Spanning Tree

↳ Microsoft, Google, facebook

- ↳ Kruskal ✓
- ↳ Prim's ✓
- ↳ Dijkstra's

↳ Tuesday session

Logical concept
- How it works ✓
- Time complexity ✓

Spanning tree → subgraph of original tree

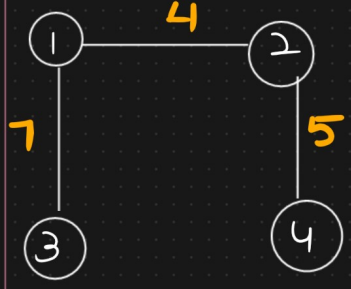
Properties

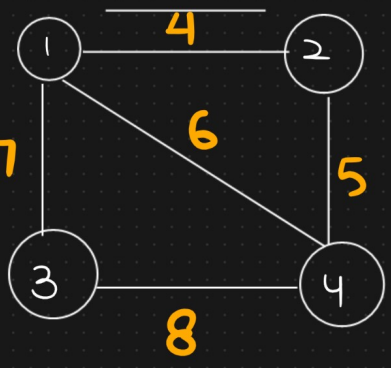
1) S contain all the vertices of graph G
2) S contain (n-1) edges → **connected**

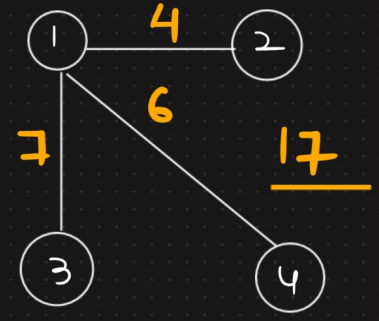
   n → # vertices

3) S cannot contains cycle

Example







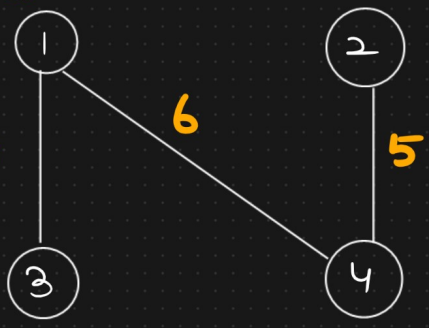
ST-1

ST-2 ✓

16

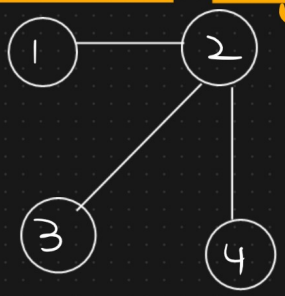
Minimum edge weight

ST-3



Minimum Spanning Tree 18

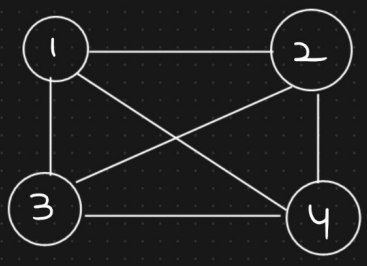
Invalid subgraph



ST-4

✗

Graph → complete graph



→ Possible number of spanning tree :-

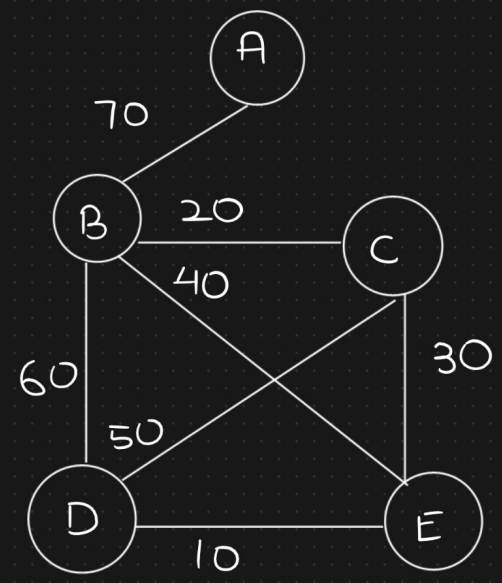
n^{n-2}

$\Rightarrow 4^{4-2} = 4^2 = 16$

Kruskal's Algorithm

↑ ~~Θ~~(E)

