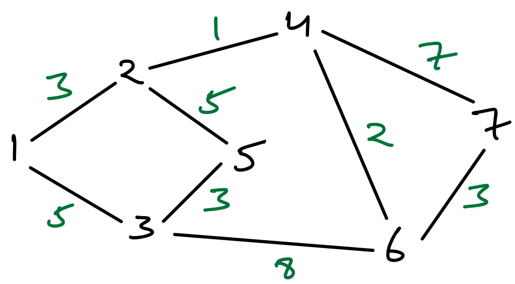
Welcome ☺

Agenda : Bridge construction
MST
Krusal's algo
Prim's algo
Djikstra's algo.

Q Given N islands & cost of construction of a bridge b/w multiple pair of islands. (cost > 0)
Find min. cost of construction req. s.t it is possible to travel from one island to any other island via bridges.
If not possible return -1

N = 7

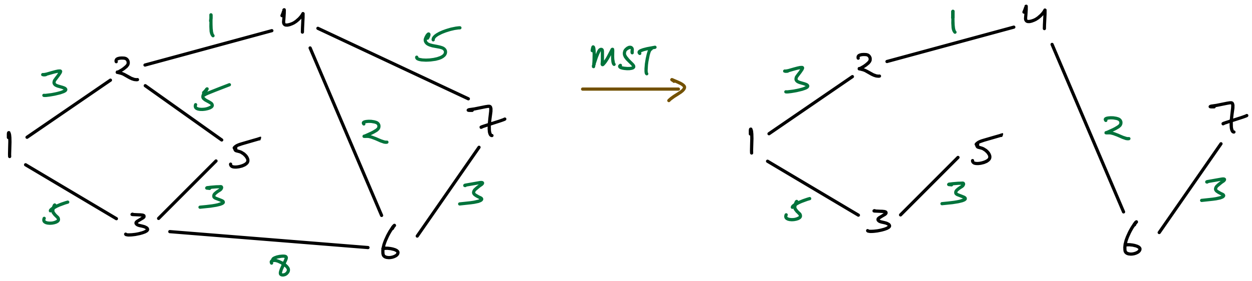
$1 \xrightarrow{3} 2$
$1 \xrightarrow{5} 3$
$2 \xrightarrow{1} 4$
$2 \xrightarrow{5} 5$
$3 \xrightarrow{3} 5$
$4 \xrightarrow{2} 6$
$3 \xrightarrow{8} 6$
$4 \xrightarrow{5} 7$
$6 \xrightarrow{3} 7$



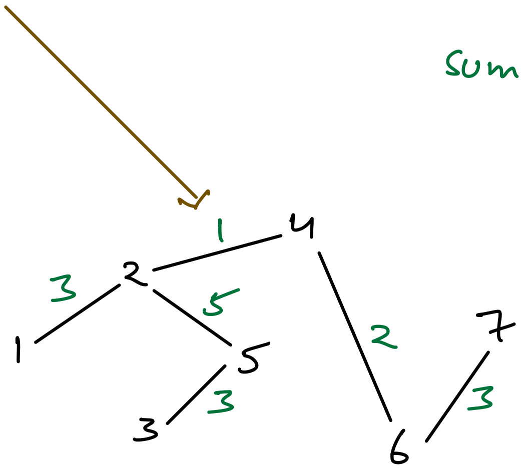
⇒ Ans = -1 → if graph is disconnected.
(check using DSU)

Q In a connected graph with N nodes, what is min # edges possible ?
Ans = N-1
(Trees)

Minimum Spanning Tree ⇒ Tree generated from a connected weighted graph s.t all nodes are connected & sum of weights of all selected edges is minimum.



sum of wt. = 17



sum of wt. = 17

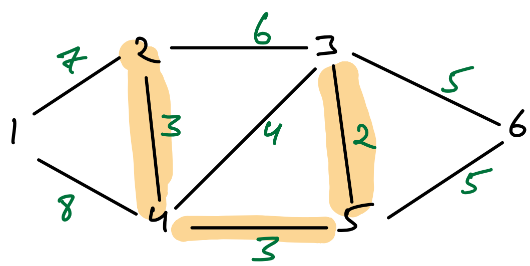
⇒ multiple MST possible for any graph.

⇒ Graph with unique weights ⇒ Unique MST.

Algo. to find MST → 1) Kruskal's algo. 2) Prim's algo. } greedy algo.

Kruskal's algo

⇒ Select edges with min weight if it is not forming a cycle, till the complete graph is connected.

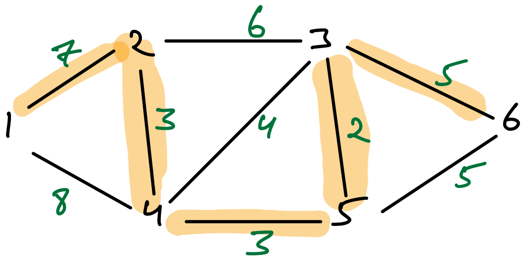


Steps

1) Sort edges w.r.t weight T.C ⇒ O(E log E)

2) Consider each node as a set i.e parent[i] = i (DSU) (visited array won't work)

