Tuesday, April 9, 2024 10:57 AM

Minimum Spanning Trees (MST)

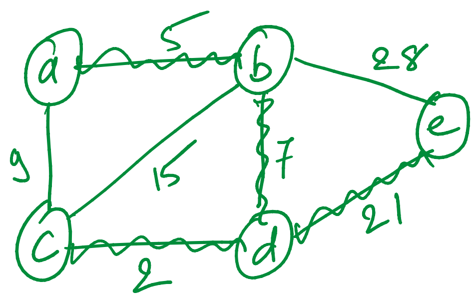
Given undirected connected graph

$G = (V, E)$, $w: E \to \mathbb{R}^+$

Find a (connected subgraph that covers all the vertices) minimizes the total weight of the selected edges

spanning all vertices

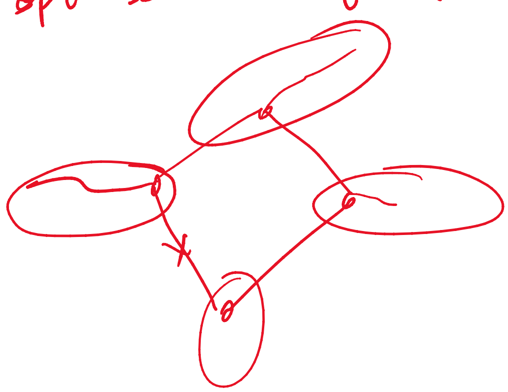
e.g.



Possible sol'n
$5 + 15 + 2 + 28 = 50$

Better sol'n
$2 + 7 + 5 + 21 = 35$
Best

Obs1: opt sol'n acyclic?



Removing any one edge from the cycle will still keep the graph connected.

$\Downarrow$

opt sol'n is acyclic, undirected

$\equiv$

Tree

Obs2: A tree w/ n nodes

Obs 2: A tree w/ n nodes has (n-1) edges.

Idea 0: Try all possible trees & pick w/ min weight. (exponential!)

Idea 1: Greedy.

Runtime

$O(m^2)$

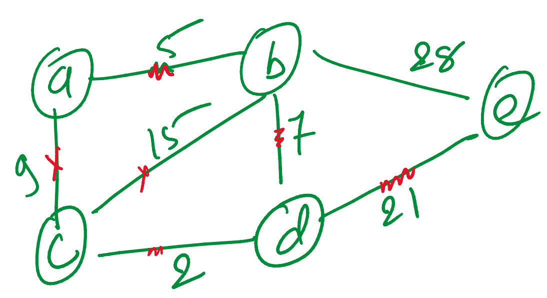
☆ Kruskal's Alg'm (1956):

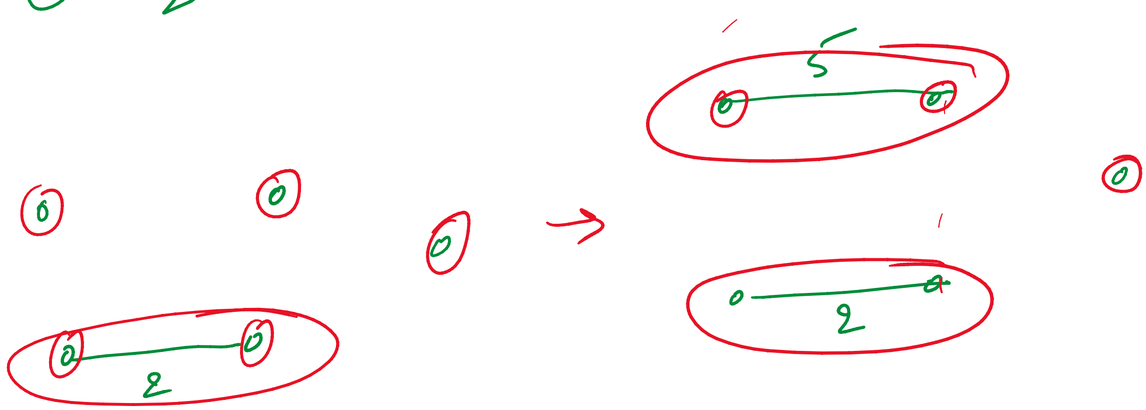
1. $T = \emptyset$
2. repeat {
3. pick edge e w/ smallest weight that hasn't been considered yet.
4. If $T \cup \{e\}$ does not have a cycle
5. Insert e in T $(T = T \cup \{e\})$

