Tutorial Sheet-06

1. A minimum spanning tree is a type of spanning tree where the cost is minimum (sum of the weights of all edges)

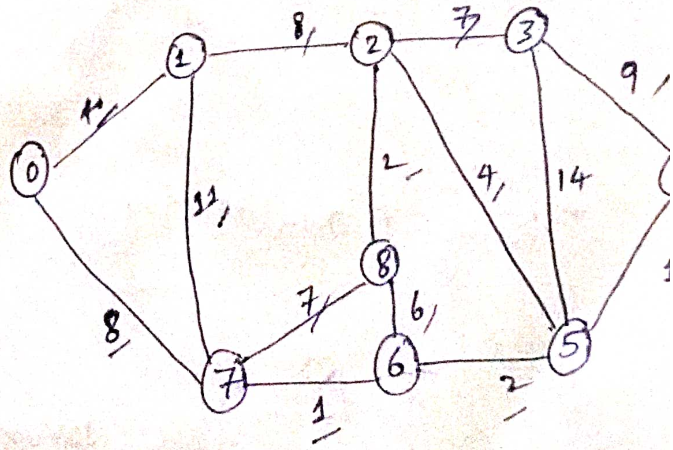
- Minimum spanning tree has applications in the designing of networks.
- Used in algorithms approximating the travelling salesman problem.
- Cluster analysis
- Image segmentation and handwriting recognition.

2.

Algorithms	Time Complexity	Space Complexity
Prim	$O((m+n)\log n)$	$O(V+E)$
Krusal	$O(n\log n)$	$O(E+V)$
Dijkstra	$O(n^2)$	$O(n^2)$
Bellman ford	$O(V.E)$	$O(n)$

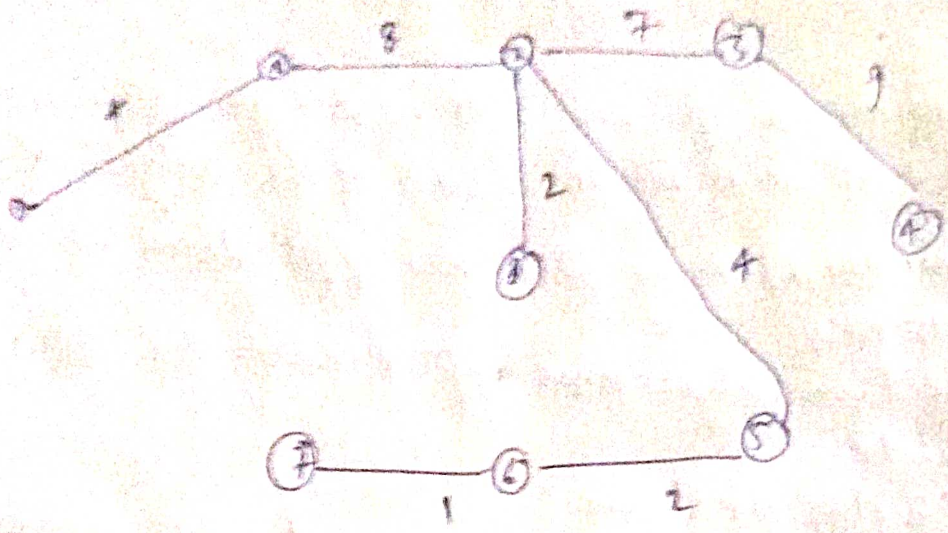
3.

(i) Applying Kruskal's Algo.



