Given <u>directed</u> $G=(V,E,w)$

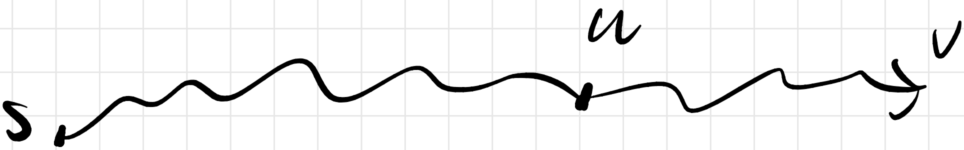
$w: E \to \mathbb{R}$.

Shortest path from $s \in V$ to $t \in V$ is the s,t-path P minimizes $w(P) = \sum_{u \to v \in P} w(u \to v)$.

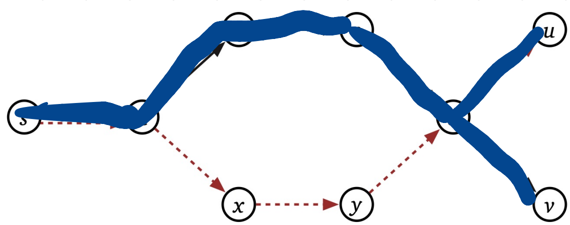
min $w(P)$ is the <u>distance</u> from s to t

Most shortest paths algs find shortest paths from s to <u>all</u> vertices v.

<u>Single-Source Shortest Paths</u>

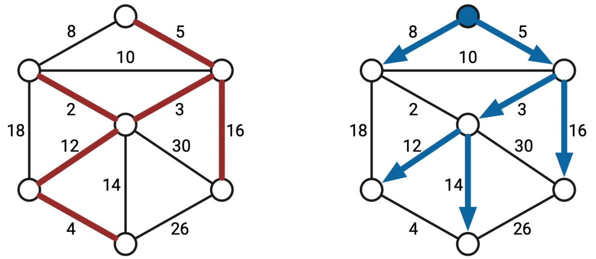


shortest paths contain shortest paths



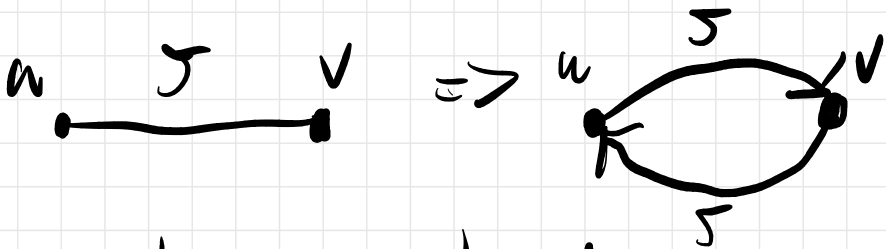
Can assume shortest paths follow a tree rooted at s.

Single-source shortest paths tree



MST $\neq$ SSSP

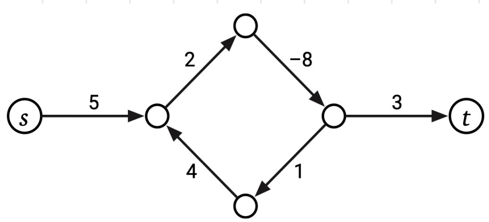
For undirected graphs, replace each edge with both of its orientations.



to make graph directed.

Weights may be negative!
We don't actually know how to find
shortest paths.

We can find shortest walks.



Shortest walks don't exist if there is a negative weight cycle.

o.w. some shortest walk is a path & SSSP tree exists

Ford, Dantzig, Minty ...

Maintain a pessimistic guess on distance to each vertex.

Two mutable variable per vertex v:

- $dist(v)$: upper bound on distance from s to v.

  $dist(s)$ starts at 0
  $dist(v)$ starts at $+\infty$, $\forall v \neq s$

- $pred(v)$: predecessor of v on same s to v walk.

