Strongly Connected Components (SCC) in a Directed Graph

Strongly Connected Components

* SCC is meaningful to directed graphs only.

$G = (V, E)$: digraph

$\Downarrow$

is Strongly connected iff $\forall (x, y),\ x \in V,\ y \in V$, $\exists$ a path from u to v.

$\Downarrow$

if G is SC and there is a path from x to y in G, then it follows that there is also a path from y to x, Since SC means there exists a path b/w every pair of vertices in the graph G.

Strongly Connected $\Longleftrightarrow$ path exists b/w every pair of vertices

Relation: R

$$a\,R\,b \equiv (a,b) \in R$$

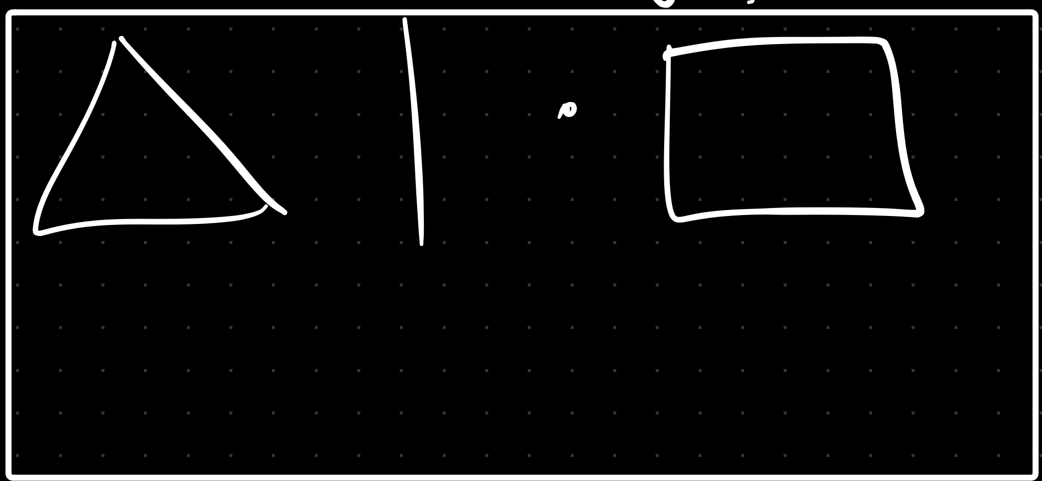
$R: V \longrightarrow V$ where $G=(V,E)$

$u\,R\,v$ iff $v \longrightarrow u$ paths exist
$u \longrightarrow v$

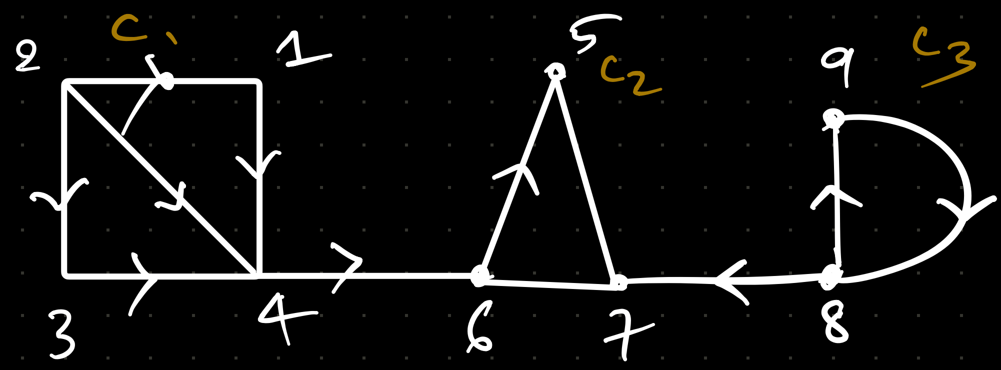
This is an Equivalence Relation.

$\Downarrow$

Induces a partition on V, the set of vertices in the graph G.



$G \longrightarrow 4$ Components



: G

No path from 5 to 1. Hence G is not Strongly Connected.

$$c_1 \longrightarrow c_2 \longleftarrow c_3$$

: Condense(G)

Condensed graph of G.

every c_i is a strongly connected graph among its own vertices.

it is a DAG.

* To determine a graph G is Strongly Connected or not, perform dfs from u to v and also from v to u if v is reachable from u, $\forall\ u, v \in V$, where $G = (V, E)$.

$\Rightarrow$ if $|V| = n$, then

$2\binom{n}{2} = n(n-1)$ times DFS

is incurred to check if G is

Strongly Connected or not?

Total Cost $= O(n^2 m)$ very huge!

$\downarrow$

$|E| = m$

Do we need to perform $n(n-1)$ DFS calls?

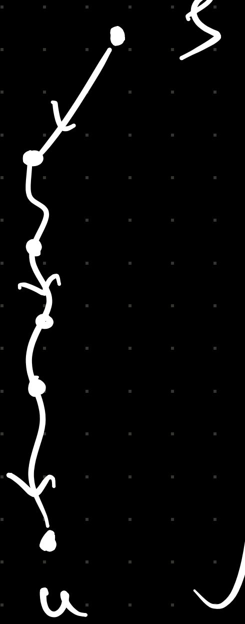
Can we modify the original graph G and solve the problem with that?

Fortunately, the answer is yes to both.

How about a graph reversal?

$G^R = (V, E')$ where $(v, u) \in E'$ iff $(u, v) \in E$

Just reversing all the edges of the original graph



DFS(G, s) $\quad s \in V$

Covering all the vertices in G

if are still vertices left, then the G might be having SCCs or not strongly connected at all.

