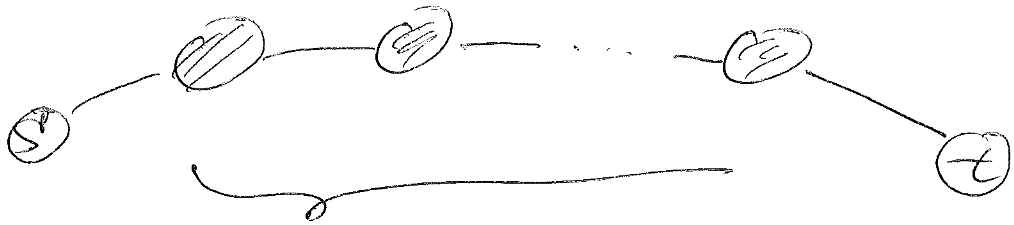
March 3

All pair shortest paths

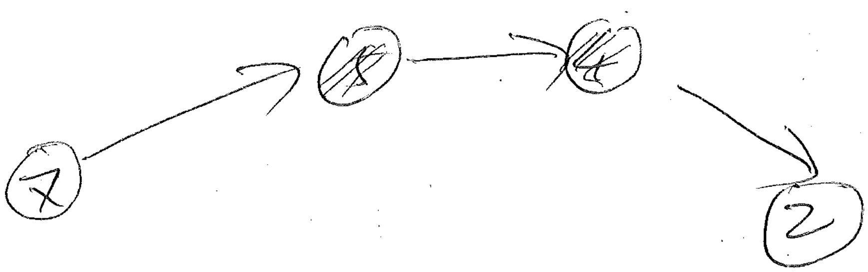
Recall intermediate vertices



intermediate vertices

k-path.

a path from s to t is called a k-path if all the labels of intermediate vertices are $\leq k$

