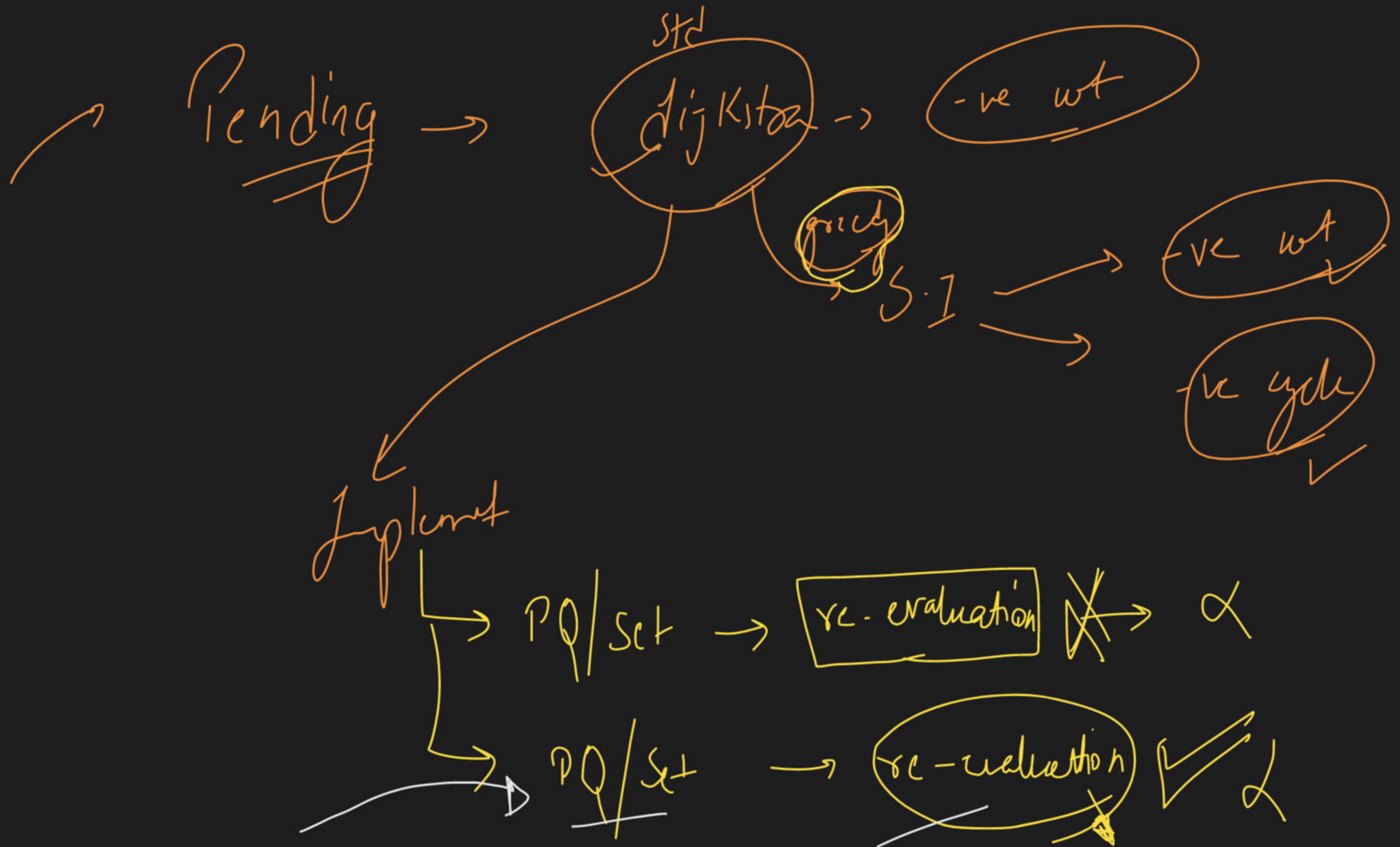
