BELLMAN-FORD CONTINUED

MINIMUM SPANNING TREE

Bellman - Ford

dist = array indexed by V initialized to ∞

prec = array indexed by V initialized to $\perp$

```
dist[S] = 0
for l = 1 to |V|-1 :
    for each v in V :
        for each edge (u,v) :
            if dist[u] + l(u,v) < dist[v] :
                dist[v] = dist[u] + l(u,v)
                prec[v] = u
```

Running time $O(|V| \cdot |E|)$

Correctness : At step ℓ of outer for

for every v $\quad$ $dist[v] \leq$ length of shortest path from s to v that uses $\leq \ell$ edges

Bellman-Ford

dist = array indexed by V initialized to ∞

prec = array indexed by V initialized to $\perp$

```
def update (u, v)
    if dist[u] + ℓ(u,v) < dist[v]:
        dist[v] = dist[u] + ℓ(u,v)
        prec[v] = u
```

```
dist[s] = 0
for t = 1 to |V| - 1:
    for each (u,v) in E:
        update (u,v)
```

For every $t = 1, \ldots$
after t iterations of outer "for" loop
$dist[v] \leq$ length of shortest path from s to v that uses $\leq t$ edges

---

$t = 0$

$dist[s] = 0$

$dist[v] = \infty$ for $v \neq 0$ ✓

Assume true up to t

consider $dist[\,]$ after $t+1$ executions of "for" loop

$\forall v$ $\quad$ $\forall$ path P from s to v that uses $\leq t+1$ edges

want to prove: $dist[v] \leq$ length P

$$s \to v_1 \to v_2 \to \cdots \to v_t \to v$$

consider path P from s to v with $= t+1$ edges

$$\text{length of } P = \underbrace{\ell(s, v_1) + \ell(v_1, v_2) + \cdots + \ell(v_{t-1}, v_t)}_{dist[v_t] \leq} + \ell(v_t, v)$$

$\forall v$ $\quad$ $\forall$ path P from s to v that uses $\le t+1$ edges

want to prove: $\text{dist}[v] \le \text{length } P$

$$s \to v_1 \to v_2 \to \cdots \to v_t \to v$$

consider path P from s to v with $= t+1$ edges

$$\text{length of } P = \underbrace{\ell(s, v_1) + \ell(v_1, v_2) + \cdots + \ell(v_{t-1}, v_t)} + \ell(v_t, v)$$

$\text{dist}[v_t] \le$

at end of iteration t

After $\text{update}(v_t, v)$ in iteration $t+1$

$$\text{dist}[v] \le \text{dist}[v_t] + \ell(v_t, v) \le \text{length } P$$

At end of iteration $t+1$

$$\text{dist}[v] \le \text{length } P$$

$$S \longrightarrow A \longrightarrow B \longrightarrow C \longrightarrow v$$

After 0 iterations

$dist[s] = 0$

After 1 iteration

consider step in which update (s, A)
in 1st iteration

$dist[A] \le \ell(s, A)$ ← remains true in all subsequent steps

After 2nd iteration

consider step in which update (A, B)

$$\begin{aligned} dist[B] &\le dist[A] + \ell(A, B) \\ &\le \ell(s, A) + \ell(A, B) \end{aligned}$$

After 3rd iteration

- - - update (B, C)

$$\begin{aligned} dist[C] &\le dist[B] + \ell(B, C) \\ &\le \ell(s, A) + \ell(A, B) + \ell(B, C) \end{aligned}$$

After 4th iteration

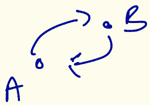
$$\begin{aligned} dist[v] \le\ & \ell(s, A) + \ell(A, B) \\ &+ \ell(B, C) + \ell(C, v) \end{aligned}$$

Tree

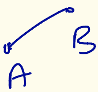
Graph undirected, connected, acyclic

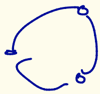
Note: undirected a cycle is of length ≥ 3