  $pred(v)$ starts at Null

```
InitSSSP(s):
    dist(s) ← 0
    pred(s) ← Null
    for all vertices v ≠ s
        dist(v) ← ∞
        pred(v) ← Null
```

Call an edge $u \to v$ <u>tense</u> if $dist(u) + w(u \to v) < dist(v)$ implies $dist(v)$ is too high, so <u>relax</u> it

```
Relax(u→v):
    dist(v) ← dist(u) + w(u→v)
    pred(v) ← u
```

The "one" algorithm:

```
FordSSSP(s):
    InitSSSP(s)
    while there is at least one tense edge
        Relax any tense edge
```

Claims: Eventually teminates if no negative cycles.

When it does dist(v) = distance to v $\forall$ v. pred(v) = pred vertex on shortest path

if dist(v) = ∞, v is not reachible from s

But...

if any neg. weight cycle is reachible from s, we'll never terminate

Lemma: At all times, for any vertex v, $dist(v)$ is either ∞ or the length of some s,v-walk that ends with $pred(v) \to v$.

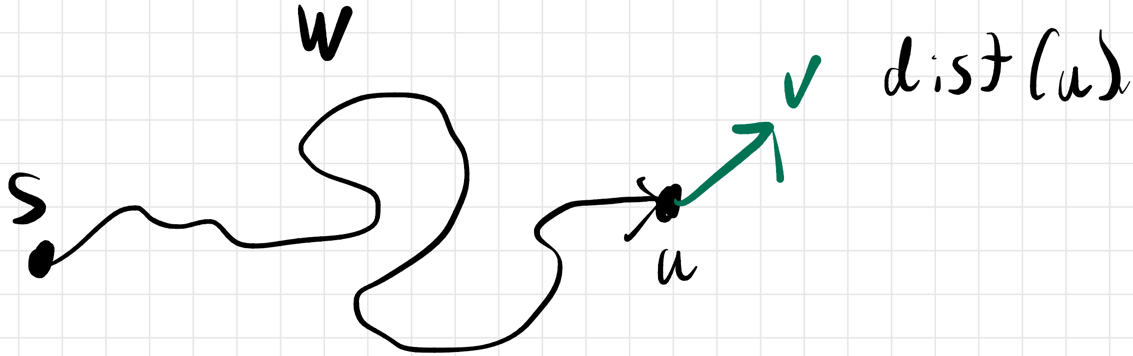
Proof: Induction on # relaxations

Suppose $dist(v) \neq \infty$ & the last change was $Relax(u \to v)$.

We $dist(v) \leftarrow dist(u) + w(u \to v)$ & $pred(v) \leftarrow u$.

$dist(u)$ was last set at an earlier relaxation.

So there is a walk W from s to u s.t. $w(W) = \operatorname{dist}(u)$



We just set $\operatorname{dist}(v)$ to be the length of $W \cdot u \to v$ which ends with $u \to v = \operatorname{pred}(v) \to v$.

Cor: $\Rightarrow$ $\operatorname{dist}(v)$ is alway $\geq$ distance from s to v.

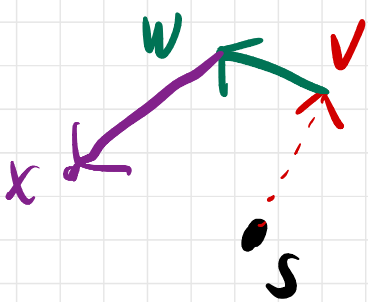
Never a Bad Idea: Bellman-Ford

```
BellmanFord(s)
    InitSSSP(s)
    while there is at least one tense edge
        for every edge u→v
            if u→v is tense
                Relax(u→v)
```

Relax ALL the edges!

(over and over)

Imagine FordSSSP. dist(v) is correct first time we relax s→v.



First time we relax v→w after setting dist(v), we're done.

First time we relax $w \to x$ after we relax $v \to w$ after we relax $s \to v$, $dist(x)$ is correct.

$dist_{\leq i}(v)$: length of shortest walk from s to v that uses at most i edges.