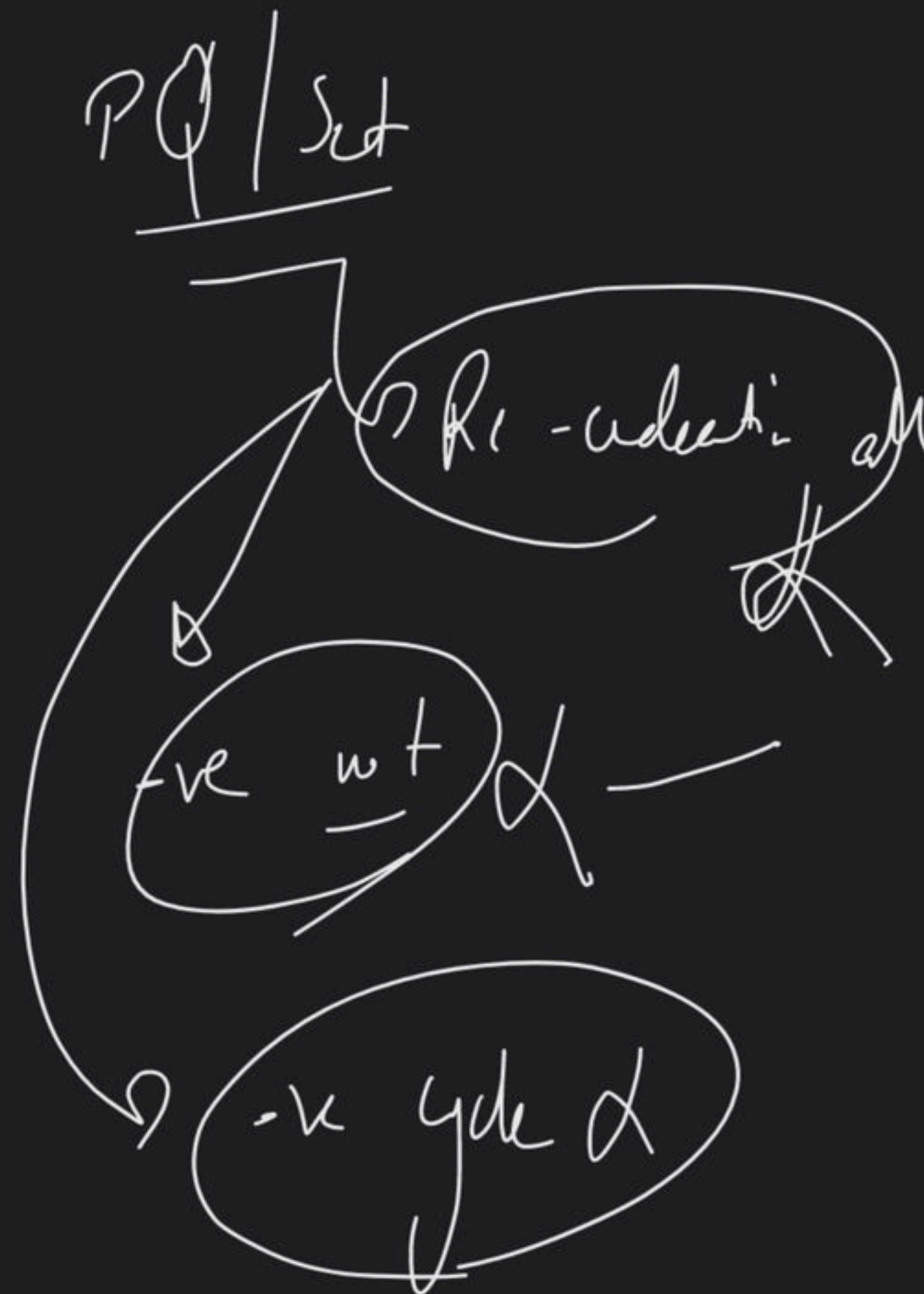
