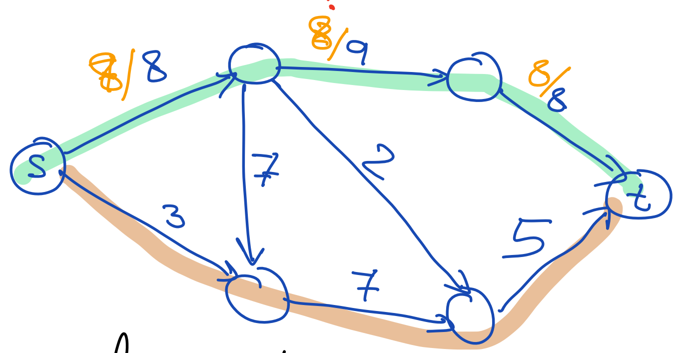
Network Flow

Given a Digraph $G(V, E, C)$, a Source s, and a target t, the goal is to find the Maximum flow from $s \to t$

e.g.



Capacity of an edge is the Max flow that can pass through that edge

① $\forall e \in E: \ f(e) \leq c(e)$

② $\forall v \in V, \ v \notin \{s, t\}: \quad \sum_{e \in in(v)} f(e) = \sum_{e \in Out(v)} f(e)$

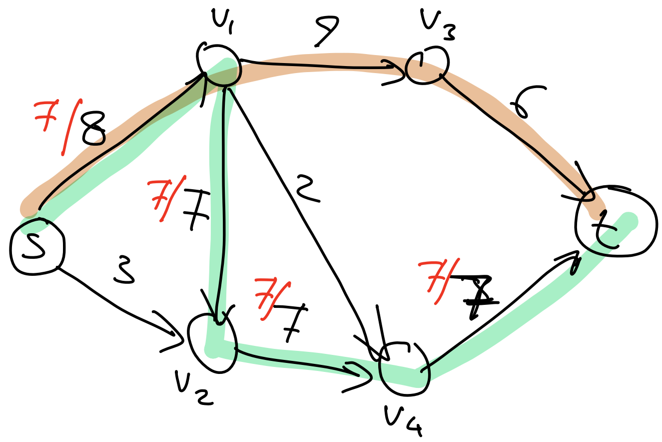
While $\exists$ Path $s \rightsquigarrow t$ s.t $C(e) - f(e) > 0 \ \ \forall e \in \text{Path}$

