Page No.
Date

# Tutorial – 6

Sol 1:- Minimum Spanning Tree – It is a subset of edges of a connected, edge-weighted undirected graph that connects all the vertices together, without any cycle and with the minimum possible total edge weight.

Applications –

1) Suppose you wants to construct highway or railroad spanning several cities then we can use the concept of minimum spanning Tree.
2) Designing of LAN.
3) Laying pipeline connecting offshore drilling sites, refineries and consumer markets.
4) Suppose you want to apply a set of houses with
   - Electric power
   - Water.

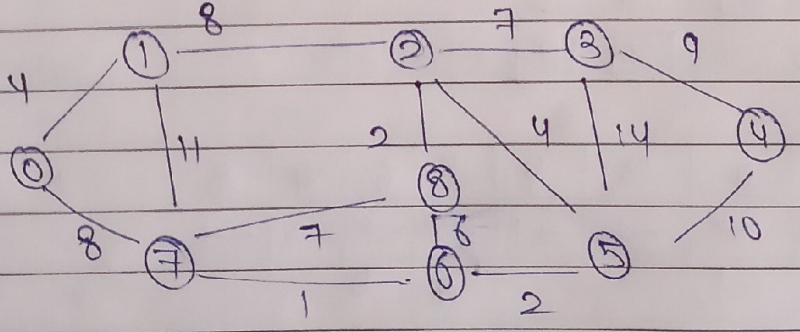
Sol 2- Prim's Algorithm ⇒ T.C = $O(|E| \log |V|)$
S.C = $O|V|$

Kruskal's Algorithm ⇒ T.C = $O(|E| \log |E|)$
S.C = $O(|V|)$

Dijkstra's Algorithm ⇒ T.C = $O(V^2)$
S.C = $O(V^2)$

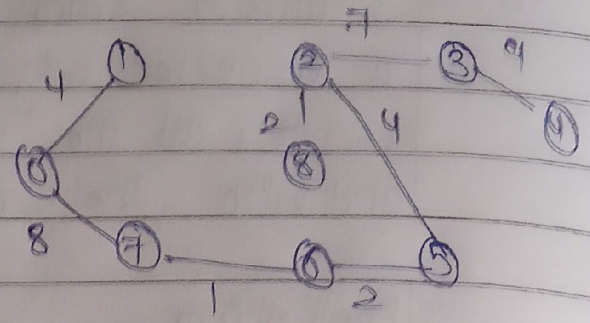
Bellman's ford Algorithm ⇒ T.C = $O(VE)$
S.C = $O(E)$

Sol 3-



- Kruskal's algorithm —

| (Source) U | (Destination) V | (Weight) W | |
|---|---|---|---|
| 6 | 7 | 1 | ✓ |
| 5 | 5 | 2 | ✓ |
| 2 | 8 | 2 | ✓ |
| 0 | 1 | 4 | ✓ |
| 2 | 5 | 4 | ✓ |
| 6 | 8 | 6 | X |
| 2 | 3 | 7 | ✓ |
| 7 | 8 | 7 | X |
| 0 | 7 | 8 | ✓ |
| 1 | 2 | 8 | X |
| 4 | 3 | 9 | ✓ |
| 4 | 5 | 10 | X |
| 1 | 7 | 11 | X |
| 3 | 5 | 14 | X |




Weight = 1+2+2+4+4+7+8+9 = 37

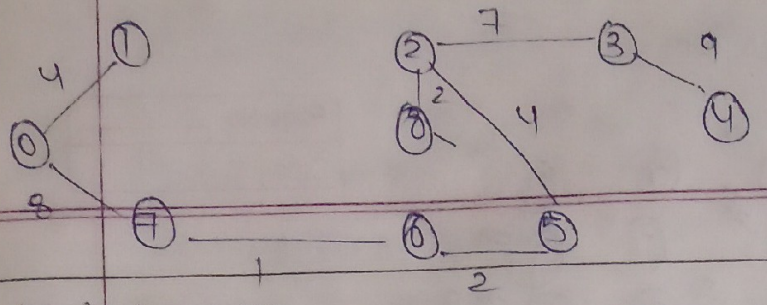
- Prism algorithm —

Weight

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| [0] ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |
| | [4] | | | | | | [8] | |
| | | [8] | | | | [11] | | 7 |
| | 11 | | 7 | | 4 | | | [2] |
| | | | | | [2] | | | 6 |
| | | [4] | 14 | 10 | | | | |
| | | | [7] | [9] | | | | |

