$\boxed{\text{Midterm 2 next Monday Nov 10th}}$

Let $G=(V,E)$ directed. Edge weights $w: E \to \mathbb{R}$. Source vertex $s \in V$. No negative weight cycles.

$dist_{\leq i}(v)$: length of shortest (s,v)-path with <u>at most</u> i edges.

$dist(v) = dist_{\leq |V|-1}(v)$.

$G' = (V', E')$. $V' = \{(v,i) \mid v \in V \text{ and } 0 \leq i \leq |V|-1\}$

$E' = \{(v, i-1) \to (v,i) \mid v \in V + 1 \leq i \leq |V|-1\}$

$\cup \{(u, i-1) \to (v,i) \mid u \to v \in E + 1 \leq i \leq |V|-1\}$

$w': E \to \mathbb{R}$. $w'((v,i-1) \to (v,i)) = 0$

$w'((u,i-1) \to (v,i)) = w(u \to v)$

Want shortest paths from $(s,0)$ to all $(v,|V|-1)$.

G' is a dag, so call DagSSSP$((s,0))$.

$|V'| = |V|^2$

$|E'| = O(VE)$ if G is connected

$O(V'+E') = O(V^2+VE) = O(VE)$

DagSSSP ↑

↑ if G is connected

All pairs shortest paths (APSP): Given directed $G=(V,E)$, $w: E \to \mathbb{R}$. (no negative cycles)

want $dist(u,v)$: distance from u to v

for <u>all</u> $u, v \in V$.

$O(V^3)$

```
ObviousAPSP(V, E, w):
    for every vertex s
        dist[s, ·] ← SSSP(V, E, w, s)
```

Acyclic: Dag SSSP in $V \cdot O(E) = O(VE)$

Unweighted: BFS in $O(VE)$

Non-negative weights: $O(V^3 \log V)$

Dijkstra: $O(VE \log V)$

o.w.: Bellman-Ford: $O(V^2 E)$

$O(V^4)$

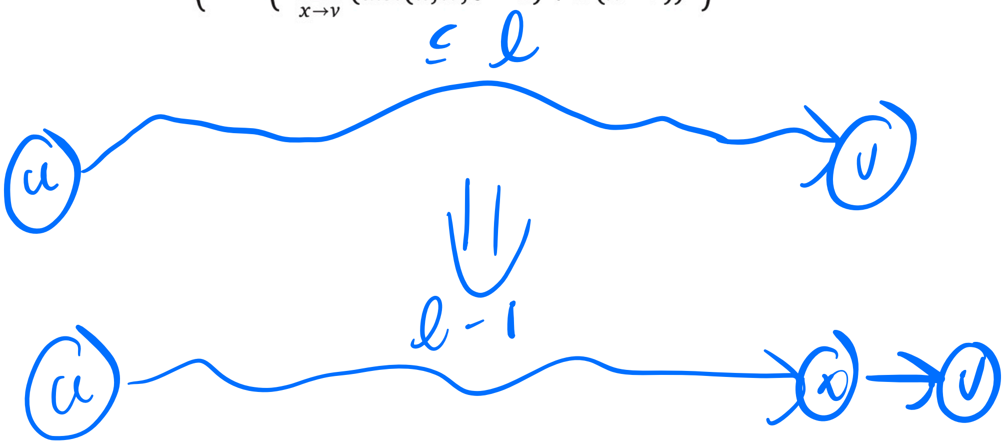
$$\cancel{dist(u,v) = \begin{cases} 0 & \text{if } u=v \\ \min\limits_{x\to v}\left(dist(u,x) + w(x\to v)\right) & \text{o.w.} \end{cases}}$$

cycles!

$dist(u,v,\ell)$: length of the shortest path from u to v with at most ℓ edges.