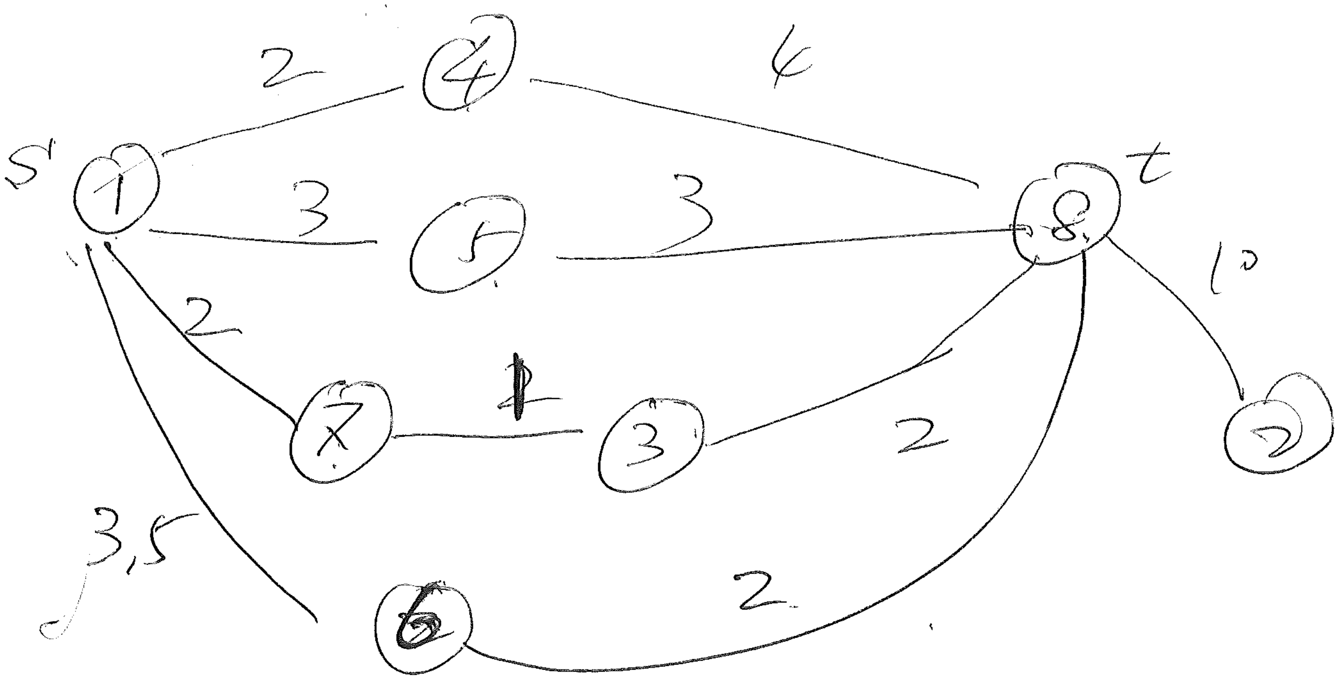


5-path

6-path

7-path

a shortest k-path from s to t is the shortest among all the k-paths from s to t.



6-paths from 1 to 8

1 → 4 → 8

1 → 5 → 8

1 → 6 → 8

shortest 6-path 1 → 6 → 8

shortest 7-path 1 → 7 → 3 → 8 5

shortest 8-path 1 → 7 → 3 → 8

shortest path = shortest 8-path

↗ larger label

Let $G(V, E)$ be a graph with n vertices

the vertices in V are labelled $1, 2, \cdots, n$

~~To~~ We will perform induction on $k = 0, 1, \cdots, n$

in each iteration, we will calculate all pair shortest k-paths, and store them in $D_{ij}^{(k)}$

where $D_{ij}^{(k)}$ stores the shortest k-path from vertex i to vertex j.

Basis $k=0$

we need the shortest 0-paths

$D_{ij}^{(0)}$ should contain the shortest 0-path from i to j,

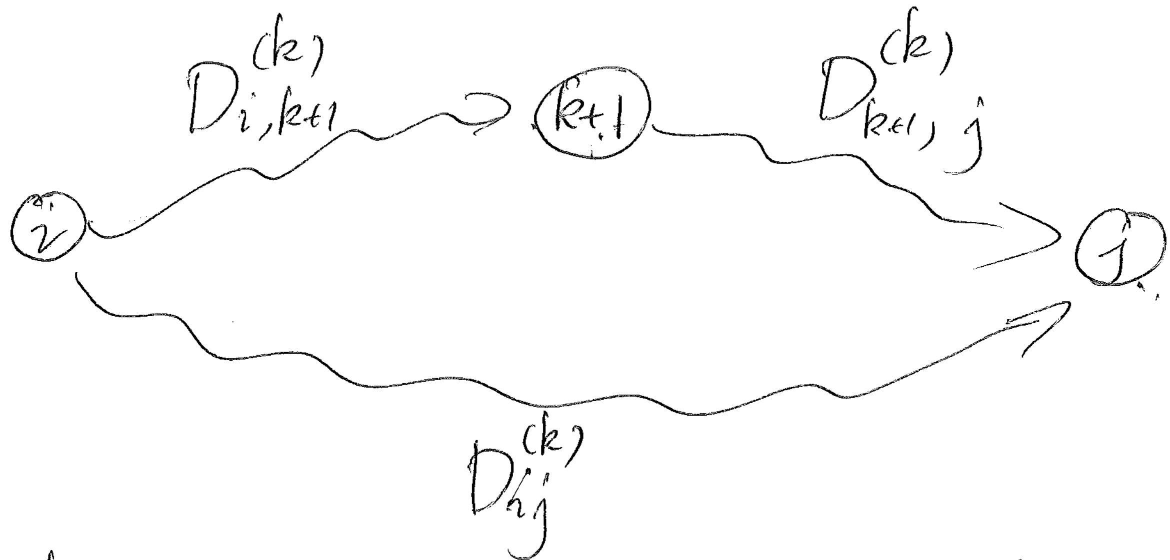
i.e., the shortest path from i to j where intermediate vertices all ~~have~~ labelled ≤ 0

Thus $D_{ij}^{(0)}$ = adjacency matrix of G.

