Tutorial-6

Q1) What do you mean by minimum spanning tree? What are the applications of MST?

→ The cost of spanning tree is the sum of weights of all the edges in the tree. There can be many spanning tree. Minimum Spanning Tree is a spanning tree where the cost is minimum among all the spanning tree. There can be multiple minimum spanning trees also.

Minimum Spanning tree has direct application in the design of networks. It is used in algorithms approximating the travelling salesman problem, multi-terminal minimum cut problem and minimum-cost weighted perfect matching.

Its other practical application are:

1. Image Segmentation
2. Handwriting recognition
3. Cluster analysis.

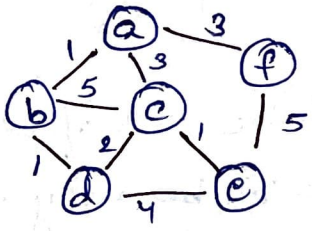
Q2) Analyse the space and time complexities of Prim's, Kruskal's, Dijkstra's and Bellman Ford's algorithms.

→

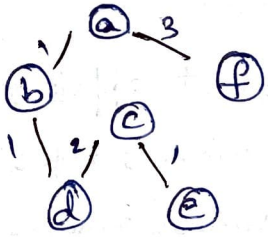
Algorithm	Time Complexity	Space Complexity
Prim's algorithm	$O(V^2)$ $O(E \log V)$ using fib. Series	$O(\|V\|)$
Kruskal's algorithm	$O(E \log E) = O(E \log V)$	$O(\|V\|)$
Dijkstra's algorithm	$O(E \log V)$ using Priority Queue	$O(\|V\| + \|E\|)$
Bellman ford's algorithm	$O(V \cdot E)$	$O(V)$

Q3) Apply Kruskal's and Prim's algorithm on graphs given below to compute MST and its weight:

(a)



∴ MST



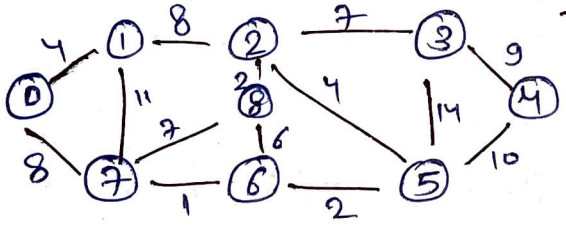
→ Appling Kruskal's algorithm

u	v	w	
a	b	1	✓
b	d	1	✓
c	e	1	✓
c	d	2	✓
a	e	3	✗
a	f	3	✓
d	e	4	
b	c	5	
e	f	5	

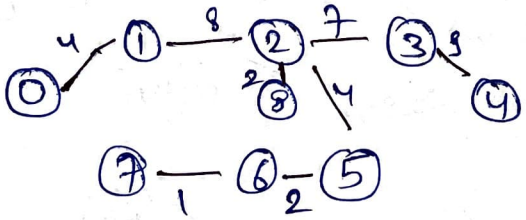
Edges in increasing order of weight

Minimum Weight = 8

(b)



∴ MST



→ Applying Prim's algorithm,

E	W	
(0,1)	7	✓
(0,7)	8	✗
(1,7)	11	✗
(1,2)	8	✓
(2,3)	7	✓
(2,8)	2	✓
(2,5)	4	✓
(8,7)	7	✗
(8,6)	6	✗
(5,6)	2	✓
(5,3)	14	✗
(5,4)	10	
(6,7)	1	✓
(3,4)	9	✓

Minimum Weight = 37

(Q4) Given a directed weighted graph. You are also given the shortest path from a source vertex 'S' to a destination vertex 'T'. Does the shortest path remain same in the modified graph in following case :-

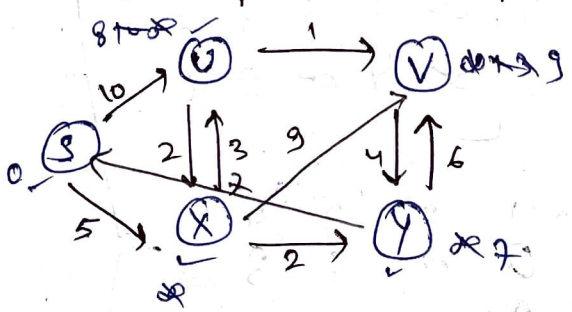
(a) If weight of every edge is increased by 10 units?

(b) If weight of every edge is multiplied by 10 units?

