Oct 23

Shortest Path Problem

Input: Directed graph $G=(V,E)$, $s \in V$ (source), $\forall e \in E$, $l_e \geq 0$ (integer)
"length" of e

Output: $\forall t \in V$, output a shortest s-t path (shortest length)

$$l(P) = \sum_{e \in P} l_e$$

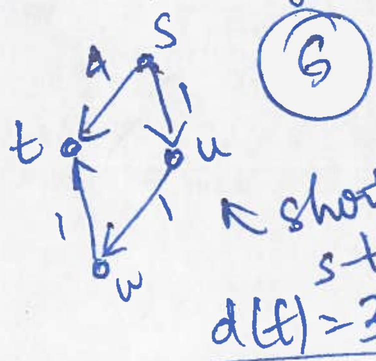
[Simpler: only output $d(t)$ $\forall t \in V$
length of a shortest s-t path]

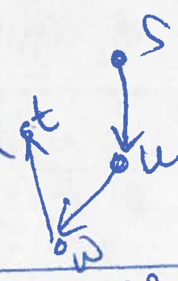
Recall: Q2 on mid-term 2

Special Case: $l_e = 1$ $\forall e \in E$ $\Rightarrow$ run algo for HW4 Q1.
[run BFS from s, layer # of t is $d(t)$]

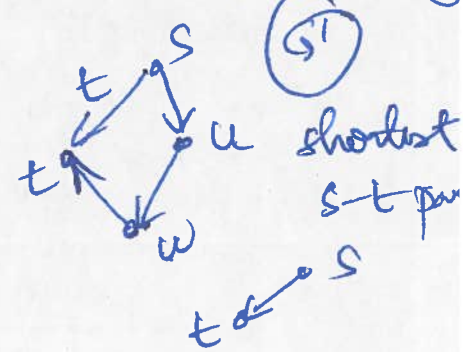
General case: $l_e \geq 0$ $\forall e \in E$ [Idea: Try to reduce to $l_e = 1$ $\forall e \in E$ case]

Idea 1: Ignore l_e i.e. assume $l_e = 1$ $\forall e \in E$ & run algo from HW4 Q1.

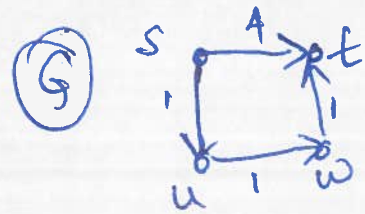




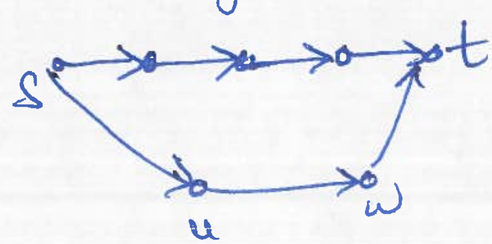
Run BFS $\longrightarrow$ ignoring l_e



Idea: (2(c) on the mid-term) Replace each edge e with a path with l_e edges in it to get G'