$(dist(u,v) = dist(u,v,|V|-1))$.

$$dist(u,v,\ell) = \begin{cases} 0 & \text{if } \ell = 0 \text{ and } u = v \\ \infty & \text{if } \ell = 0 \text{ and } u \neq v \\ \min \left\{ \begin{array}{c} dist(u,v,\ell-1) \\ \min\limits_{x\to v}\left(dist(u,x,\ell-1) + w(x\to v)\right) \end{array} \right\} & \text{otherwise} \end{cases}$$

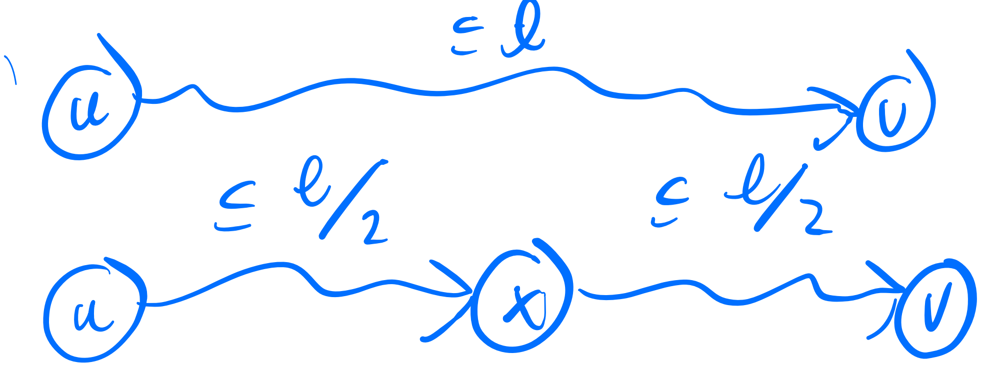



```
SHIMBELAPSP(V, E, w):
  for all vertices u
      for all vertices v
          if u = v
              dist[u, v, 0] ← 0
          else
              dist[u, v, 0] ← ∞

  for ℓ ← 1 to V − 1
      for all vertices u
          for all vertices v ≠ u
              dist[u, v, ℓ] ← dist[u, v, ℓ − 1]
              for all edges x→v
                  if dist[u, v, ℓ] > dist[u, x, ℓ − 1] + w(x→v)
                      dist[u, v, ℓ] ← dist[u, x, ℓ − 1] + w(x→v)
```

```
ALLPAIRSBELLMANFORD(V, E, w):
  for all vertices u
    for all vertices v
      if u = v
        dist[u, v] ← 0
      else
        dist[u, v] ← ∞
  for ℓ ← 1 to V − 1
    for all vertices u
      for all edges x→v
        if dist[u, v] > dist[u, x] + w(x→v)
          dist[u, v] ← dist[u, x] + w(x→v)
```

Time: $O(V^2 E) = O(V^4)$



(Assume $w(v \to v) = 0$.

$w(u \to v) = \infty$ if $u \to v \notin E$.

$\ell = 2^i$, Just make $\ell = 2^{\lceil \lg V \rceil}$ the largest choice.

$$dist(u,v,\ell) = \begin{cases} w(u \to v) & \text{if } \cancel{i = 1} \;\; \ell = 1 \\ \min\limits_{x} \big(dist(u,x,\ell/2) + dist(x,v,\ell/2) \big) & \text{otherwise} \end{cases}$$

what vertex is in the middle of my path?

```
FischerMeyerAPSP(V, E, w):
  for all vertices u
    for all vertices v
      dist[u, v, 0] ← w(u→v)
  for i ← 1 to ⌈lg V⌉          ⟨⟨ℓ = 2^i⟩⟩
    for all vertices u
      for all vertices v
        dist[u, v, i] ← ∞
        for all vertices x
          if dist[u, v, i] > dist[u, x, i − 1] + dist[x, v, i − 1]
            dist[u, v, i] ← dist[u, x, i − 1] + dist[x, v, i − 1]
```

$dist[V, V, 0..\lceil \lg V \rceil]$

$dist[u,v,i]$ stores $dist(u,v,2^i)$

$dist(u,v) = dist[u,v,\lceil \lg V \rceil]$

Time: $O(V^3 \log V)$ (with negative weights!)