→ (a) The shortest path may change. The reason is, there may be a different no. of edges in different paths from 'S' to 'T'. e.g., let shortest path be of weight 15 and has 5 edges. let there be another path with 2 edges & total weight 25. The weight of the shortest path is increased by 5*10 and becomes 15+50.
Weight of other path is increased by 2*10 becomes 25+20. So, the shortest path change to the other path with weight as 45.

(b) If we multiply all edge weights by 10, shortest path doesn't change. The no. of edges on a path doesn't matter.

(Q5) Apply Dijkstra's and Bellman's algorithm on the graph given below to compute shortest path to all nodes from node 's':



→ Applying Dijkstra's algorithm,

Node	Shortest Distance from source node 's'
U	8
V	9
X	5
Y	7

Applying Bellman ford's algorithm.

- Initially,

0	∞	∞	∞	∞
S	U	V	X	Y

- Iteration - 1

0	~~∞~~ 8	~~∞~~ 11	~~∞~~ 5	~~∞~~ 7
S	U	V	X	Y

0	8	11	5	7
S	U	V	X	Y

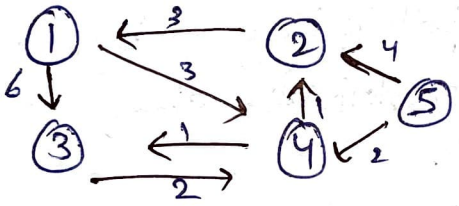
- Iteration - 2

0	8	~~11~~ 9	5	7
S	U	V	X	Y

0	8	9	5	7
S	U	V	X	Y

Node	Shortest distance from source node 's'
U	8
V	9
X	5
Y	7

Q6) Apply all pair shortest path algorithm. Floyd Warshall's on the given graph and also analysis the time & space complexity of algorithm.



→

$$D_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 0 & \infty & 6 & 3 & \infty \\ 3 & 0 & \infty & \infty & \infty \\ \infty & \infty & 0 & 2 & \infty \\ \infty & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$D_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} 0 & \infty & 6 & 3 & \infty \\ 3 & 0 & 9 & 6 & \infty \\ \infty & \infty & 0 & 2 & \infty \\ \infty & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix} \end{array}$$

- $A^0[5,2]$ 4 $<$ $A^0[5,1]+A^0[1,2]$ $\infty+\infty$
- $A^0[5,3]$ ∞ $=$ $A^0[5,1]+A^0[1,3]$ $\infty+6$
- $A^0[5,4]$ 2 $<$ $A^0[5,1]+A^0[1,4]$ $\infty+3$
- $A^0[2,3]$ ∞ $>$ $A^0[2,1]+A^0[1,3]$ $3+6=9$
- $A^0[2,4]$ ∞ $>$ $A^0[2,1]+A^0[1,4]$ $3+3=6$
- $A^0[2,5]$ ∞ $=$ $A^0[2,1]+A^0[1,5]$ $3+\infty$
- $A^0[3,2]$ ∞ $=$ $A^0[3,1]+A^0[1,2]$ $\infty+\infty$
- $A^0[3,4]$ 2 $<$ $A^0[3,1]+A^0[1,4]$ $\infty+3$
- $A^0[3,5]$ ∞ $=$ $A^0[3,1]+A^0[1,5]$ $\infty+\infty$
- $A^0[4,2]$ 1 $<$ $A^0[4,1]+A^0[1,2]$ $\infty+\infty$
- $A^0[4,3]$ 1 $<$ $A^0[4,1]+A^0[1,3]$ $\infty+6$
- $A^0[4,5]$ ∞ $=$ $A^0[4,1]+A^0[1,5]$ $\infty+\infty$

$D_2 =$

	1	2	3	4	5
1	0	∞	6	3	∞
2	3	0	9	6	∞
3	∞	∞	0	2	∞
4	4	1	1	0	∞
5	7	4	13	2	0

$D_3 =$

	1	2	3	4	5
1	0	∞	6	3	∞
2	3	0	9	6	∞
3	∞	∞	0	2	∞
4	4	1	1	0	∞
5	7	4	13	2	0

$D_4 =$

	1	2	3	4	5
1	0	4	4	3	∞
2	3	0	7	6	∞
3	6	3	0	2	∞
4	4	1	1	0	∞
5	6	3	2	2	0

$D_5 =$

	1	2	3	4	5
1	0	4	4	3	∞
2	3	0	7	6	∞
3	6	3	0	2	∞
4	4	1	1	0	∞
5	6	3	3	2	0

- Time Complexity : $O(V^3)$
- Space Complexity : $O(V^2)$