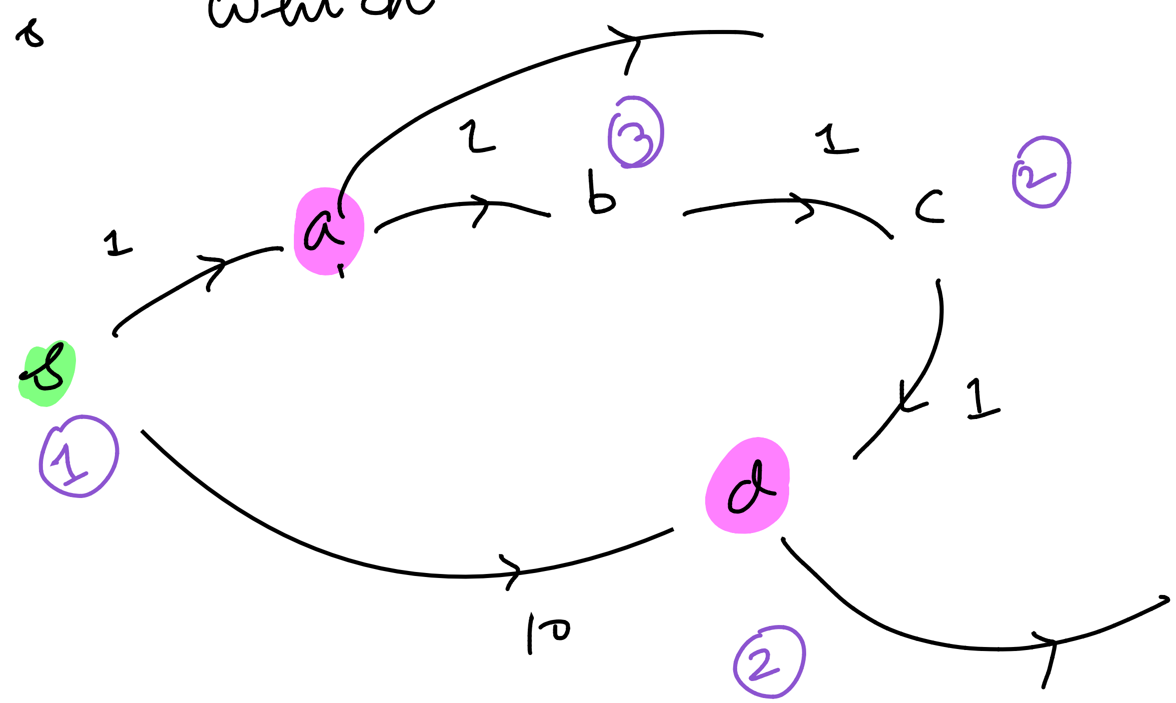
As and when a node finishes I will add to the front of a list

- Say I want to iterate over vertices instead of edges

(motivation: number of vertices are smaller than number in general $O(|E|) = O(|V|^2)$) → inner loop of Bellman Ford

Question is which vertex do I relax first



Moral: 1, 2, 3 ordering of vertices will not find shortest path from s to d.

If the directed Graph has no cycles $\Rightarrow$ Directed Acyclic Graph

For a DAG, we can order the vertices.