Minheap



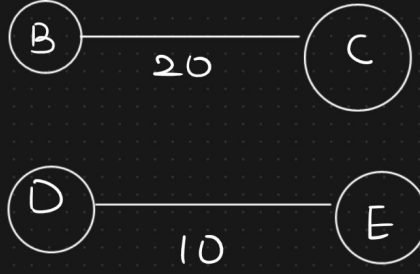
Edges = 7

logn ↓ Pop

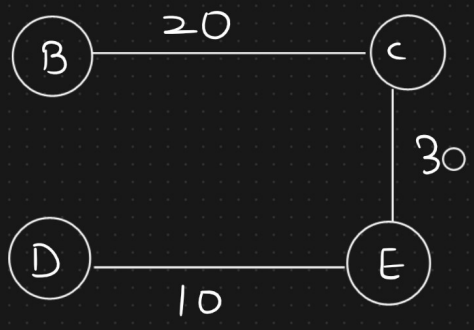
1) D — 10 — E

2) Is there a cycle or not

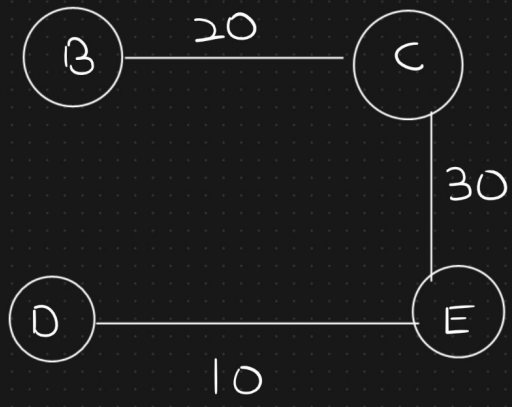
↳ add the edge in MST otherwise ignore & move forward



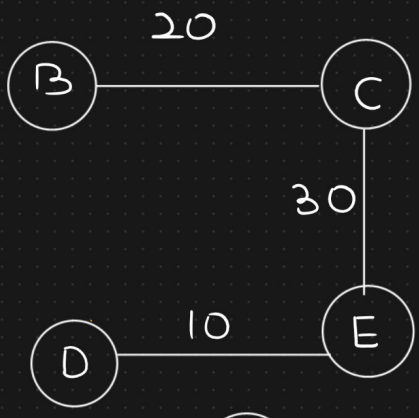
3)



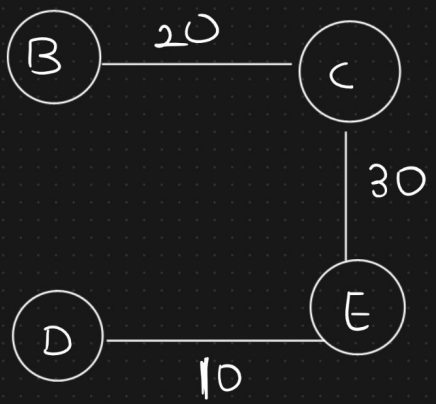
4)



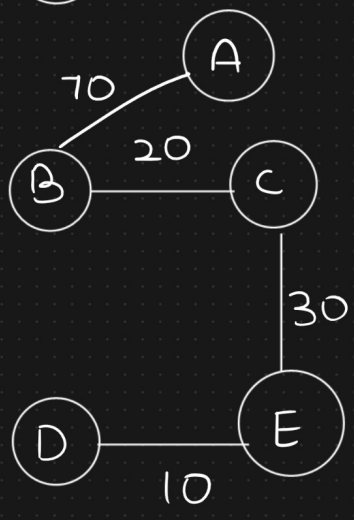
5)



6)



7)