bottleneck = $\min_{\forall e \in \text{Path}} (C(e) - f(e))$

for $e \in$ Path

$f(e) = f(e) +$ bottleneck

let Path be Max b/n Path

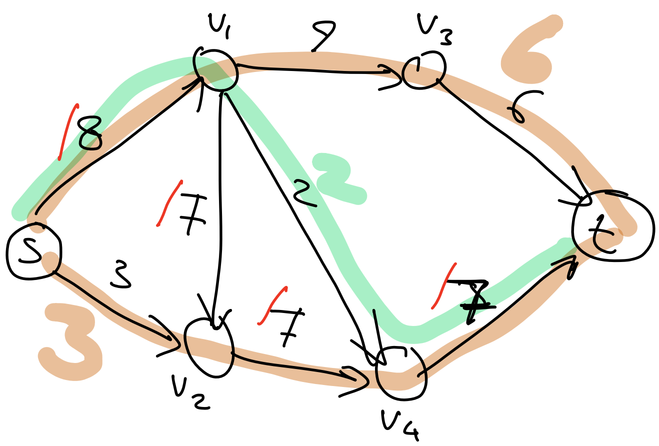


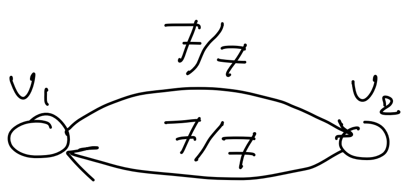
① S → 1 → 2 → 4 → t : 7

② S → 1 → 3 → t : 1

+

8 flow





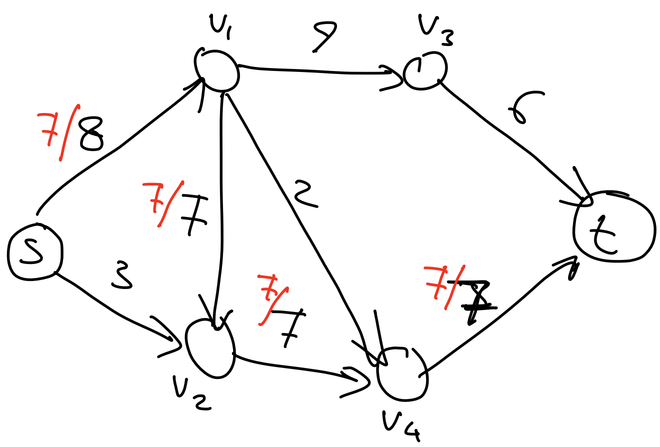
G_a : augmented Graph

$\forall e \in E$

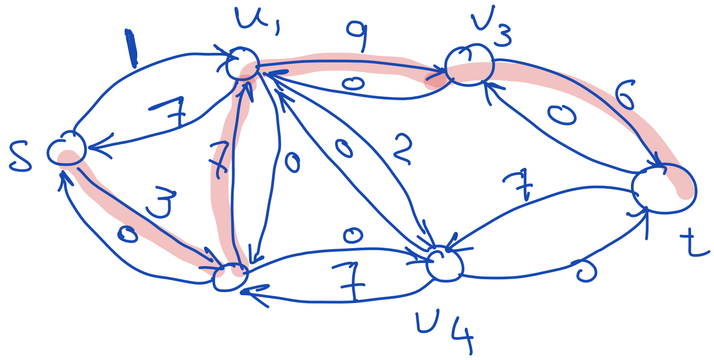
$e \in E_a \quad w_e = c(e) - f(e)$

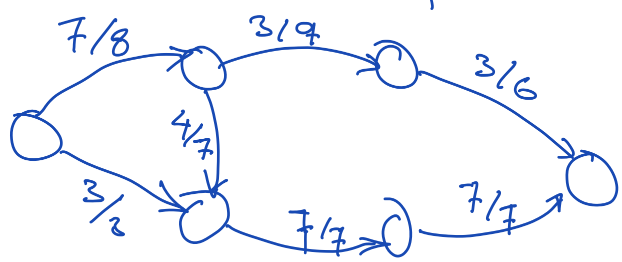
$e^{reverse} \in E_a \quad w_e = f(e)$

G



G_a

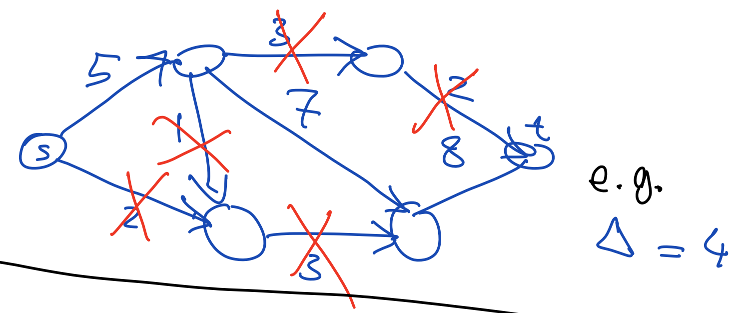




Ford-Fulkerson

```
maxflow(G(V,E), C, s, t)
    // Initialization
    for e ∈ E   f(e) = 0
    Ga ← A-graph(G, f)
        // ∀e ∈ E, add e ∈ Ea ; e' ∈ Ea      (e': reverse of e)
        // We = C(e) - f(e) ; We' = f(e)
    // Finding Paths
    while (∃ path p from s to t on Ga)
        tmp ← Augment(G, P, f)
        update(Ga, P, tmp)
    return f
```

```
Augment(G, P, f)
    bottleneck ← Min (We)
                 ∀e∈P
    for e ∈ P
        if e ∈ E : f(e) ← f(e) + bottleneck
        else // e is reverse
            f(e') ← f(e') - bottleneck
    return bottleneck
```



$\Delta = 2C$ // C is the max Capacity of any edge

$G_\Delta = \Delta\text{-augmented}(G)$ // remove all edges w/t Capacity Smaller than Δ

while ($\Delta \geq 1$) — $O(\log_2 C)$ iterations

  while ? $\exists$ a path P from s to t) — $O(m)$

    $f \leftarrow \text{augment}(P)$

    update (G_Δ)

  $\Delta = \Delta / 2$

  $G_\Delta = \Delta\text{-augment}(G)$

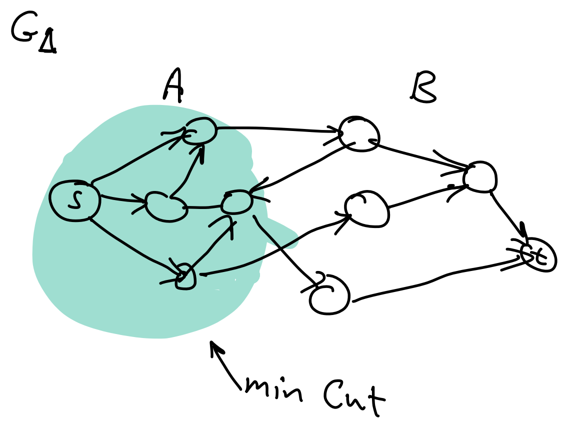
return f

$$O(\log_2 C \times ? \times m)$$

Lemma: at the end of every Δ iteration:

$$V_\Delta(f) \geq V^* + m\Delta$$

flow found on G_Δ — max flow



$$V_\Delta(f) = \sum_{out} C_\Delta(e) - \sum_{in} 0$$

$$\geq \sum_{out} (c(e) - \Delta) - \sum_{in} \Delta$$

$$\geq \sum_{out} c(e) - m\Delta$$

$$= Cap(A, B) - m\Delta$$

$$\geq V^* - m\Delta \quad \checkmark$$

---

at the beginning of iteration:

① $V_{2\Delta}(G) \geq V^* + m(2\Delta) = V^* + 2m\Delta$

② every Path at iteration Δ, gives a flow of at least Δ

①,② $\Rightarrow$ max # Paths at iter Δ is $\overset{\cancel{2}}{2m}$ $=?$

$\Rightarrow O(\log_2 C \; m^2)$ is the Runtime of the Alg.