge of minimum weight. It is obvious that this edge should not result into on cycle.
4. In other words, ***always select that edge of minimum weight whose one vertex is already in the tree and another vertex is not in the tree.***
5. Consider following example
6. Following are the stages of Prim's algorithm if vertex 1 is assumed to be root of the minimum spanning tree.

7. Algorithm

    1. Start
    2. Let G be a connected weighted undirected graph. Let V be the set of vertices of G.
    3. Let T be the set of edges and W be the set of vertices of minimum spanning tree of G. T and W are currently empty.
    4. Choose any vertex r Є V as the root of the minimum spanning tree. Hence W = {r}.
    5. While (W ≠ V)
       {

       Find a minimum weight edge (x,y) where x Є W and y Є V – W.

       T = T U {(x,y)}.

       W = W U {y}.

       }
    6. Stop

8. The efficiency of Prim's algorithm is $O(v^2)$ where v is the number of vertices.

Q.7 Explain single source DSPA (Dijkstra's Shortest Path Algorithm)

1. Very often it is required to find shortest path between any two vertices of a weighted graph or network. For example, while travelling from one city to another, we may want to take that path which has minimum distance or minimum cost. Such problems can be solved using DSPA.

2. This algorithm starts with one vertex as the source vertex of the tree and stops when all the vertices are included in the tree.

3. The following steps are then used for tree creation:

    i) Insert the source vertex as the very first vertex in the tree.

    ii) From every vertex in the tree, examine the ***total path length*** to all the adjacent vertices not in the tree. Select that path (or edge or vertex) which gives the minimum total path length and insert it into the tree.

    iii) Repeat step (ii) as long as all the vertices are not included in the tree.

4. Consider the following graph:

   Let vertex 1 be the source vertex. The following are the stages of Dijkstra's shortest path algorithm.

5. The tree thus created is called 'Shortest Paths Spanning Tree' and it gives lengths of shortest paths from source vertex to all other vertices.
6. This is single source DSPA. For all pairs DSPA, we will have to execute the above steps for each vertex as source vertex.
7. The efficiency of single source DSPA is $O(v^2)$ where v is the number of vertices.