$\Downarrow$

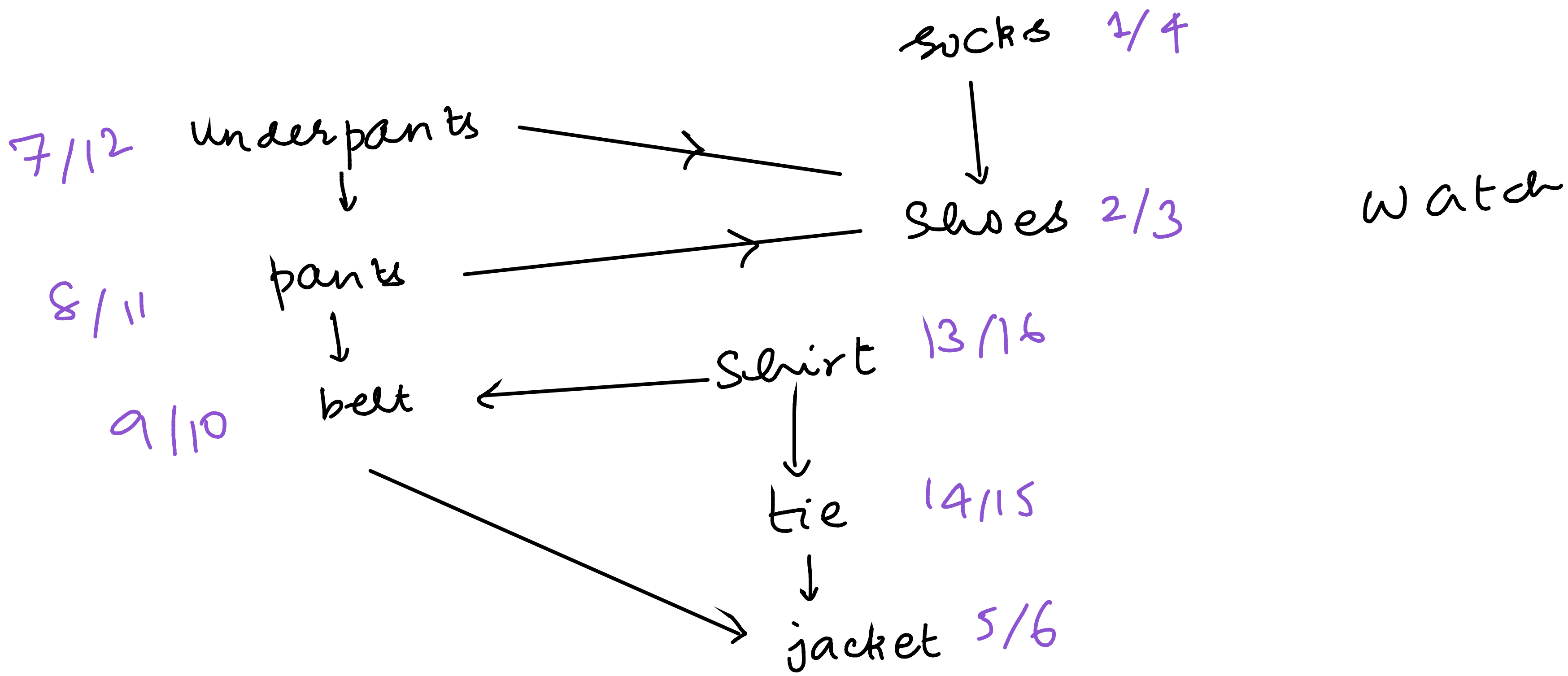
Topological Sort

TOPOGOLICAL-SORT(G)

call DFS(G) and compute finishing times $f[v]$

as vertex finishes insert in the front of the list

return list



$u \in V = \{$ socks, jacket, underpants, shoes, belt, shirt, watch $\}$ . . .

