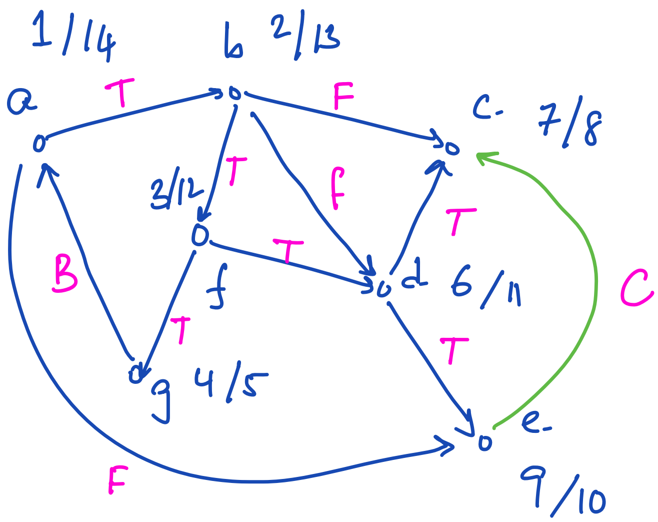
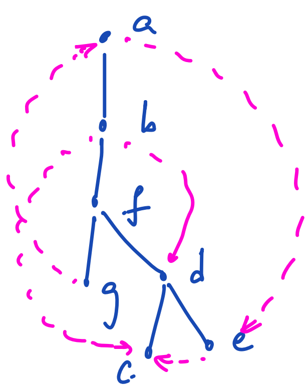
- No OH TODAY
- Exam 1
  - Q.1, Q.2, Q.3.
- Lot of grade still open.

Depth first Search





Thm: DFS on an undirected graph yields tree edges & back edges.

Proof Sketch: $e = (u,v)$: arb but particular edge in G.

WLOG, let $d[u] < d[v]$.

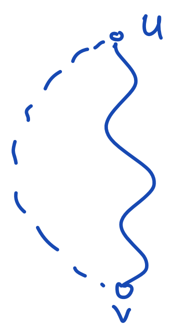
Claim: v is a descendant of u in the DFS forest.

White Path Theorem.

Case I: v is a child of u in the DFS forest.

- e is a tree edge

Case II: v is not a child of u.



e is explored first when the search is at v.

u is v's ancestor.

e is a back edge.

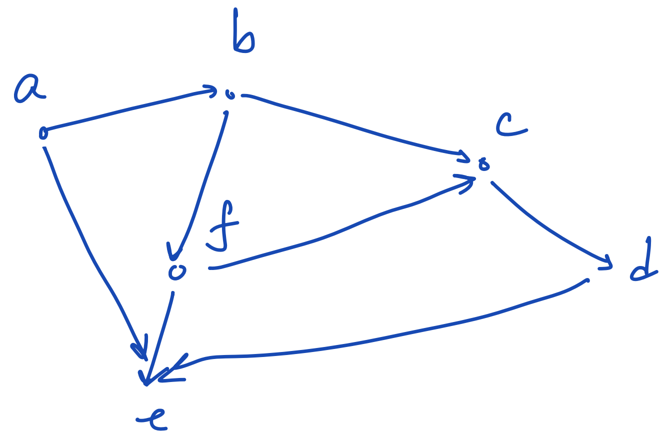
✓.

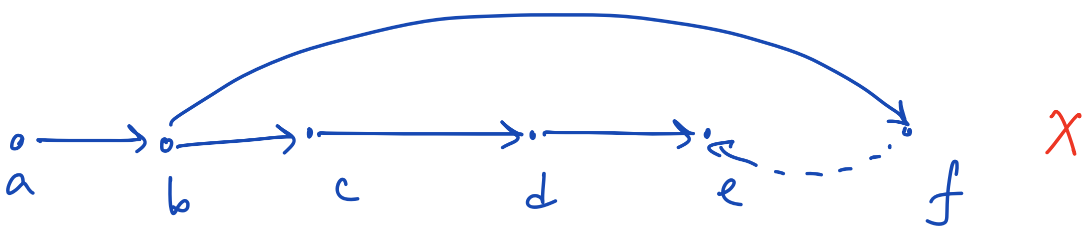
Topological Sort:

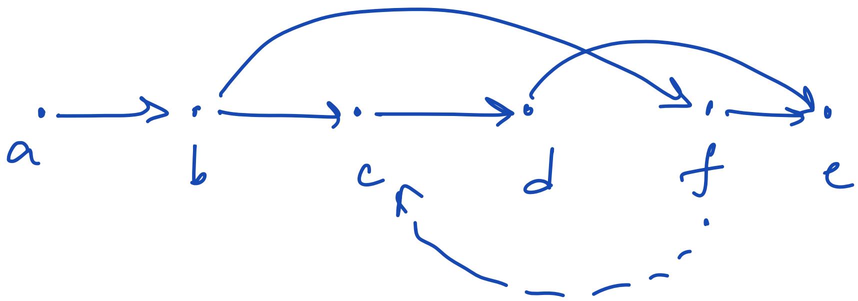
Input: Directed Acyclic Graph $G = (V, E)$

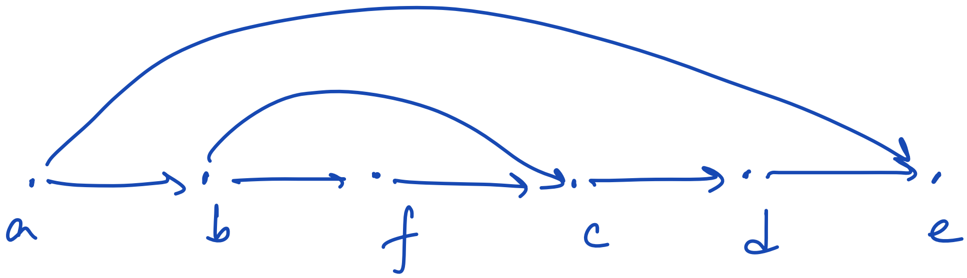
Objective: To obtain a topological sort of G.

order the vertices of G in a row s.t. all edges in E go from left to right.









Alg.

- Try every permutation of vertices.

  $n!$

- check if the edges go from left to right.

TS(G)

1. $u \leftarrow$ source vertex in G. // u does not have any incoming edges.

$\Theta(n)$

2. $G' \leftarrow G - u$

3. $L \leftarrow TS(G')$ // IH

L contains a permutation of vertices in G'.

4. Output u followed by vertices in L.

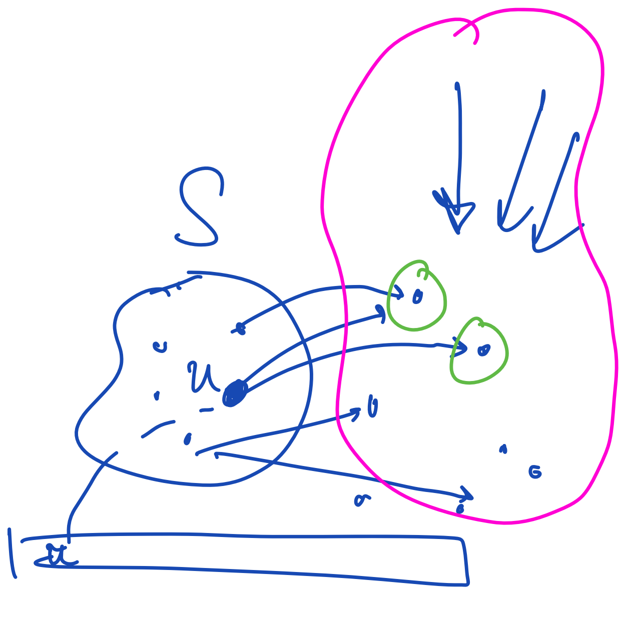
Running time: $O(n^2)$.

