

Tutorial - 6

Name → Janvi Sah

Section → G

Roll No. → 62

Ans 1. Minimum spanning tree (MST) or minimum weight spanning tree is a subset of the edge of a connected edge-weighted undirected graph that connects all the vertices together, without any cycles & with minimum possible total edge weight.

Applications: →

- i) Design LAN
- ii) laying pipelines connecting offshore drilling sites, refineries and consume markets.

Ans 2.

Algorithm

Prim's Algorithm

Kruskal's Algo.

Dijkstra Algo.

Bellman ford Algo.

Time Complexity

$O((V+E)\log)$

$O(E(\log V))$

$O(V^2)$

$O(VE)$

Space Complexity

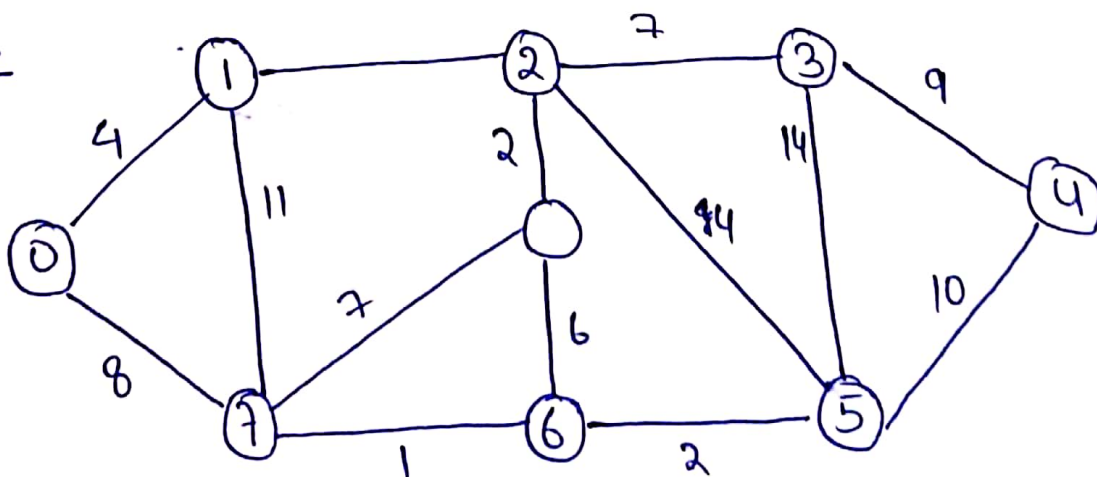
$O(V)$

$O(V^2)$

$O(V^2)$

$O(E)$

Ans 3.

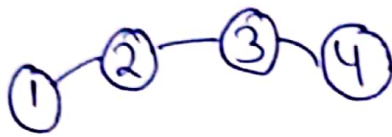


Depth First Search (DFS) can be used to detect cycle in Graph. DFS for a connected graph produces tree. For disconnected graph it produces forest.

Ans 5. Disjoint set data structure

It is also known as union-find data structure and merge find set. It is a data structure that contains a collection of disjoint or non-overlapping sets. The disjoint set means that when the set is partitioned into the disjoint subsets.

for example $\rightarrow S_1 = \{1, 2, 3, 4\}$ and $S_2 = \{5, 6, 7, 8\}$



No elements in common.

Operations

(i) Making new sets

The MakeSet operation adds a new element into a new set containing only the new element, and the new set is added to the data structure.

function MakeSet(x) is

if x is not already in the forest then

$x.parent = x$

$x.size = 1$ // if node store size

$x.rank = 0$ // if node store rank

end if

end function

(ii) Finding set representative

The find operation follows the chain of parent pointers from a specified query node x until it reaches a root element.

queue

node	G	D	F	H	C	E	A	B
parent	X	G	G	G	D	C	E	A

dequeue

	1	1	1	1	1	1	1	1
	G	D	F	H	C	E	A	

Traversal → **G D F H C E A B**

visited

G							
---	--	--	--	--	--	--	--

Stack

D	F	H					
---	---	---	--	--	--	--	--

visited

G	D	F					
---	---	---	--	--	--	--	--

Stack H C

visited

G	D	F	H	C			
---	---	---	---	---	--	--	--

Stack E

visited

G	D	F	H	C	E	A	
---	---	---	---	---	---	---	--

Stack B

(ii) Visited

G	D						
---	---	--	--	--	--	--	--

Stack

F	H	C					
---	---	---	--	--	--	--	--

(iv) Visited

G	D	F	H				
---	---	---	---	--	--	--	--

Stack C

vi) Visited

G	D	F	H	C	E		
---	---	---	---	---	---	--	--

Stack A

viii) Visited

G	D	F	H	C	E	A	B
---	---	---	---	---	---	---	---

Stack Empty.