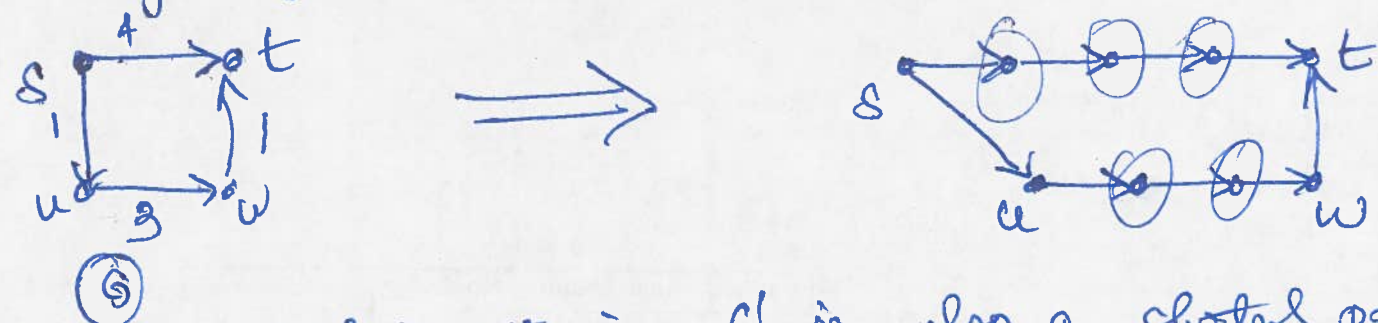


$\Longrightarrow$



$l_e = 1$ $\forall e$

General reduction: Replace each edge e w/ a path ~~of G~~ with l_e ~~set~~ edges. Important: each edge gets its own "private"/dummy node



Claim 1: Shortest path in G' is also a shortest path in G

$\Rightarrow$ then run algo from HW4 Q1 on G'.

Correctness: Claim 1 + HW4Q1.

Runtime analysis $l_{max} = \max_{e \in E} l_e$

(on Q2: $l_{max} = W$)

Claim 2: n' & m' be # of new nodes & edges in G',

$n', m' = O(l_{max}(m+n))$

HW4Q1 algo on G' runs in time $O(n'+m') = O(l_{max}(n+m))$ ∎.

Q: What is the input size?

Since any l_e can be represented in $O(\log l_{max})$ bits

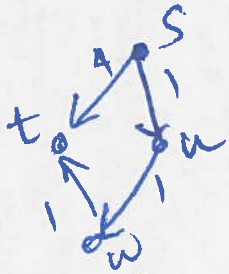
$\Rightarrow$ i/p size $O((\log l_{max})(m+n))$ ↖ $l_{max} = n^{100}$

Q: How do we avoid counting in unary?

Reminder: RAM model: basic operations on $O(\log n)$ bit registers is $O(1)$ time

ASSUME: l_{max} is $n^{O(1)}$.

Towards Dijkstra's algo

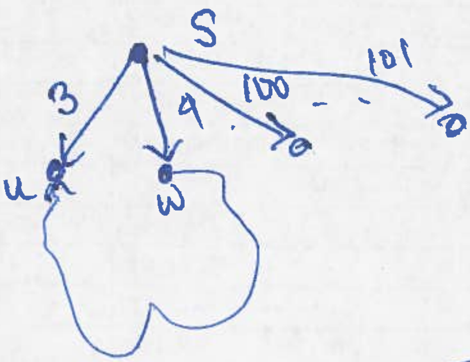


(s,t) is not shortest s-t path

BUT

(s,u) is shortest s-t path.

Claim: Pick u s.t. $\ell(s,u)$ is minimized & set $d(u) = \ell_{(s,u)}$



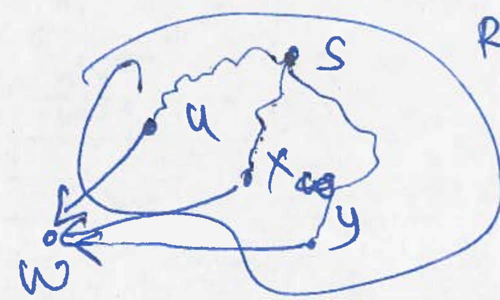
Consider any s-u path P

$\ell(P) = 4 + \underbrace{\text{stuff}}_{\geq 0 \text{ as } l_e \geq 0\ \forall e}$

≥ 4

$> 3 = \ell(s,u)$.

More generally:



Have: R

Want: To include a new vertex w in R

Maintain:

$d'(w) \rightsquigarrow$ an upper on $d(w)$

$$d'(w) = \min\{d(u) + \ell(u,w),\ d(x) + \ell(x,w),\ d(y) + \ell(y,w)\}$$

Pick w (to include in R) that min

$$d'(w) = \min_{\substack{u \in R \\ (u,w) \in E}} \{d(u) + \ell(u,w)\}$$