$n + n-1 + \cdots + 1 = \Theta(n^2)$.

$T(n) = T(n-1) + O(n)$

$= \Theta(n^2)$.

A better implementation.



$S \leftarrow$ vertices in G with indegree $= 0$.
(all sources of G)

$O(n+m)$

while $S \neq \emptyset$ **do**

  $u \leftarrow$ any vertex in S — $O(1)$

  append u to the o/p list L. — $O(1)$

  for each $v \in N(u) \cap V \setminus S \setminus L$ do

    indeg(v) --

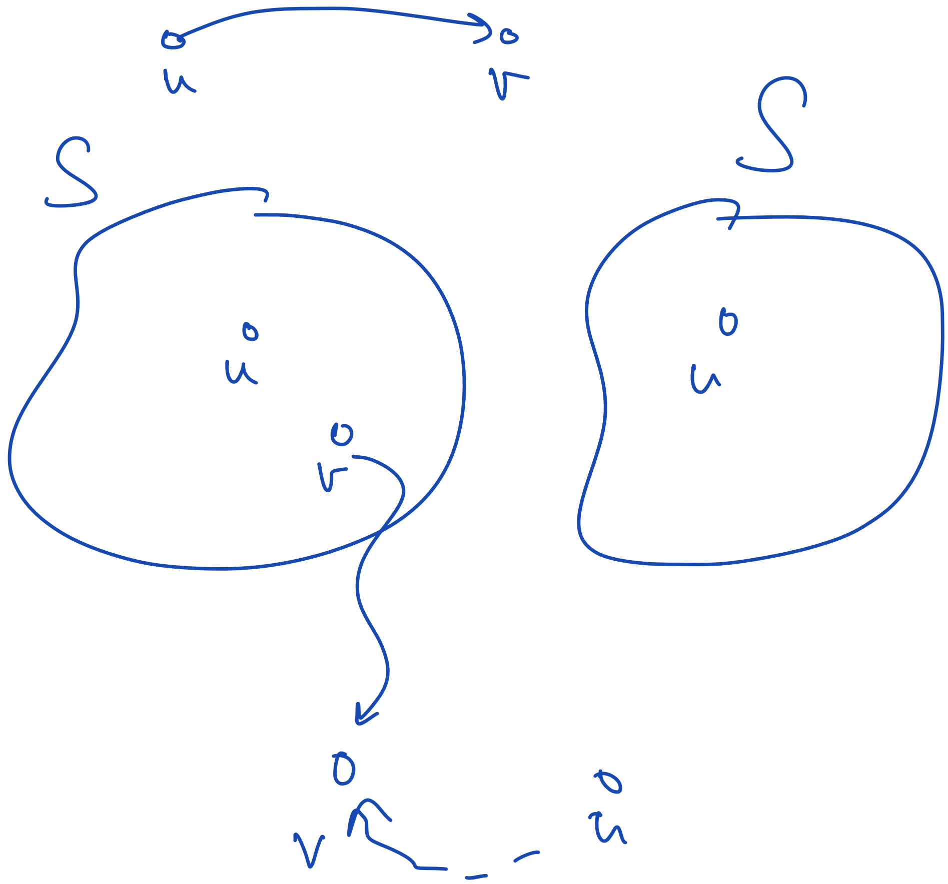
    if indeg$(v) = 0$ **then**

      $S \leftarrow S \cup \{v\}$

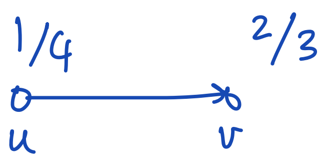
$$\sum_{u \in V} out(u) = \Theta(m)$$

output L

$\therefore$ Running time $= \underline{\underline{\Theta(n+m)}}$.



Alg.: Hint: Use DFS.



DFS(G)
Sort vertices in $\downarrow$ order of $f[\cdot]$.