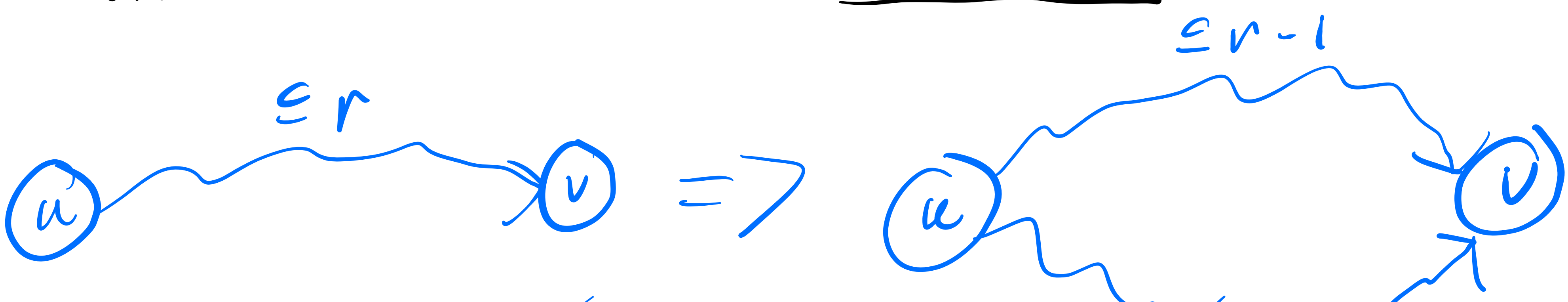
```
LeyzorekAPSP(V, E, w):
  for all vertices u
      for all vertices v
          dist[u, v] ← w(u→v)
  for i ← 1 to ⌈lg V⌉          ⟨⟨ℓ = 2^i⟩⟩
      for all vertices u
          for all vertices v
              for all vertices x
                  if dist[u, v] > dist[u, x] + dist[x, v]
                      dist[u, v] ← dist[u, x] + dist[x, v]
```

$O(V^3 \log V)$

A path <u>passes through</u> vertex x if it <u>enters</u> + <u>leaves</u> x. $(w \to x \to y \to z)$

Number the vertices arbitrarily 1 to $|V|$.

$\pi(u,v,r)$: the shortest path from u to v that passes through vertices only vertices numbered <u>at most</u> r.



$\pi(u,v,r)$

$\pi(u,v,0) = u \to v$.

$dist(u,v,r)$: length of $\pi(u,v,r)$.

$dist(u,v) = dist(u,v,|V|)$

$$dist(u,v,r) = \begin{cases} w(u \to v) & \text{if } r = 0 \\ \min \left\{ \begin{array}{c} dist(u,v,r-1) \\ dist(u,r,r-1) + dist(r,v,r-1) \end{array} \right\} & \text{otherwise} \end{cases}$$

```
KleeneAPSP(V, E, w):
  for all vertices u
    for all vertices v
      dist[u, v, 0] ← w(u→v)

  for r ← 1 to V
    for all vertices u
      for all vertices v
        if dist[u, v, r − 1] < dist[u, r, r − 1] + dist[r, v, r − 1]
          dist[u, v, r] ← dist[u, v, r − 1]
        else
          dist[u, v, r] ← dist[u, r, r − 1] + dist[r, v, r − 1]
```

$O(V^3)$

```
FloydWarshall(V, E, w):
  for all vertices u
    for all vertices v
      dist[u, v] ← w(u→v)

  for all vertices r
    for all vertices u
      for all vertices v
        if dist[u, v] > dist[u, r] + dist[r, v]
          dist[u, v] ← dist[u, r] + dist[r, v]
```

$O(V^3)$ time

($O(V^2)$ space)

(probably no $O(n^{2.999})$)

```
LeyzorekAPSP(V, E, w):
  for all vertices u
      for all vertices v
          dist[u, v] ← w(u→v)
  ~~for i ← 1 to ⌈lg V⌉      ⟨⟨ℓ = 2^i⟩⟩~~   i ← 1 to 3
      for all vertices u
          for all vertices v
              for all vertices x
                  if dist[u, v] > dist[u, x] + dist[x, v]
                      dist[u, v] ← dist[u, x] + dist[x, v]
```

```
FloydWarshall(V, E, w):
  for all vertices u
      for all vertices v
          dist[u, v] ← w(u→v)
  for all vertices r
      for all vertices u
          for all vertices v
              if dist[u, v] > dist[u, r] + dist[r, v]
                  dist[u, v] ← dist[u, r] + dist[r, v]
```

Hide, Kumabe, Maehara '19: You do get correct with any loop order as long as you run it 3 times

$O(V^3)$!