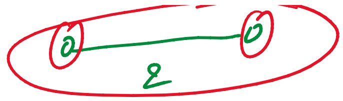
} until all vertices are connected.

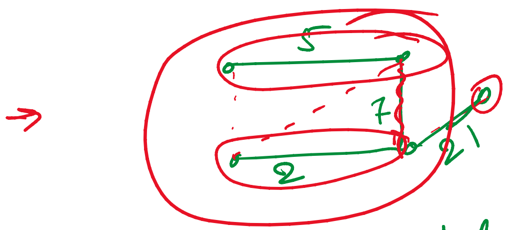
$O(m)$ (lines 3) $O(n)$ (lines 4–5)

e.g.



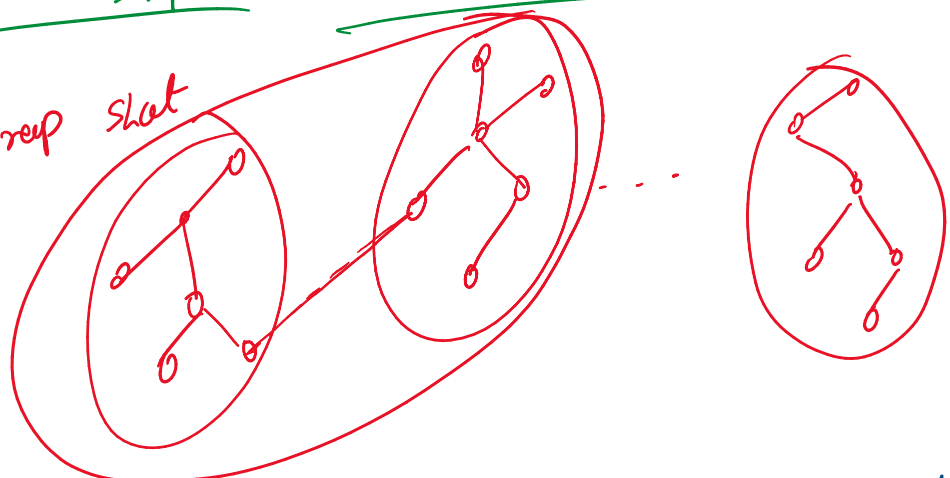






★ Better Implementation using Union Find Data structure

Snap shot



$m \lg m$ 1. Sort edges in decreasing order of weights.

2. Create $\{v\}$, $v \in V$

3. for each edge (u,v) in the order. {

4. if u & v are in two disjoint "sets"

5. output (u,v) or add (u,v) to tree . Union the sets of u & v.

}

Runtime :

line 4: $O(\alpha(n))$ inverse Ackermann Func $\qquad \alpha(n) << \log\log\log \dots \log n.$

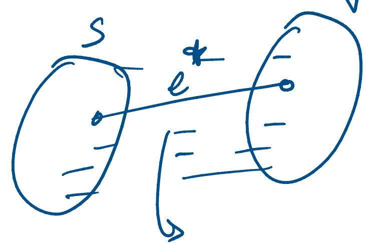
line 5: $O(1)$

Total time : $O(m \lg m + m\,\alpha(n)) = \underbrace{O(m \lg m)}_{(m \le n^2)} = O(m \lg n)$