I.H. assume we have $D_{ij}^{(k)}$, i.e., the shortest k-paths between every pair of verices

I.S. Need $D_{ij}^{(k+1)}$

Consider $D_{ij}^{(k+1)}$: shortest $(k+1)$ path from i to j.



$$D_{ij}^{(k+1)} = \min \left\{ D_{ij}^{(k)} , \; D_{i,k+1}^{(k)} + D_{k+1,j}^{(k)} \right\}$$

Floyd-Warshall Algorithm

input. adjacency matrix $D^{(0)}$ of $n \times n$

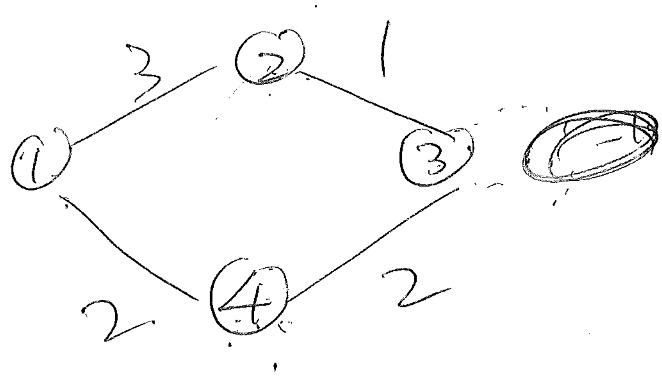
output $D^{(n)}$

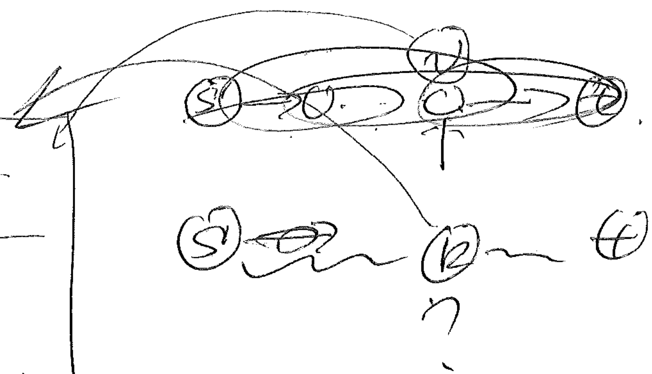
For $k = 1$ to n