↳ **Minimum Spanning Tree**

$70+20+30+10$

$= 130$

$m \approx v$

Worst case scenario → Pop element from minheap

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

$\Rightarrow \Theta(E \log V) = \Theta(E \log E)$

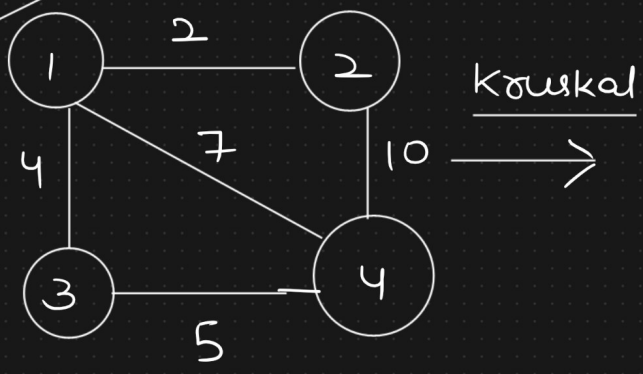
Best case scenario → Pop element from minheap

$\Rightarrow E + (V-1)(\log V) = \Theta(V \log V)$

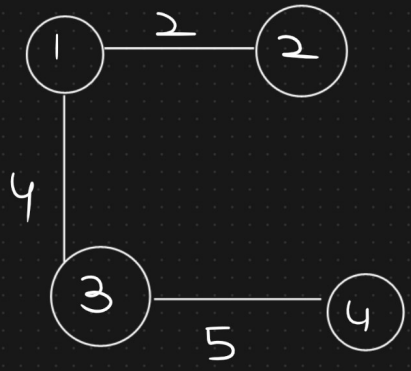
E → Build minheap

$(V-1)$ → Pop — 3 times

Best case



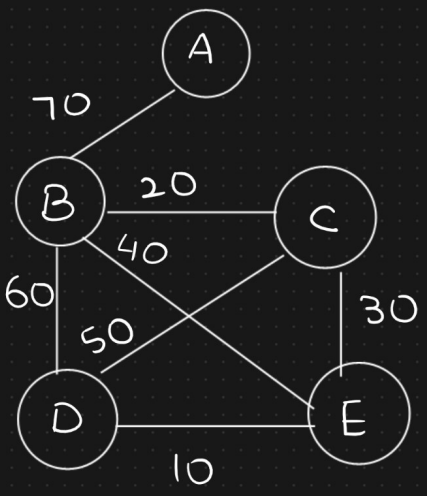
Kruskal →

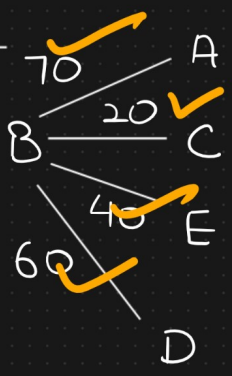


$\log V = \log E$

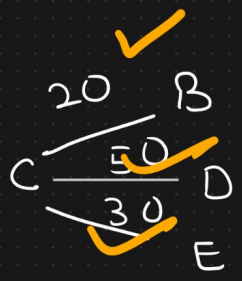
Prim's algorithm

(Adjacent for each vertex)

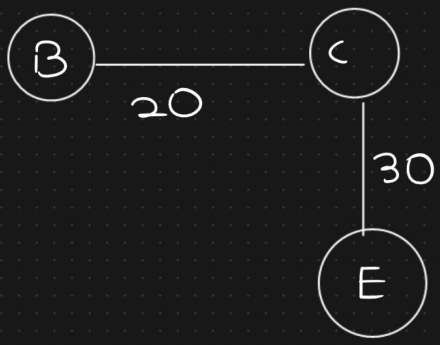




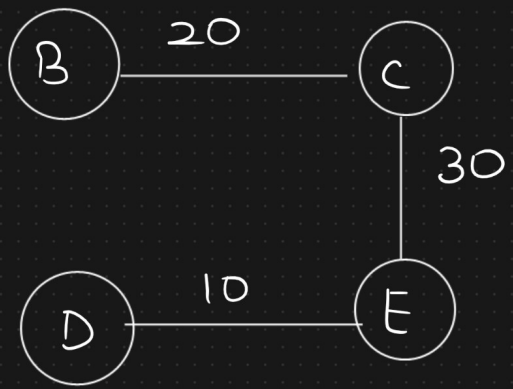
1)

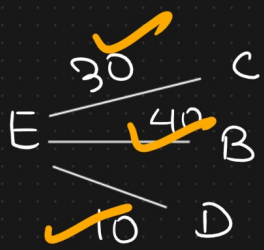


2)

