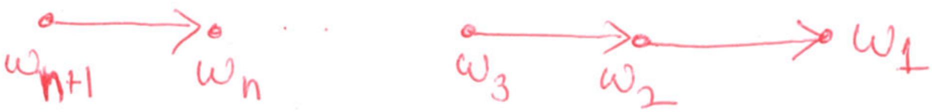
Oct 5 **Lemma 2:** If $G = (V, E)$ is a DAG $\Rightarrow$ $\exists$ a $w \in V$ s.t. w has 0 incoming edges.

Pf (idea): By contradiction.

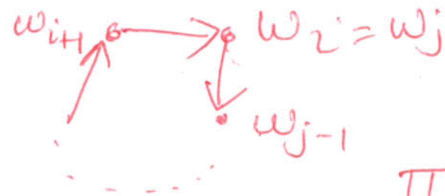
Assume G is a DAG but <u>all</u> vertices w have ≥ 1 incoming edge



There are $n+1$ nodes $w_1, \dots, w_{n+1}$ but only n distinct nodes

$\Longrightarrow$ (by pigeon hole principle) $\exists$ $i < j$ s.t. $w_i = w_j$

$\exists$ a cycle, a contradiction to fact that G is a DAG. $\Longleftarrow$

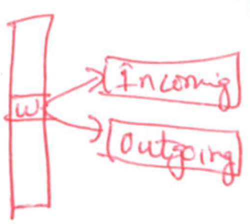


Correctness: Lemmas 2 + 3 + induction

TopOrd $(G = (V, E))$

1. If $V = \{u\}$; return u $\leftarrow O(1)$
2. Let w be a vertex that has 0 incoming edges $\leftarrow O(n)$
3. $G' = G \setminus \{w\}$ $\leftarrow$ can be done in $O(n)$ time
4. Return w; TopOrd(G') $\leftarrow O(n)$

Q1: Given w, how much time would it need to check if w has 0 incoming edges?



$\stackrel{?}{=} \{\} \rightarrow O(1)$ time

$\Longrightarrow$ can find such a w in $O(n)$ time

$\Rightarrow O(n)$ time in one recursive call

Also since in each call, the vertex w is removed

$\Rightarrow n$ recursive calls. $\Rightarrow O(n^2)$ time overall ■

Next: Show $O(m+n)$ runtime

Let In_w & Out_w be the incoming & outgoing degrees of w

we can show: $\sum_{w \in V} In_w = \sum_{w \in V} Out_w = m$

Lemma: Each recursive call takes $O(In_w + Out_w + 1)$ time.

$\Rightarrow$ Overall time $= \sum_{w \in V} O(In_w + Out_w + 1) = O\left(\sum In_w + \sum Out_w + n\right)$

$= O(m+n)$

Pf of Lemma: Main obs: Do not need to compute G' entirely.

$\rightarrow$ Only need to keep track of indegrees of the graphs

Data structures:

InDeg $\rightarrow$ array of length n InDeg[w] = indegree of w in current G.

$L \rightarrow$ linked list of vertices with 0 indegree in current G.

Initialization (i) InDeg[w] $= In_w \Longleftarrow O(In_w + 1)$ time

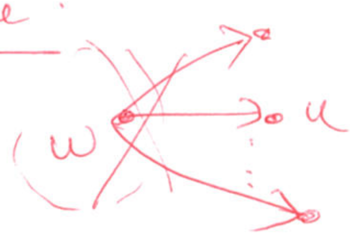
$\Rightarrow$ Overall tm $= O\left(\sum_w In_w + 1\right) = O(m+n)$

ii) L $\rightarrow$ Scan through InDeg & add w to L if InDeg[w] $= 0 \Longleftarrow O(n)$

Overall Initialization $= O(m+n)$

Query: Delete front of $L \rightarrow O(1)$

Update:



(i) InDeg:

For every $(w, u) \in E$ $\Big\}\ O(\text{Out}_w + 1)$

InDeg[u]--

If (InDeg[u] == 0) add u to L $\Big\}\ O(1)$

(ii) L : $\longrightarrow$ (If (InDeg[u] == 0) add u to L)

$\blacksquare$