* Assume $s \longrightarrow v_1 \longrightarrow v_2 \longrightarrow \dots \longrightarrow u$ as the path resulted from DFS(G, s) and also assume G is Strongly Connected.
* Now, from DFS(G, s), we learn that there exists a path from s to all other vertices in G.

Does this mean $\exists$ a path from u to s?

How do we find that?

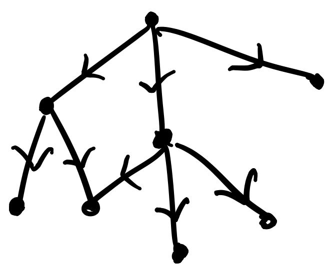
performing $O(n^2)$ DFS is naive and is Computationally expensive if G is large.

Now, we've established a path from s to u covering all the vertices in the graph.

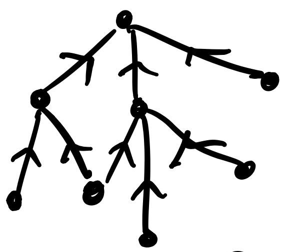
We can leverage on this fact and perform a DFS on the reversed graph G^R from the vertex u.

If there exists a path from u to s in G^R, then we can establish that our original graph G is strongly connected.

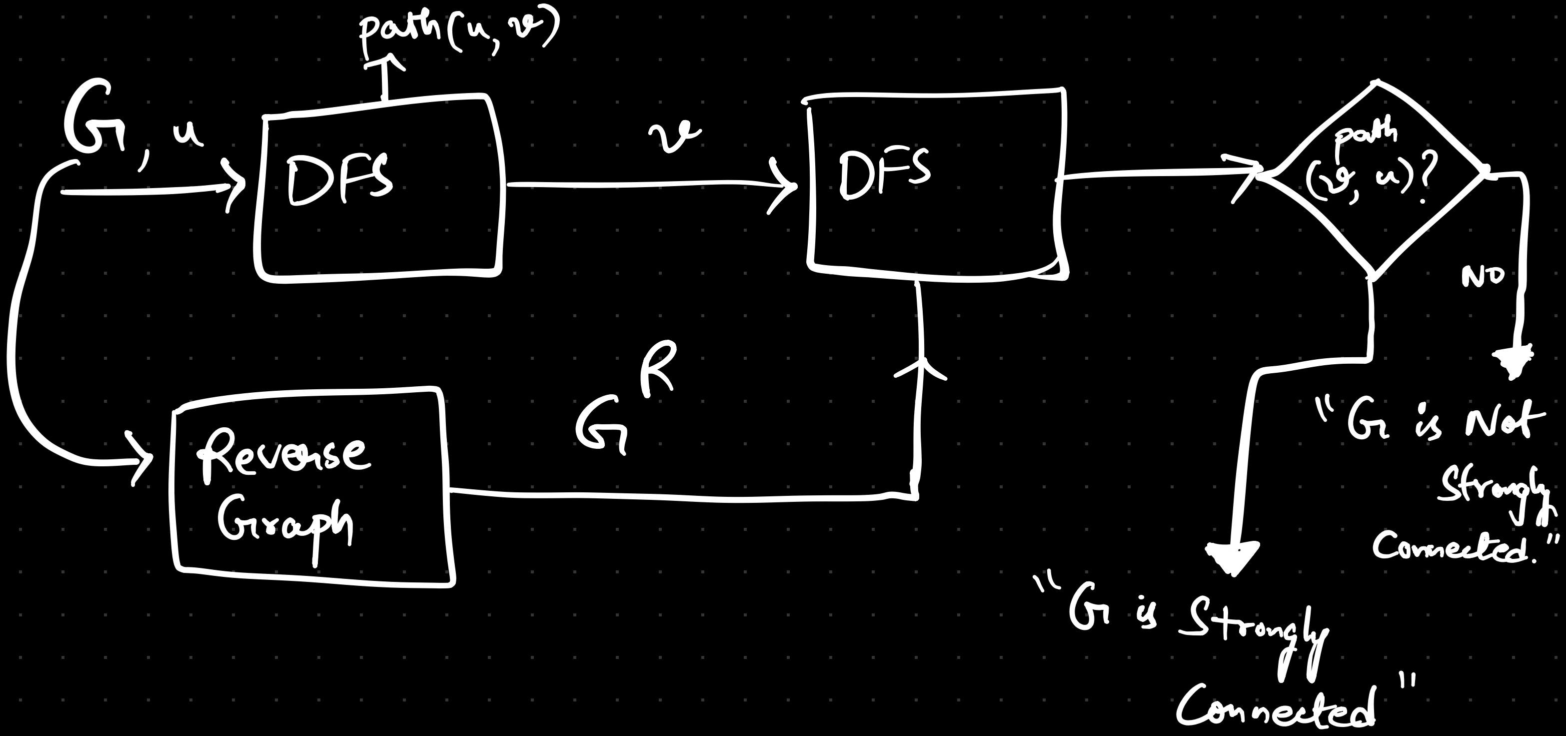
That's it. we're done. Instead of $n(n-1)$ DFS traversals, we have boiled the problem down to 2 DFS traversals by simply doing a transformation on the input graph G.



G



G^R


path(u, v)
G, u
DFS
v
DFS
path (v, u)?
NO
Reverse Graph
G^R
"G is Not Strongly Connected."
"G is Strongly Connected"