For all i, j

$$D_{ij}^{(k)} = \min\{D_{i,k}^{(k-1)} + D_{kj}^{(k-1)}, D_{ij}^{(k-1)}\}$$

Running Time $O(n^3)$





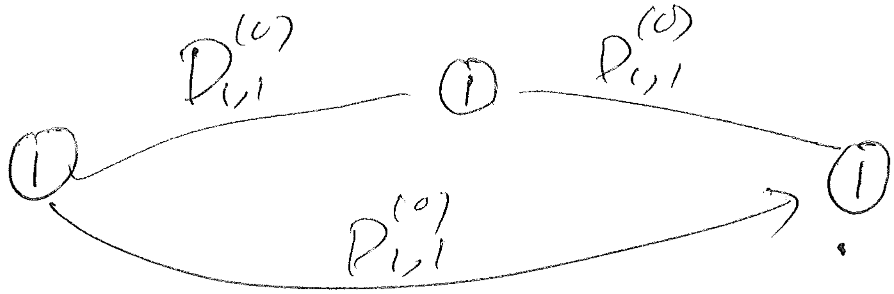
$D^{(0)}$

0	3	∞	2
3	0	1	∞
∞	1	0	2
2	∞	2	0

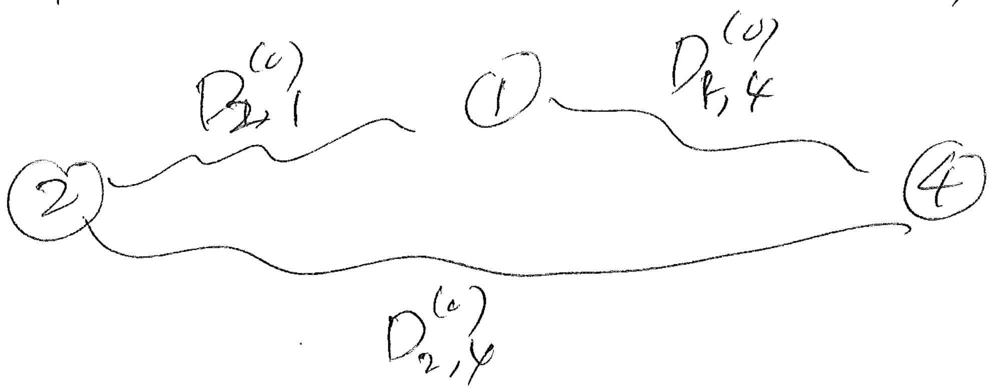
$D^{(1)}$

0	3	∞	2
3	0	1	5
∞	1	0	2
2	5	2	0

$$D^{(1)}_{1,1} = \min\{ D^{(0)}_{1,1} + D^{(0)}_{1,1} , D^{(0)}_{1,1} \}$$



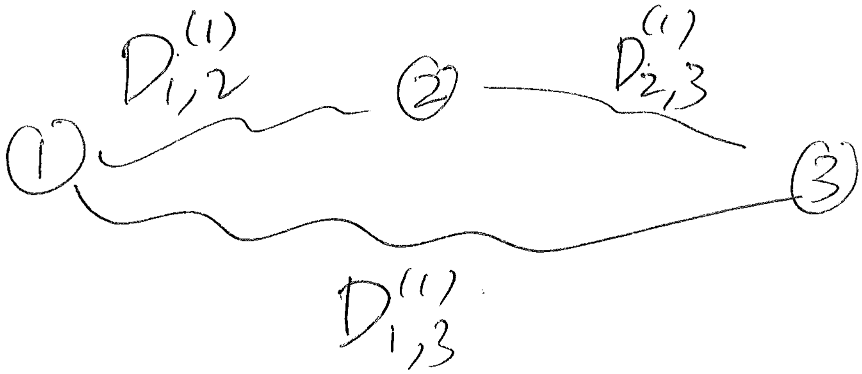
$$D^{(1)}_{2,4} = \min\{ D^{(0)}_{2,1} + D^{(0)}_{1,4} , D^{(0)}_{2,4} \}$$