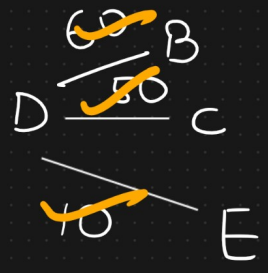


3)

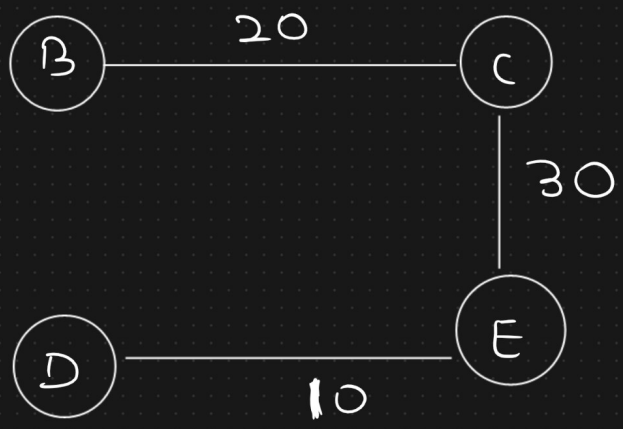




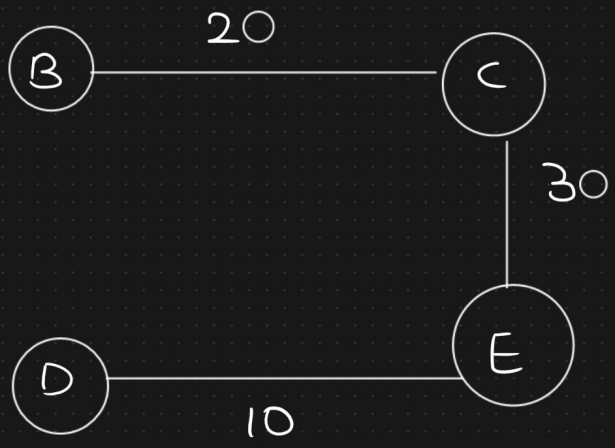
4)



