Discussion Week 2

1. Warm Up
2. TopSort
3. S-t strongly connected

11. True or False?
   a) Running DFS on a directed acyclic graph (DAG) results in no back-edges, i.e., a back-edge means cycle, DAG has no cycles.
   b) In a directed graph G with 2 SCCs, it is always possible to add one edge to make it 1 SCC.

   \[ a \xrightarrow{1} b \xrightarrow{2} d \text{ can't add edge to make 1 SCC.} \]

12. TopSort: Input: DAG
    
    output: A numbering \( n(v) \) of vertices such that for all edges \( (u,v) \) in the graph, \( n(u) < n(v) \).

    "Linearize" graph.
    All DAGs can be linearized w/DFS!

   Only edges for which \( post(u) = post(v) \) are back edges... DAG has none!

   decreasing post: \( a, b, c, d \)
   \( n(v) = 1, 2, 3, 4 \)
    All edges go from left to right.
Input: DAG $G = (V, E)$; vertices $s, t$
Output: Number of different paths from $s$ to $t$.

Example:

```
Ex:
   s
  / \
 /   \   \\
 a   b   c
   \   /
    \ /
     v
```

$s, b, t$
$s, a, b, t$
$s, b, c, t$
$s, a, b, c, t$

$= 4$ paths.

Idea: TopSort

```
S, a, b, c, E, e, d
```

- Let $\text{numpaths}[v] = \# \text{of paths from } v \text{ to } t$.
- If $v$ is to right of $t$ in TopSort:
  - $\text{numpaths}[e] = \text{numpaths}[d] = 0$
- $\text{numpaths}[E] = 1$
- $\text{numpaths}[V] = \sum_{e=(v, x) \in E} \text{numpaths}(x)$
  
  e.g.
  
  $\text{numpaths}(c) = \text{numpaths}(t) = 1$
  
  $\text{numpaths}(b) = \text{numpaths}(c) + \text{numpaths}(t) = 2$

Alg: NumPaths($G, s, t$)

1) TopSort($G$) (uses DFS + post numbers), let $n(v)$ be their TopSort order
2) Set $\text{numpaths}(v) = 0 \ \forall v \ \text{ s.t. } n(v) > n(t)$
3) Set $\text{numpaths}(t) = 1$
4) For $\text{TopSort}(G) - 1 \to 1$ decreasing
   - Let $X$ have $n(x) = i$
   - Set $\text{numpaths}(x) = \sum_{e=(x, z) \in E} \text{numpaths}(z)$
5) Return $\text{numpaths}(s)$.
**Correctness**

Claim: $G$ has $k$ paths from $s$ to $t$ iff $\text{NumPaths}(G,s,t)$ returns $k$.

SubClaim: $\text{NumPaths}(x) = \# \text{ of paths from } x \text{ to } t$.

**PF:** By induction.

Base: Let $x \neq s,t$ have $n(x)+1 = n(t)$.

- If there's an edge $(x,t)$ then $\text{NumPaths}(x) = 0$.
- For $x$ with $n(x) > n(t)$, $\text{NumPaths}(x) = 0$.

Induction: Assume $\text{NumPaths}(x') = \# \text{ of paths from } x' \text{ to } t$. For $x > n(x) \geq n(t)$.

Show $\text{NumPaths}(k)$ is correct.

- Any way to get to $t$ is through $k$'s neighbors which all have $\text{TopSort}#$, bigger than $k$. So $\text{NumPaths}(k) = \sum_{(k,j) \in E} k$.

Back to claim:

$\Rightarrow$: Assume $G$ has $k$ distinct paths from $s$ to $t$.

Since we'll return $\text{NumPaths}(s)$, $k$ will be returned by subclaim.

$\Leftarrow$: If $\text{NumPaths}$ returns $k$, then there must be $k$ paths to $t$ from $s$ by subclaim.