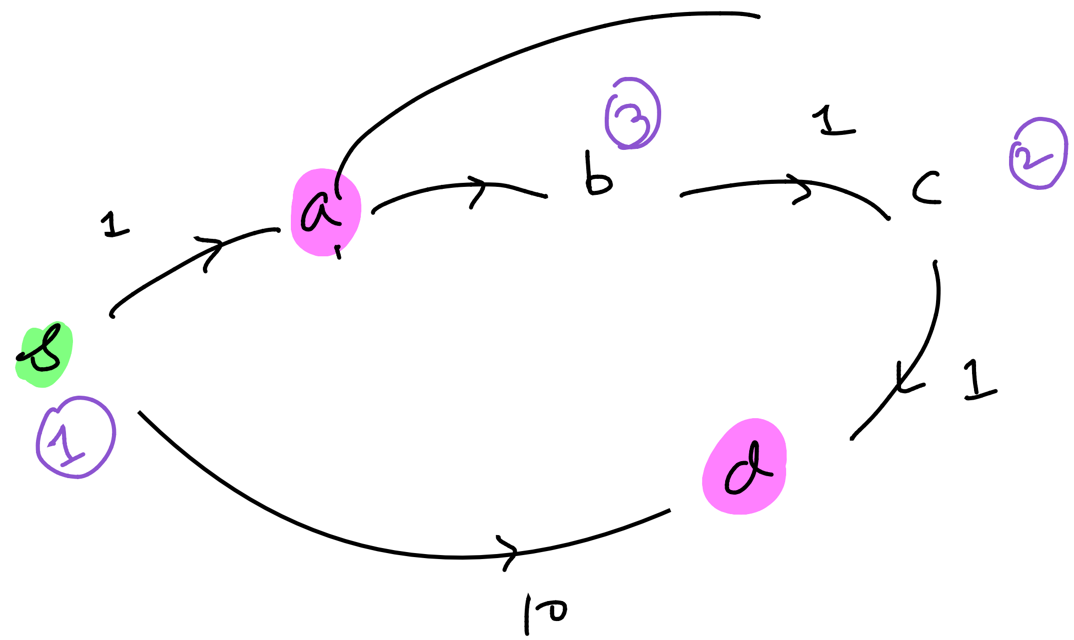
arbitrary

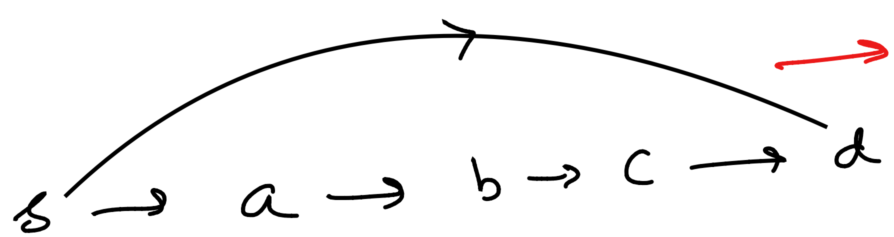
1

watch shirt tie underpants pants belt jacket socks shoe

topologically sorted order

Property Topological Sort : Edges go from left to right





→ Forward looking

As I step through the vertices, I have finished all the pre requisites.

DAG-SHORTEST-PATH (G, w, s)