U	V	W
~~8~~	~~4~~ ~~8~~	~~4~~
~~0~~	~~7~~	~~8~~
7	6	1 ✓
6	5	2 ✓
8	2	2 ✓
0	1	4 ✓
5	2	4 ✓
8	6	6 ×
7	8	7 ×

2	3	7 ✓
1	2	8 ✓
0	~~4~~ 7	8 ×
3	4	9 ✓
5	4	10 ✓
7	1	11
5	3	14

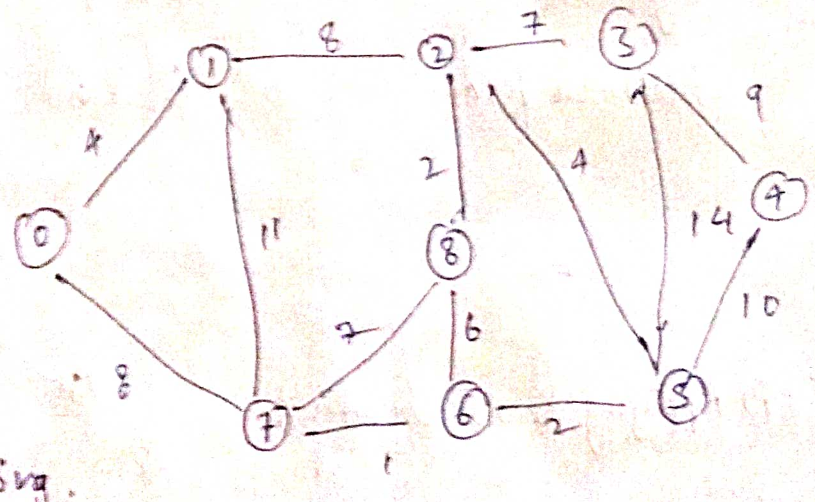


No. of edges = $(n-1) = 9-1 \Rightarrow 8$

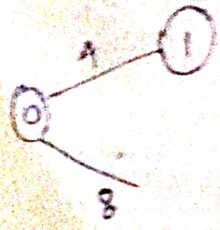
Min. Weight = $1+2+2+4+4+7+8+9$

$\Rightarrow 37$

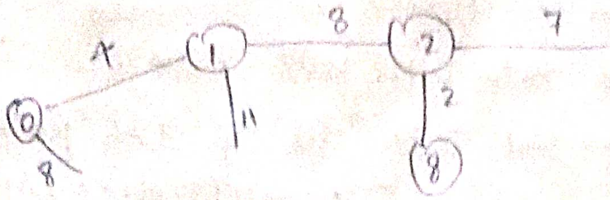
(ii) Applying Prims Algo

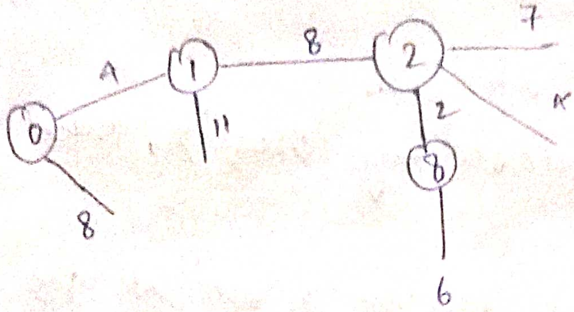


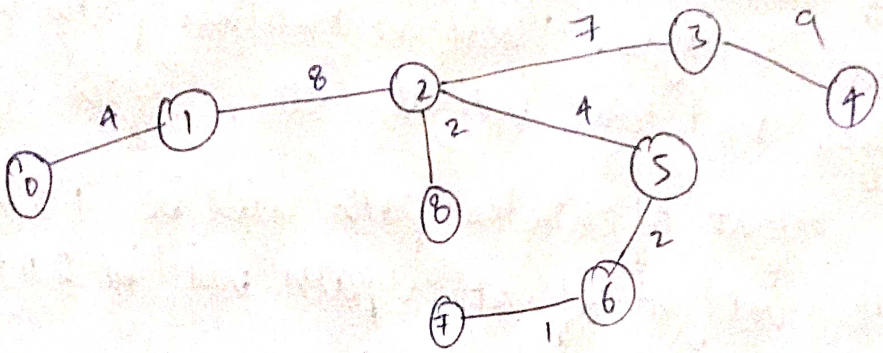
Taking 0 as starting.



taking less cost vertex every time.