3/4 1/2

<u>Alg.</u>

1. DFS(G) $O(n+m)$

2. Sort vertices in $\downarrow$ order of $f(\cdot)$. $O(n \lg n)$

$\therefore$ Running time: $O(n \lg n + m)$.

Add vertex to the front of the linked list while in DFS, when the vertex finishes.

Running time: $O(n+m)$.

Correctness : We want to show that the alg. works.

Proof Sketch : Let $e = (u,v)$ be any edge in G. We want to show that u appears to the left of v in the o/p. That is, we want to show that $f[u] > f(v)$.

Case I : $d[u] < d[v]$.

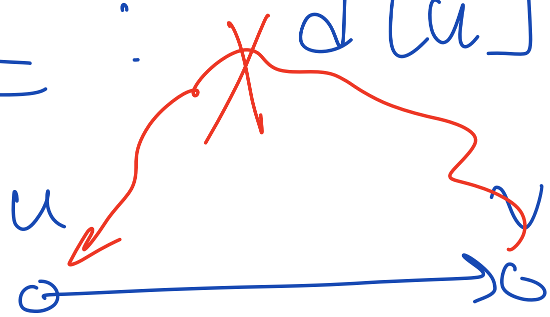
At $d[u]$ there is a trivial white path from u to v in G.

Thus by the WPT, $\cancel{f[u]} > f[v]$

v is a descendant of u in the DFS forest.

By the Parenthesis thm, we have $d_u < d_v < \boxed{f_v < f_u}$.

Case II : $d[u] > d[v]$



G is a DAG.

~~At $d[v]$ there is a WP from v to u in G & hence by WPT, $f[u] > f[v]$~~ ??

At $d[v]$ there is no WP from v to u in G.

Thus, by WPT, u is not a descendant of v in the DFS forest. By the PT, we have

$$\underset{d_v \qquad f_v}{\longmapsto} \qquad \underset{d_u \qquad f_u}{\longmapsto}.$$

$\therefore f_u > f_v$ ✓.

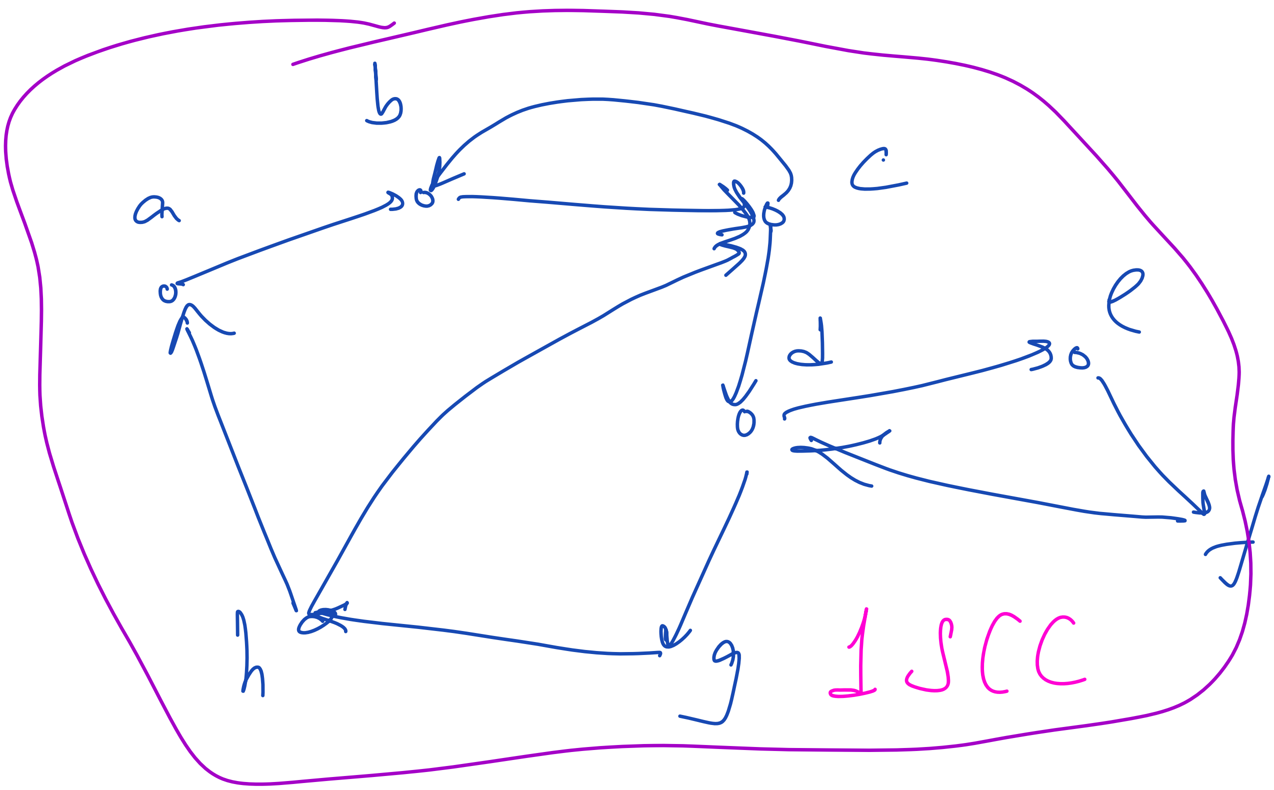
Strongly Connected Components (SCC)