$D^{(2)}$

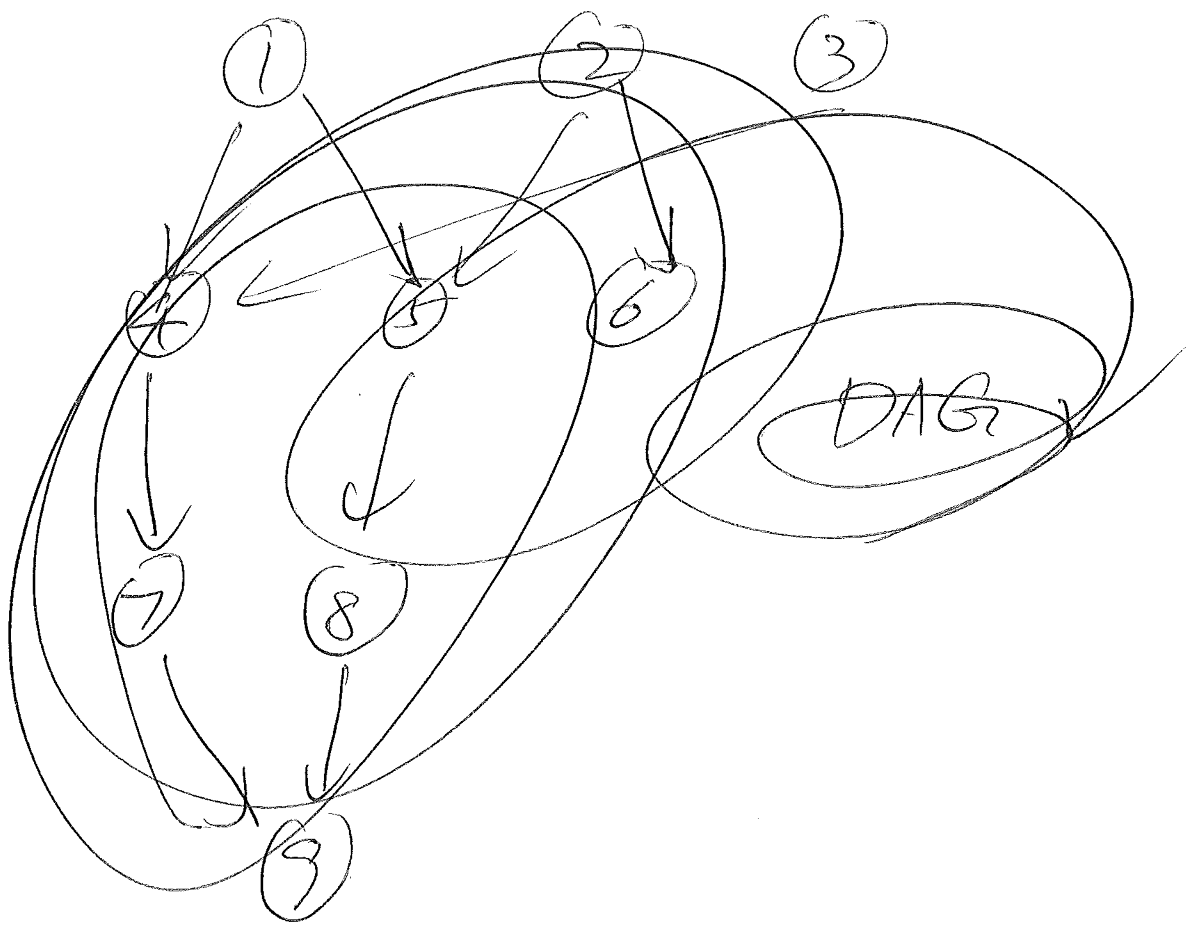
		2 / 4	

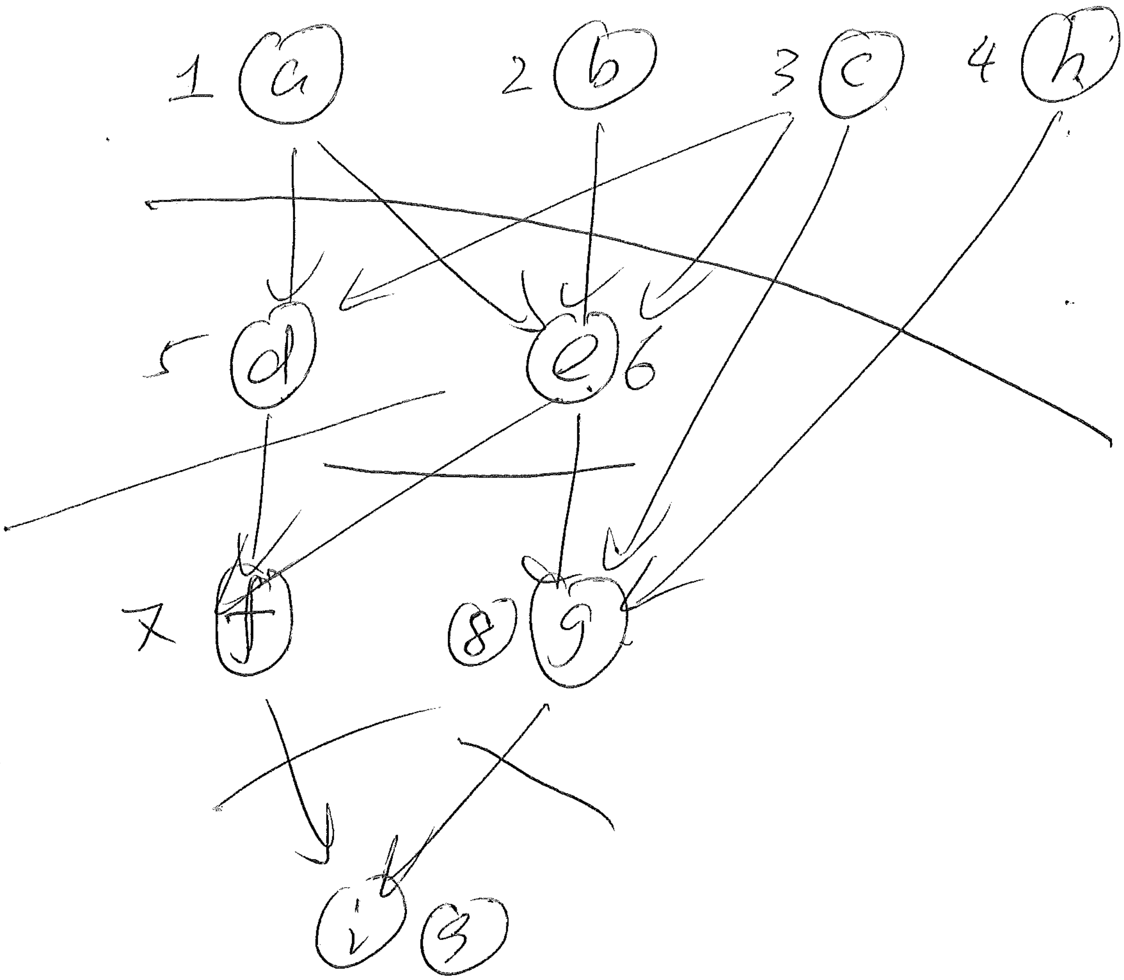
$$D^{(2)}_{1,3} = \min\{D^{(1)}_{1,2} + D^{(1)}_{2,3},\ D^{(1)}_{1,3}\} = \min\{3+1, \infty\} = 4$$



Def (Topological Order)

Given a DAG (Directed Acyclic Graph) G(V, E), ~~the~~ a topological order of V is a linear ordering of the vertices such that if G contains an arc (u,v) then u appears in front of v in the ordering