Total time : O(...) $(m \le n^2)$

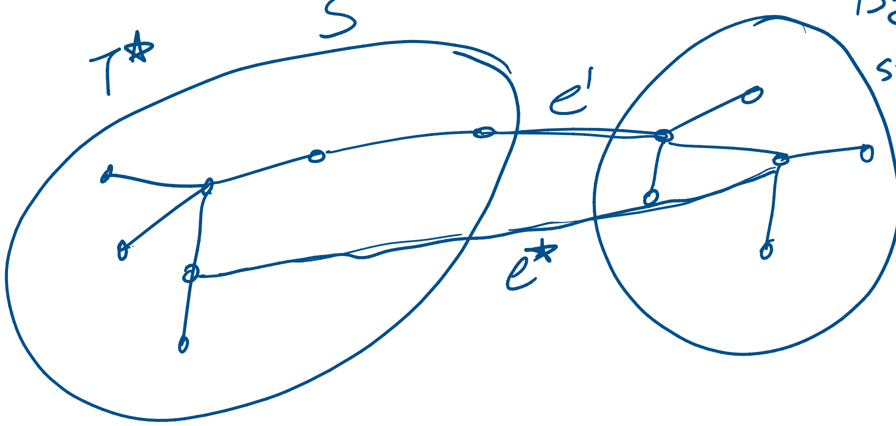
★ Correctness Pf : T^* is MST, (unknown to me) (Assume all weights are different)

Key lemma : For any $S \subseteq V$, & e^* be min weight edge among all edges crossing S to $V \setminus S$

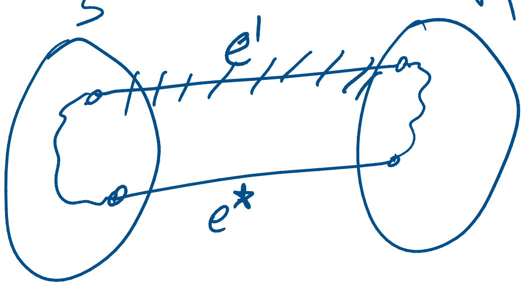


$$\boxed{\Longrightarrow \quad e^* \in T^*}$$

Pf : By contradiction suppose $e^* \notin T^*$



Consider $\underline{T^* \cup \{e^*\}}$. It must have a cycle passing through e^*.



$\underline{w(e^*) < w(e')}$

(exchange argument)

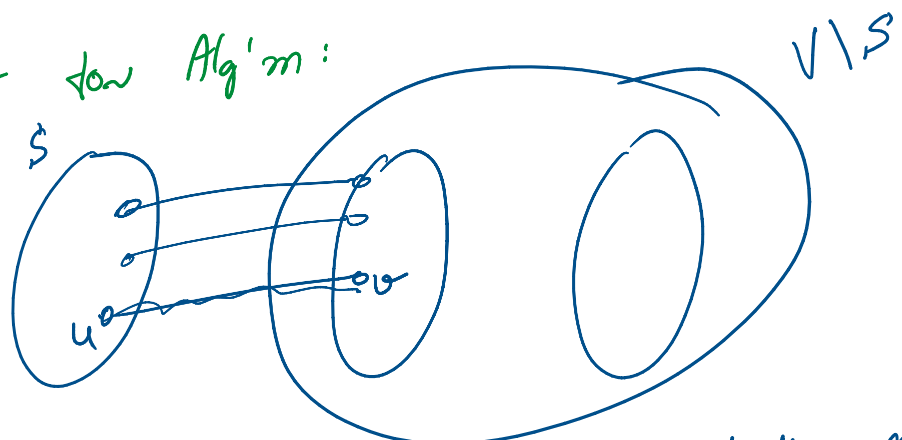
$\tilde{T} = T^* \cup \{e^*\} \setminus \{e'\}$ is also a spanning tree

$\tilde{T} = T^* \cup \{e^*\} \setminus \{e'\}$

$$W(\tilde{T}) = W(T^*) + W(e^*) - W(e')$$
$$< W(T^*) \qquad (\because W(e^*) - W(e') < 0)$$

Contradiction to T^* being MST. ∎

Connected pf for Alg'm:



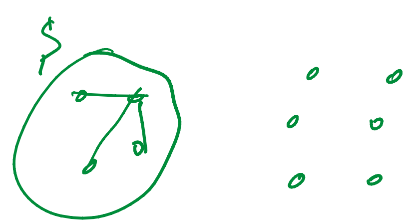
Let (u,v) be an edge added at the current iteration.