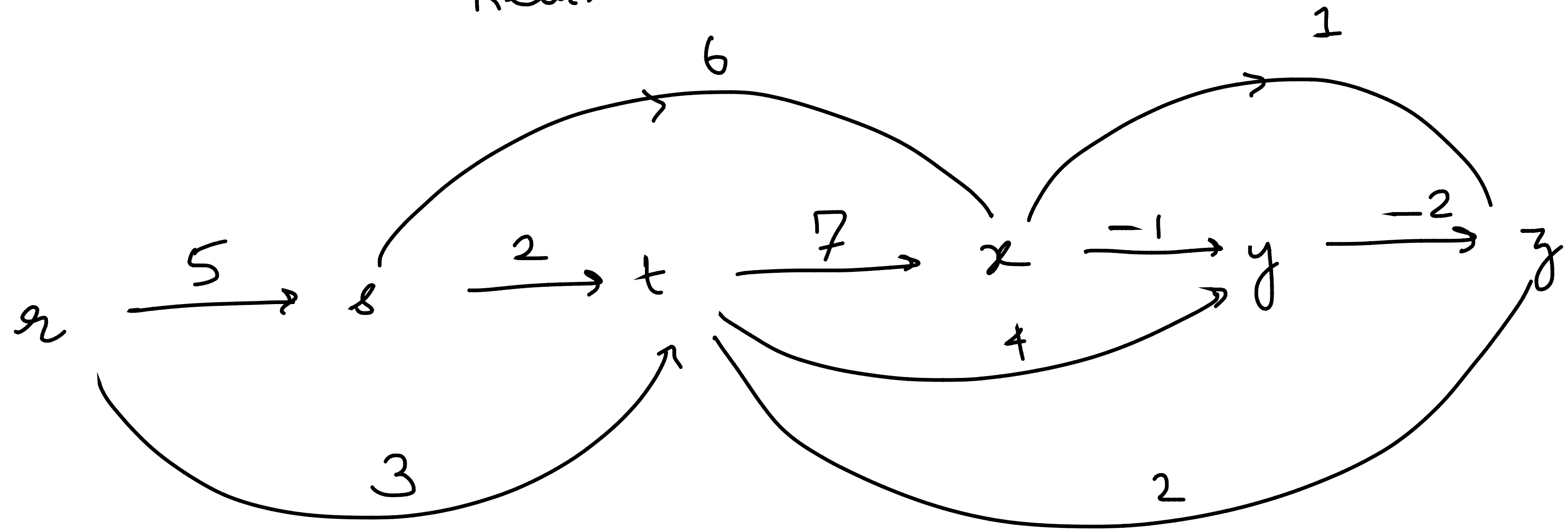
topologically sort the vertices of G

INIT-SINGLE-SOURCE (G, s)

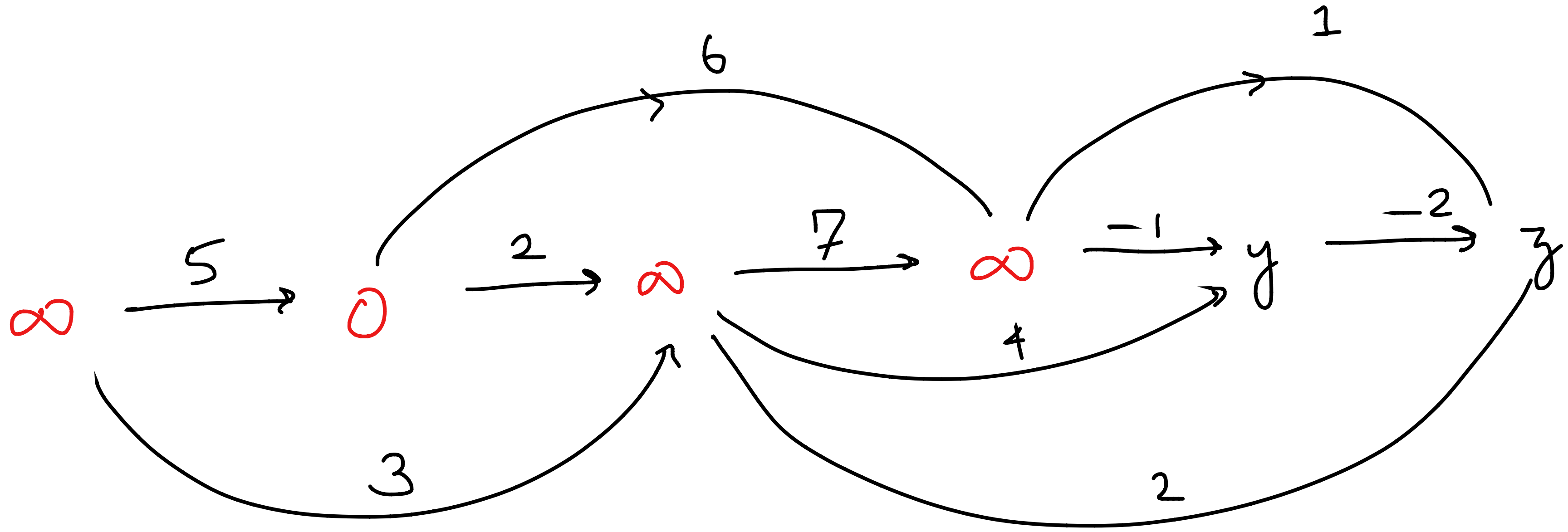
for each vertex u, taken in topologically sorted order

do for each v ∈ Adj [u]