- $dist_{\leq 0}(s) = 0$
- $dist_{\leq 0}(v) = \infty \quad \forall v \neq s$
- if the shortest walk from s to v uses k edges,

  $dist_{\leq i}(v) > \text{distance}$ if $i < k$

  $dist_{\leq j}(v) = \text{distance}$ if $j \geq k$

Lemma: For every vertex v & non-neg i, after i iterations of outer while loop,

$$dist(v) \leq dist_{\leq i}(v).$$

Proof: If $i=0$, $dist(s) = dist_{\leq 0}(s) = 0$

$$dist(v) = \infty = dist_{\leq 0}(v)$$

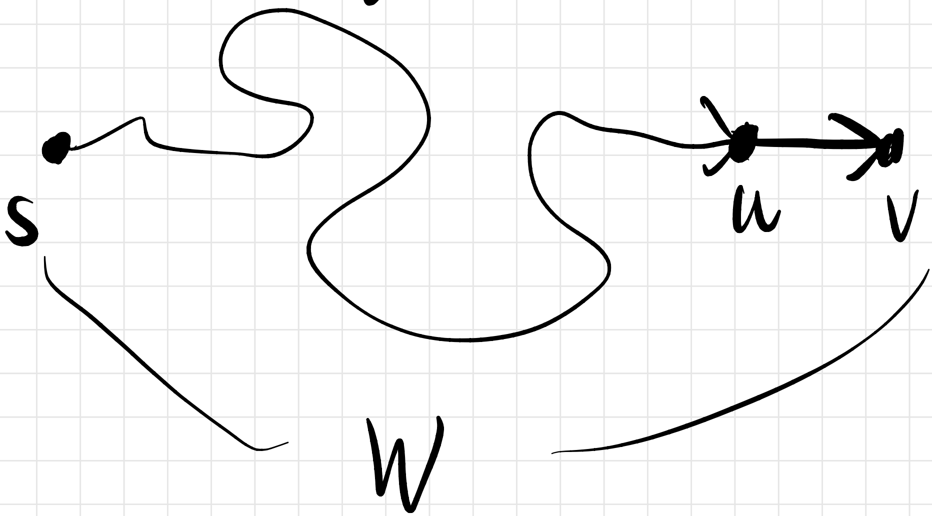
Suppose $i > 0$.

Let W be a shortest walk from s to v using $\leq i$ edges. By def. W has length $dist_{\leq i}(v)$.

If $W = \emptyset$, then $v = s$ & $dist_{\leq i}(v) = 0$.

$\operatorname{dist}(v) = \operatorname{dist}(s) \le 0 = \operatorname{dist}_{\le i}(v)$

o.w. Let $u \to v$ be the last edge of W.



By induction, $\operatorname{dist}(u) \le \operatorname{dist}_{\le i-1}(u)$ after $i-1$ iterations.

During iteration i, we check $u \to v$. At that moment

a) $\text{dist}(v) \le \text{dist}(u) + w(u \to v)$

-or- b) $\text{dist}(v) > \text{dist}(u) + w(u \to v)$

$\Rightarrow$ $u \to v$ is tense

$\Rightarrow$ we relax it so now

$\text{dist}(v) = \text{dist}(u) + w(u \to v)$

$\text{dist}(v)$ doesn't rise, so when iteration i ends,

$$\begin{aligned} \text{dist}(v) &\le \text{dist}(u) + w(u \to v) \\ &\le \text{dist}_{\le i-1}(u) + w(u \to v) \\ &= \text{dist}_{\le i}(v) \end{aligned}$$

Proof works even with neg cycles!

Suppose no neg. cycles...

Shortest walks/paths have $\le |V|-1$ edges.

$\Rightarrow$ dist(v) $\le$ distance after $|V|-1$ iterations

(we know dist(v) $\ge$ distance too..)

$\Rightarrow$ dist(v) = distance

O(E) per iteration, so

O(VE) time

Fail safe version:

```
BELLMANFORD(s)
  INITSSSP(s)
  repeat V − 1 times
      for every edge u→v
          if u→v is tense
              RELAX(u→v)
  for every edge u→v
      if u→v is tense
          return "Negative cycle!"
```

still $O(VE)$ time.