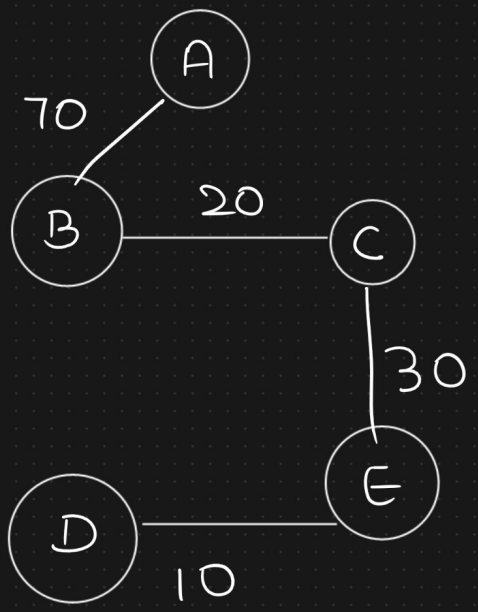
5



6

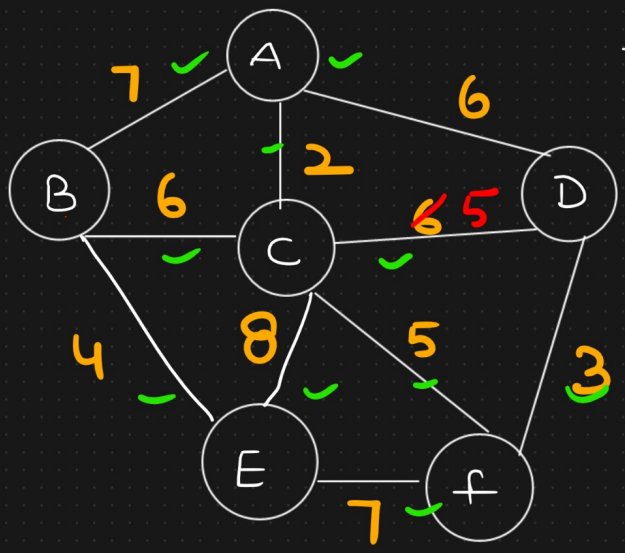


7



70 + 20 + 30 + 10

130



Time complexity

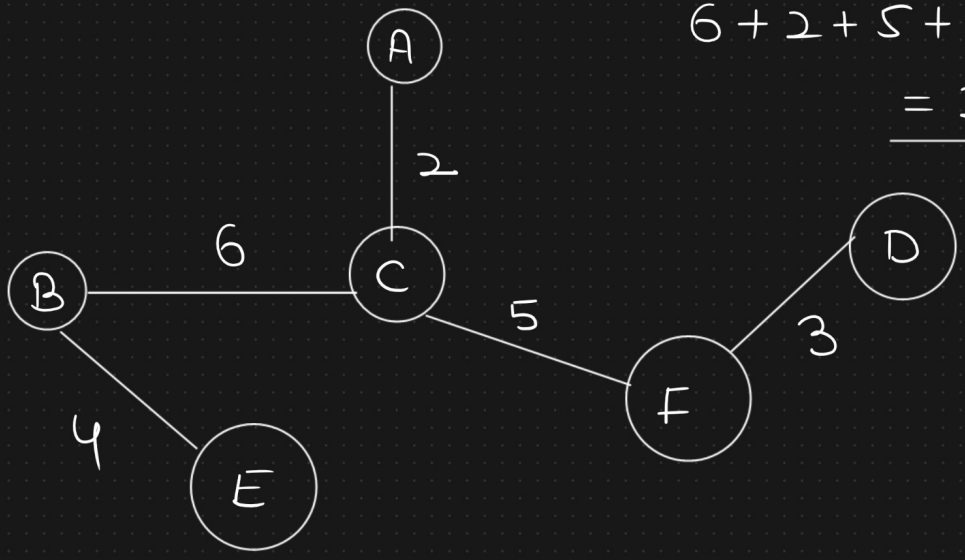
Build minheap

$V + V\log V +$ (Pop) $E\log V + 2E$

Adjacency list

	A	B	C	D	E	f	
Initialization	0 N	∞ N	∞ N	∞ N	∞ N	∞ N	
A		7 A	2 A	6 A	∞ N	∞ N	→ 3 log n + 3
C		6 C		6 A	8 C	5 C	3 log n + Degree(C)
f		6 C		3 E	7 f		2 log n + Degree(f)
D		6 C			7 f		0 + Degree(D)
B						4 B	log n + Degree(B)
E							

Minimum spanning Tree



$6+2+5+3+4$

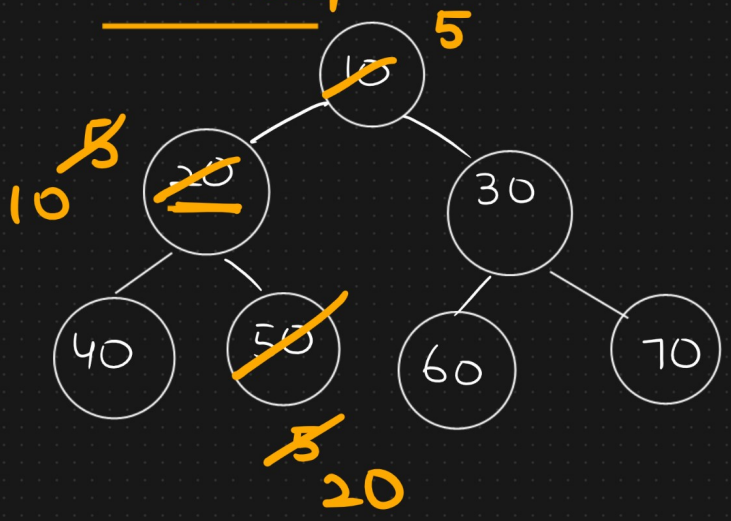
$= 20$

Heap Data structure

complete Binary Tree ↑

↳ Decrease key operation

Minheap



$\log_2 n$

$\Theta(\log n)$

Time complexity

Adjacency List

$$V + V \log V + E \log V + 2E$$

Pop

Build minheap

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

$$\log E = \log V$$

Adjacency Matrix

$$V + V \log V + E \log V + V^2$$

Degree

$$\Rightarrow \Theta(V^2)$$

$n \log n < n^2$