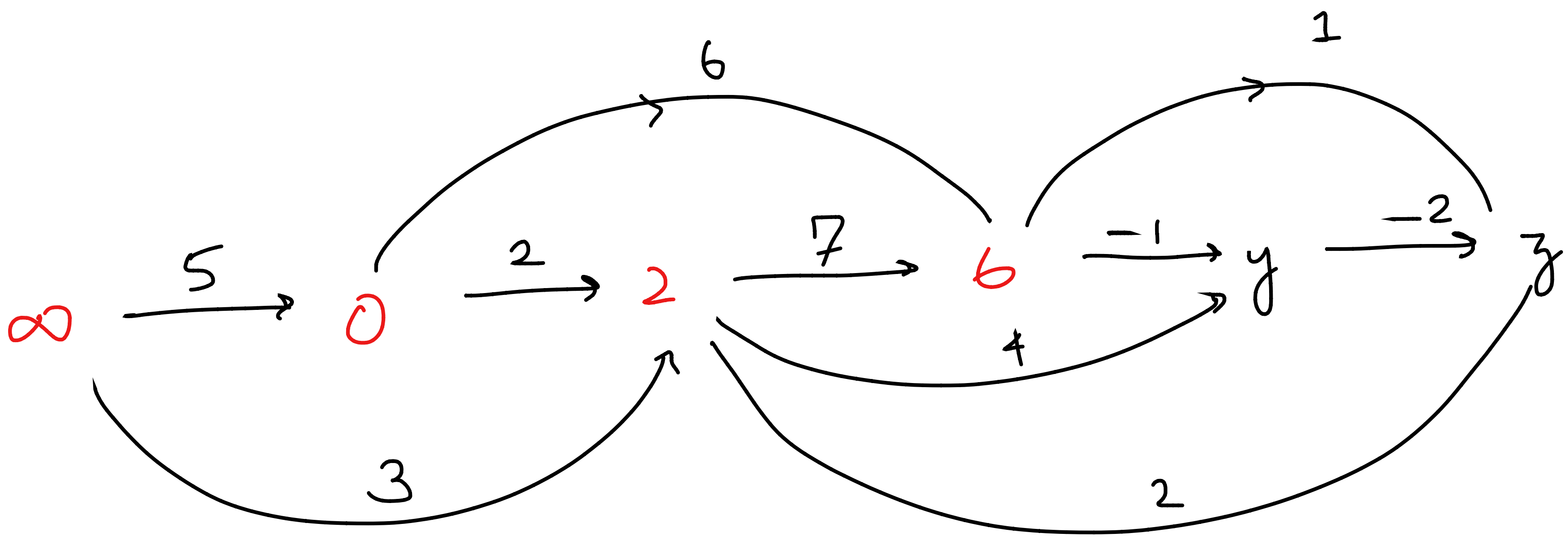
Relax (u, v, w)



r is relaxed first



s is relaxed next



Say we know edges are going to be non-negative

$w(u, v) \geq 0$

DIJKSTRA (G, w, s)

INIT-SINGLE-SOURCE (G, s)

$S \leftarrow \phi$

(Priority queue) $Q \leftarrow V[G]$

while $Q \neq \phi$

do $u \leftarrow$ EXTRACT-MIN (Q)

$S \leftarrow S \cup \{u\}$

for each vertex $v \in Adj[u]$

do Relax (u, v, w)

Difference between Dijkstra and DAG SP

* we don't have to sort first

* we sort on the fly

Proof: The first time we reach a vertex v, we reach it optimally.

vertices keep leaving the queue so we touch it only once

Total Number: $V+E$ steps

touched only once

For each extract-min V operations if we are using an array

$(V+E)V$

Via Min-Heap route: $(V+E)\log V$

$O((V+E)\log V) = O(E\log V)$

$O(V^2)$

All Path Shortest Problem:

- Table of $\delta(u,v)$

- Run single source shortest Path

(For general case) — Bellman Ford $O(V^2 E) = O(V^4)$

$O(V^2)$

(For non-negative edge case) — Dijkstra $O(VE \log V)$