Parent :-

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| -1 | -1 | 7 | -1 | -1 | -1 | 7 | -1 | -1 |
| | 0 | 1 | | | 7 | 0 | | |

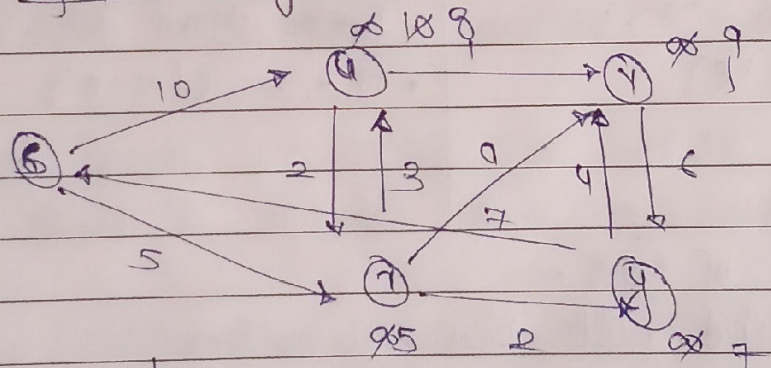



Weight = 37.

Sol 4 i) The shortest path may change. The reason is there may be different number of edges in diff. paths from 's' to 't'. For example, let shortest path be of weight 15 and has 5 edges. Let there be another path with 2 edges and total weight 25. The weight of the shortest path is increased 5 * 10 and become 5+50. Weight of other path is increased by 2 * 10 and becomes 25+20. So the shortest path changes to the other path with weight as 45.

ii) If we multiply all edges weight by 10, the shortest path doesn't change. The reason is simple. weight of all paths from 's' to 't' get multiplied by same amount. The number of edges on a path doesn't matter. It is like changing units of weights.

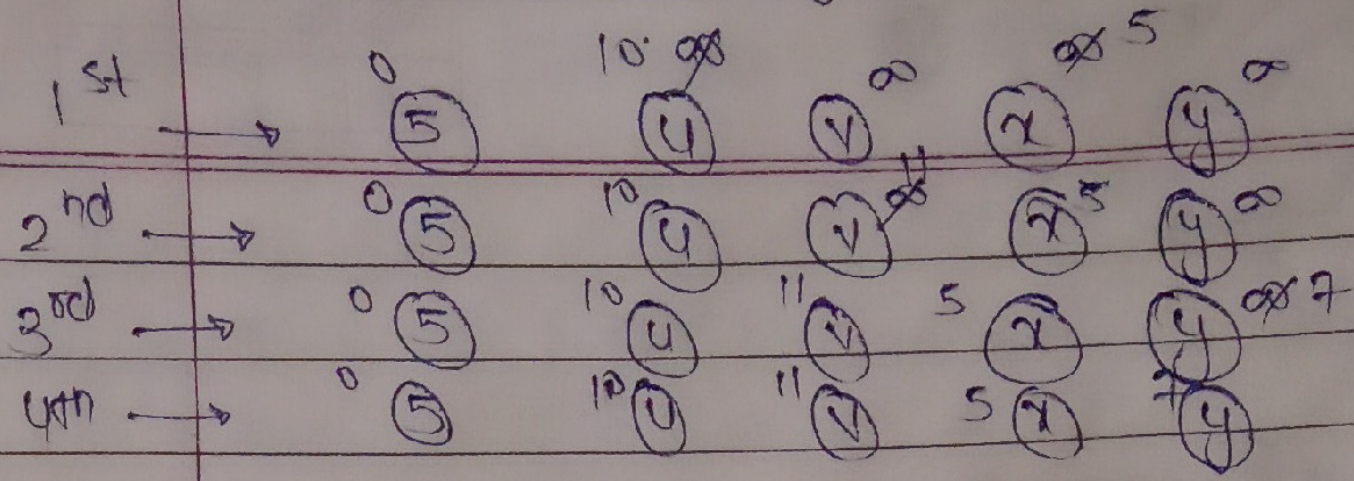
Sol 5- i) Dijkstra algorithm -




| node | shortest distance from source node |
|---|---|
| v | 8 |
| x | 5 |
| v | 9 |
| y | 7 |

# Bellman ford algorithm

Page No.
Date




final graph.