~~As long~~

input $G(V,E) \leftarrow$ DAG

output topological labels

Put all vertices in V with indegree 0 to a queue Q. label = 1

While Q is not empty

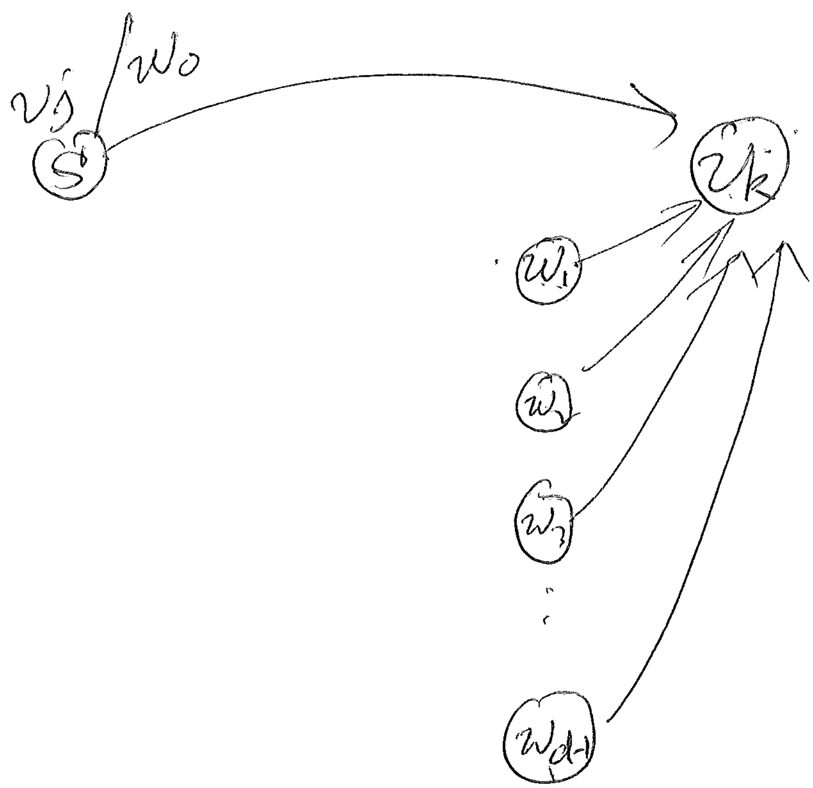
  v = deQueue(Q)

  v.label = label

  label++;

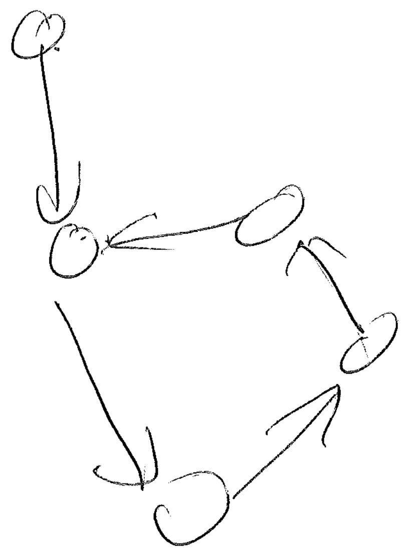
  For all arcs (v, w), w.indegree--

  If w.indegree = 0, enQueue(w)



induction on topological labels, the vertices with a smaller topological label will be calculated first.

Basis s



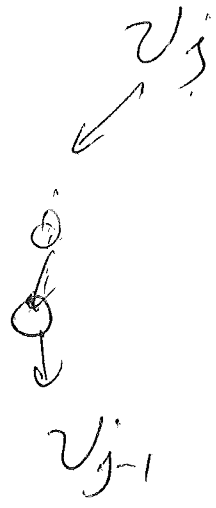
Shortest paths in DAG $G(V, E)$

Single source shortest path $s \in V$

Consider a topological order of G.

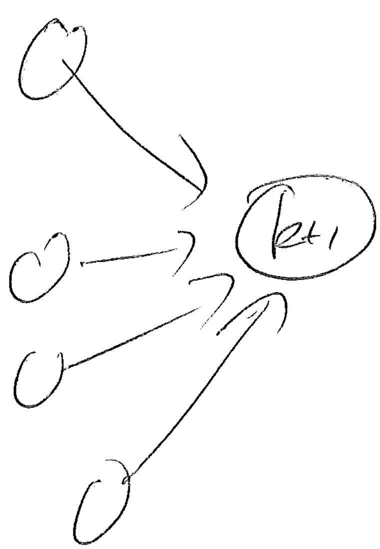
$v_1, v_2, \dots v_{j-1}, v_j = s, v_{j+1}, \dots, v_n$

∞



Assume the source S has a topolgical label 1
and all other vertices are reachable from S.
induction on the topological label k
Basis k = 1 ✓

I.H. assume we know the shortest paths from S = 1 to all vertices labelled ≤ k,
I.S. can we find the shortest path to k+1



For all (i, k+1), find the i that minimize $SP(1, i) + C(i, k+1)$

$O(|V| + |E|)$