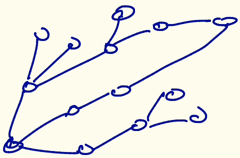
directed



undirected



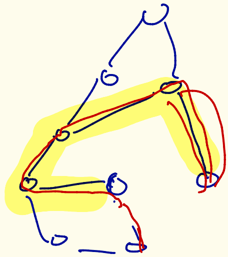




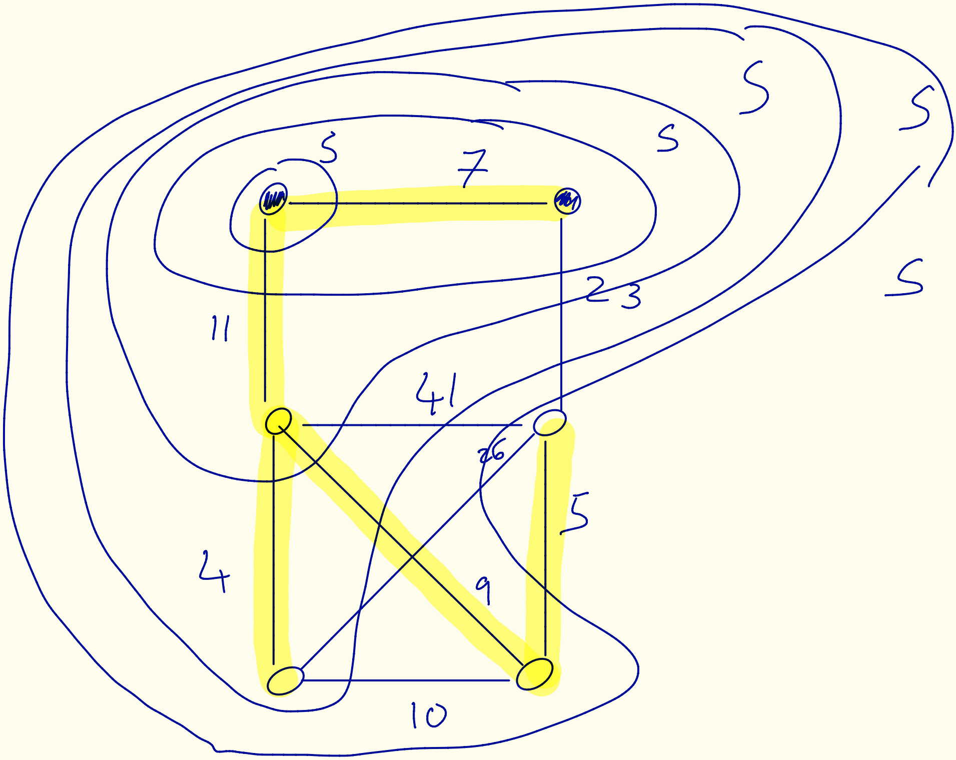
Tree on n vertices has $= n-1$ edges

A graph is a tree $\Leftrightarrow$ it is connected and $|V|-1$ edges

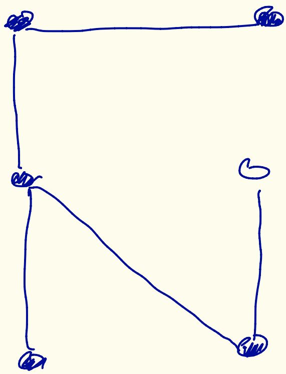
Suppose G connected and has a cycle



Then we can remove any one edge from cycle without compromising connectivity

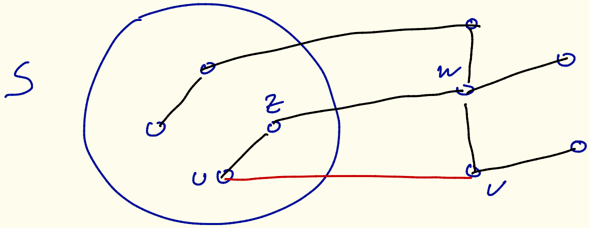


S
7
S
S
S
S
11
23
41
26
5
4
9
10



Thm: Suppose G is connected, all edge costs are different, $S \subseteq V$, $1 \leq |S| \leq |V|-1$, then cheapest edge out of S must belong to all minimum spanning trees

Proof



Let S be a set of vertices
(u,v) be cheapest edge out of S

Suppose T is an optimal tree that does not use (u,v)

- Add (u,v) to T
  We create a cycle
  Let (z,w) be a T-edge in the cycle that crosses from S to $V-S$
- Take out z,w

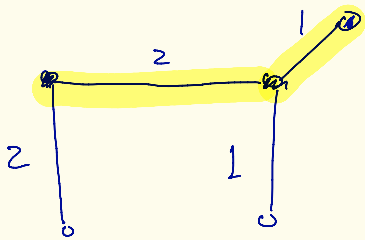
We a new tree $(T \cup \{(u,v)\} - \{(z,w)\}$
of cost $\text{cost}(T) + \text{cost}(u,v) - \text{cost}(z,w) < \text{cost}(T)$
T is not optimal

G undirected graph

S subset of vertices

(u,v) a cheapest edge out of S

Then there is an optimal tree that contains (u,v)



Thm

G connected weighted graph

S subset of vertices

F set of edges of G that

- don't create any cycle
- don't cross from S to $V-S$

(u,v) a cheapest edge from S to $V-S$

Then size of minimum spanning tree of G including all edges of F

=

size of minimum spanning tree of G including all edges of F and also (u,v)