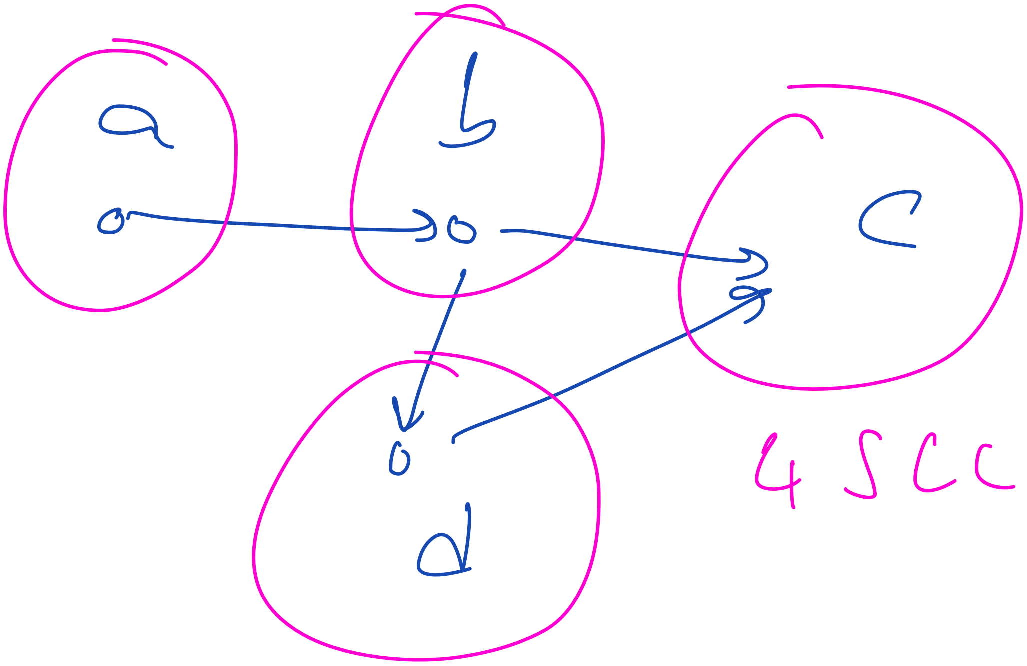
I/P : Directed graph $G = (V, E)$

O/P : To find all SCCs of G.

$H = (V_H, E_H)$ is a SCC of $G = (V, E)$ if

- H is a subgraph of G
- $\forall$ u, v in V_H, $u \neq v$, $u \rightsquigarrow v$ & $v \rightsquigarrow u$ in G.
- H is maximal.





4 SCC