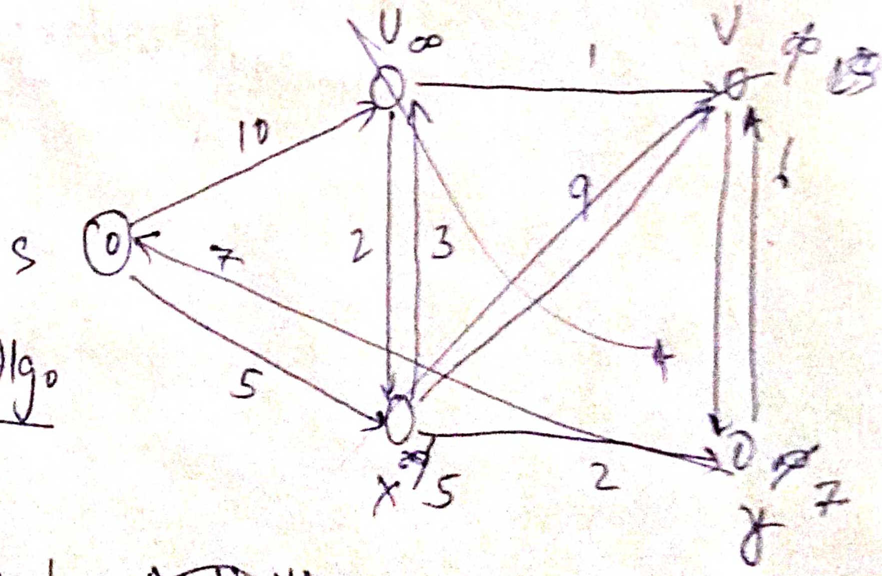
No. of edges = 9-1 = 8.

Min cost/weight = 37.

4. If the weights are ~~inter~~ multiplied by 10 units then the shortest path will not change.
The reason being is the weights of all paths from 's' to 't' will be a multiplied by 10. (same amount).

If the weights are increased by 10units, then shortest path may change
Eg→ if the shortest path is of weight 10 and have 5 edges. There is another path with 2 edges and total weight 20, After increment of 10
The weight of shortest path will be $10+50 \Rightarrow 60$.
while weight of other path will be $20+20 = 40$.
∴ The shortest path will change.

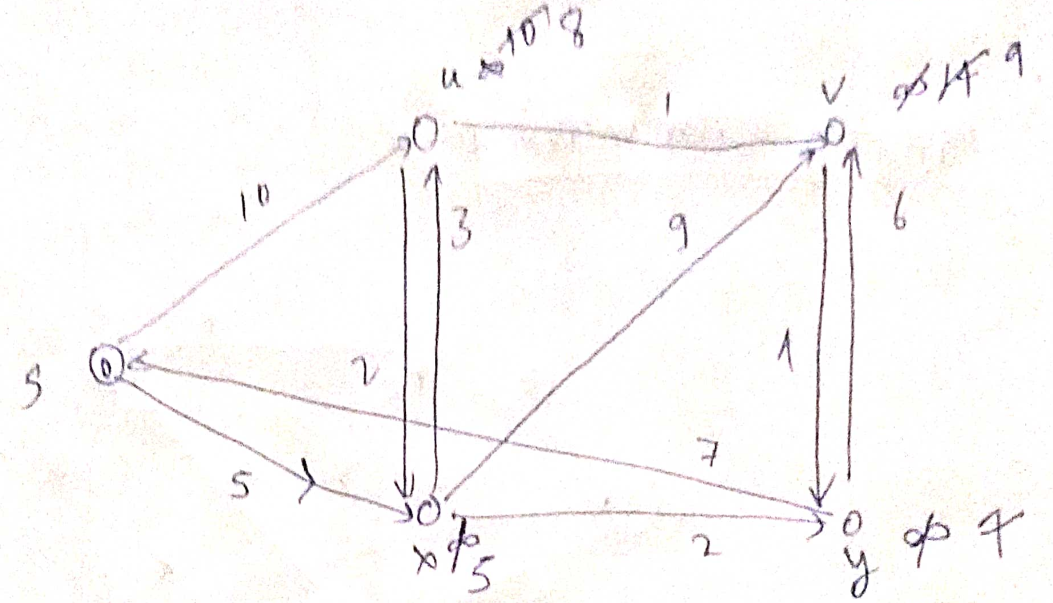
5.



Applying Dijkstra's Algo

Source Node S.

Source	Destination
S	X

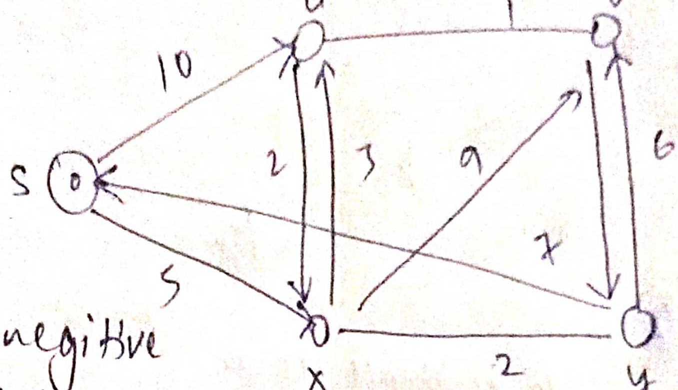


- Initialize other distance to ∞
- from source node go take shortest path.