S = be the component/set of u.

then (u,v) is the min weight crossing edge for set S.

$\Rightarrow (u,v) \in T^*$ (in MST) by the key lemma ∎

Prim's Algm (1957): (Dijkstra like)



1. $S = \{s\}$, $T = \emptyset$
2. while $S \neq V$ {

pick an edge $(u,v) \in E$, $u \in S$, $v \notin S$

2. while

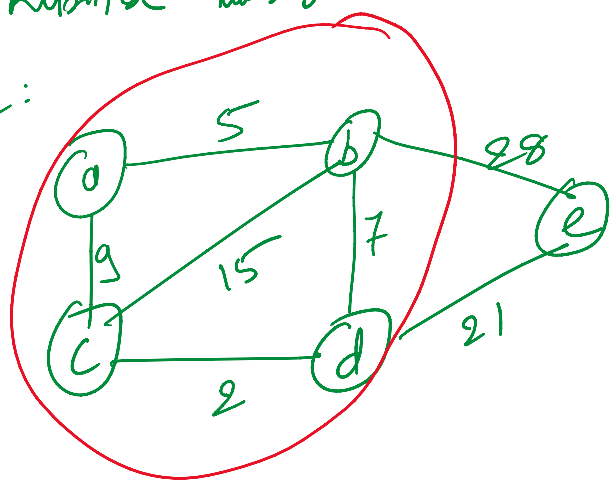
3. Pick an edge $(u,v) \in E$, $u \in S'$, $v \notin S'$ with minimum weight.

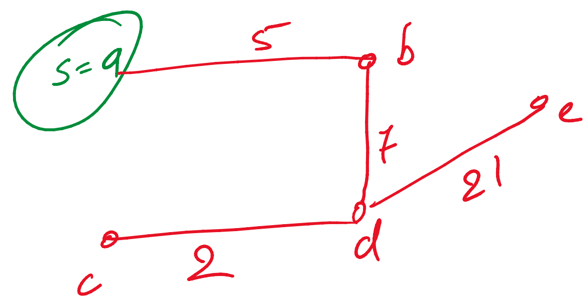
4. Add (u,v) To T & $v \in S'$
$(T = T \cup \{(u,v)\})$ $(S' = S' \cup \{v\})$

}

Runtime using Fibonacci heap in $O(n \log n + m)$

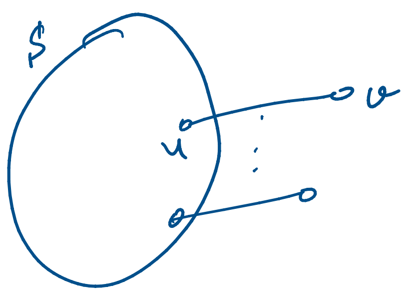
eg:





★ Correctness pf: (Just by the key lemma).

In any iteration of the Alg'm. let S be the current set of connected vertices.



We picked (u,v), $u \in S$, $v \notin S$ w/ min weight

$\Rightarrow (u,v)$ is the min weight crossing edge for set S.

$\Rightarrow (u,v) \in MST$ by the key lemma ∎

History :

Boruvka (1926) $O(m \log n)$

K : (1956) $O(m \log n)$ $(m \sim O(n^2))$

P : (1957) $O(n \log n)$

⋮

Yao (1975) $O(m \log \log n)$

Fredman, Tarjan (1985) $O(m \log^* n)$

Karger, Klein, Tarjan (1994) $O(m)$ Randomized.

Chazelle (1997) $O(m\, \alpha(n))$ Det.

Pettie & Ramachandran (2001) $O(T^*(m,n))$ Det.

$\leq O(m\, \alpha(n))$

$O(m)$ OPEN! det