3) Travel all edges (u, v) & take union.
If disjoint → add in ans



3 —2— 5 ✓
2 —3— 4 ✓
4 —3— 5 ✓
3 —— 4 ✗
3 —5— 6 ✓

5 —— 6 ✗
2 —— 3 ✗
1 —7— 2 ✓
1 —— 4 ✗

Ans = 2 + 3 + 3 + 5 + 7 = 20

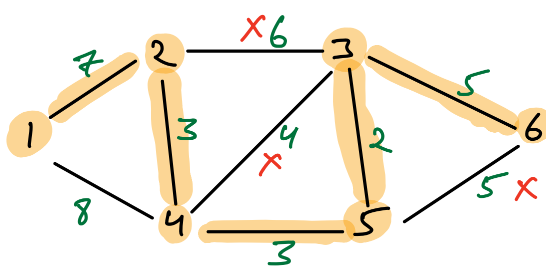
T.C → O(E log E + E)

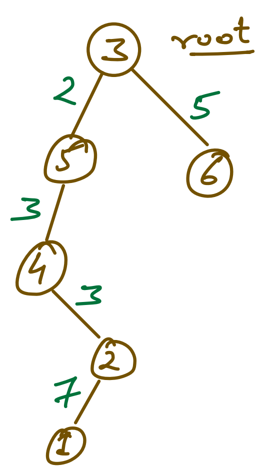
S.C → O(N)

Prim's algo

1) Start with any node as root of MST & keep adding the other nodes as its children.

2) Give priority to nodes that have less edge weight.





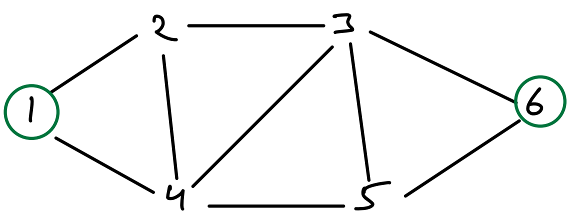
1) Start with root, insert its edges in a min. heap (w.r.t to weight)

2) Pick min. weight edge from heap → if it forms a cycle i.e connect both visited nodes, repeat step 2 else add the other node as part of tree & insert its connected edges in min. heap.

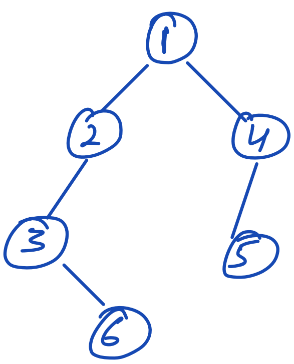
3) Continue step 2 till complete tree is formed.

T.C $O(E \log E)$ SC $O(N + E)$
$N \to$ vis[ ] $E \to$ heap

Q Find min # edges to travel from u to v in undirected simple graph.



u = 1 v = 6 ans = 3

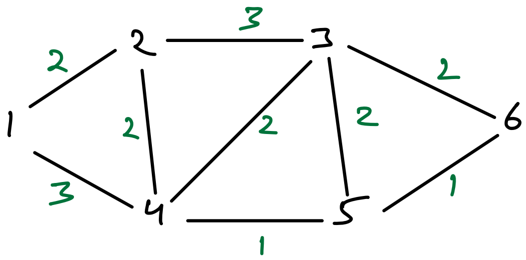


~~1~~ ~~2~~ ~~4~~ ~~3~~ ~~5~~ ~~6~~

ans ⇒ Start bfs from u & level of v is ans.

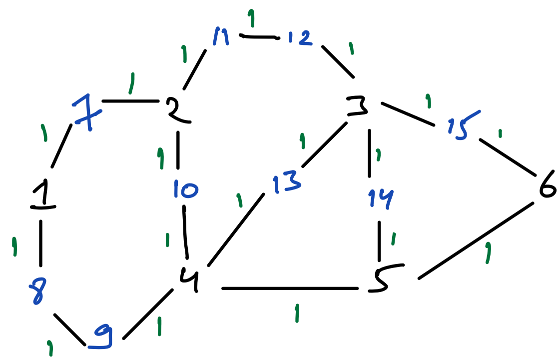
T.C → $O(N+E)$ S.C → $O(N)$

Q Find min distance to travel from u to v in undirected simple graph. $(1 \leq \underline{\underline{w.t}} \leq 3)$



$u = 1$ $\quad v = 6$ $\quad \underline{ans} = 5$

Dummy Nodes



$\Rightarrow$ Insert dummy nodes s.t weight = 1 $\forall$ edges.

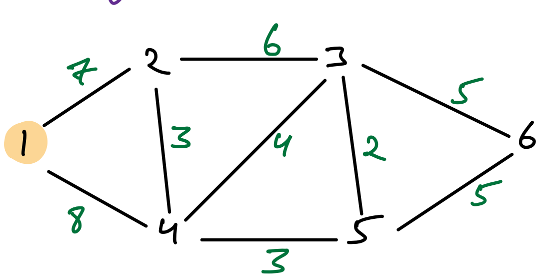
$\Rightarrow$ Then apply bfs.

$\underline{\underline{T.C}} \Rightarrow O(N+E)$

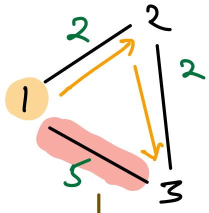
$\rightarrow$ It will not work for larger weights.

Dijkstra's algo $\Rightarrow$ Single source shortest path algo for weighted graph with +ve weights.

Q There are N cities in a country, you are living in city 1. Find min. distance to reach every city from city 1



$d[1] = 0$ (source)



Relaxing an edge.

$d[1] = 0$

$d[2] = 2$

$d[3] = 2+2 = 4$

$$\text{if } (d[u\_\_w] > d[u\_\_v] + d[v\_\_w])$$

$$d[u\_\_w] = d[u\_\_v] + d[v\_\_w]$$

To be continued ....