S ⟶ X

node	Shortest Distance from Source Node S
S ⟶ X	5
S ⟶ X ⟶ u	8
S ⟶ X ⟶ y	7
S ⟶ X ⟶ u ⟶ v	9

Applying Bellmanford Algo



As there is no negitive ~~cycle~~ weight, there will be no negitive cycle.

0	~~∞~~ 5	~~∞~~ 7	~~∞~~ ~~14~~ ~~13~~ 9	~~∞~~ ~~10~~ 8
S	X	Y	V	U
S	X 5	Y 7	V 9	U 8

Shortest distance from

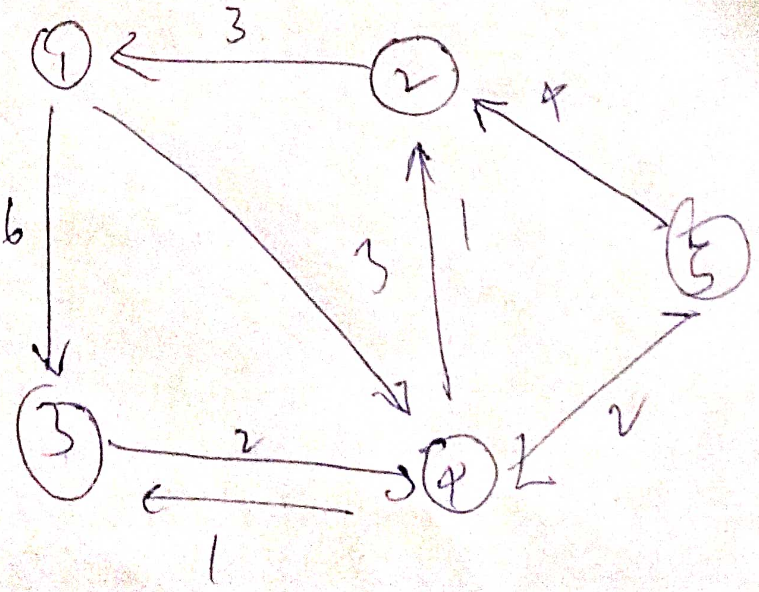
S → X = 5

S → Y = 7

S → V = 9

S → U = 8.

Applying Floyd Warshall Algo



$$A^0 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & \infty & 6 & 3 & \infty \\ 2 & 3 & 0 & \infty & \infty & \infty \\ 3 & \infty & \infty & 0 & 2 & \infty \\ 4 & \infty & 1 & 1 & 0 & \infty \\ 5 & \infty & 4 & \infty & 2 & 0 \end{array}$$

$$A^1 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & \infty & 6 & 3 & \infty \\ 2 & 3 & 0 & 9 & 6 & \infty \\ 3 & \infty & \infty & 0 & 2 & \infty \\ 4 & \infty & 1 & 1 & 0 & \infty \\ 5 & \infty & 4 & \infty & 2 & 0 \end{array}$$

$$A^2 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & \infty & 6 & 3 & \infty \\ 2 & 3 & 0 & 9 & 6 & \infty \\ 3 & \infty & \infty & 0 & 2 & \infty \\ 4 & 4 & 1 & 1 & 0 & \infty \\ 5 & 7 & 4 & 13 & 2 & 0 \end{array}$$

The time complexity will be $O(n^3)$ as 3 nested loops are used.

Space complexity $= O(n^2)$

$$A^3 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 9 & 6 & 3 & \infty \\ 2 & 3 & 0 & 9 & 6 & \infty \\ 3 & \infty & \infty & 0 & 2 & \infty \\ 4 & 4 & 1 & 1 & 0 & \infty \\ 5 & 7 & 4 & 13 & 2 & 0 \end{array}$$

$$\boxed{A^4 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 4 & 4 & 3 & \infty \\ 2 & 3 & 0 & 7 & 6 & \infty \\ 3 & 6 & 3 & 0 & 2 & \infty \\ 4 & 4 & 1 & 1 & 0 & \infty \\ 5 & 6 & 3 & 3 & 2 & 0 \end{array}}$$

Answer

Shortest distance Matrix