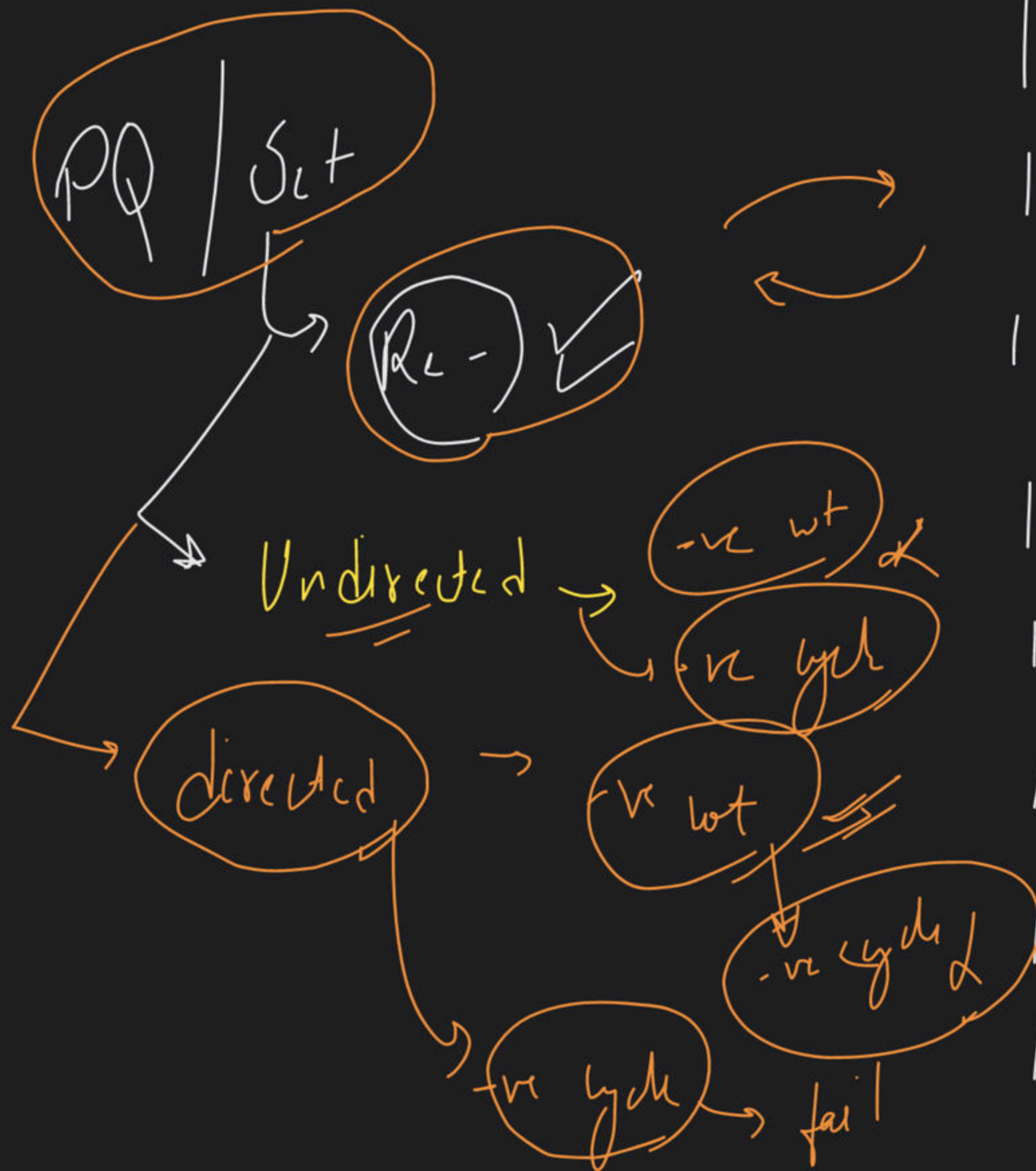


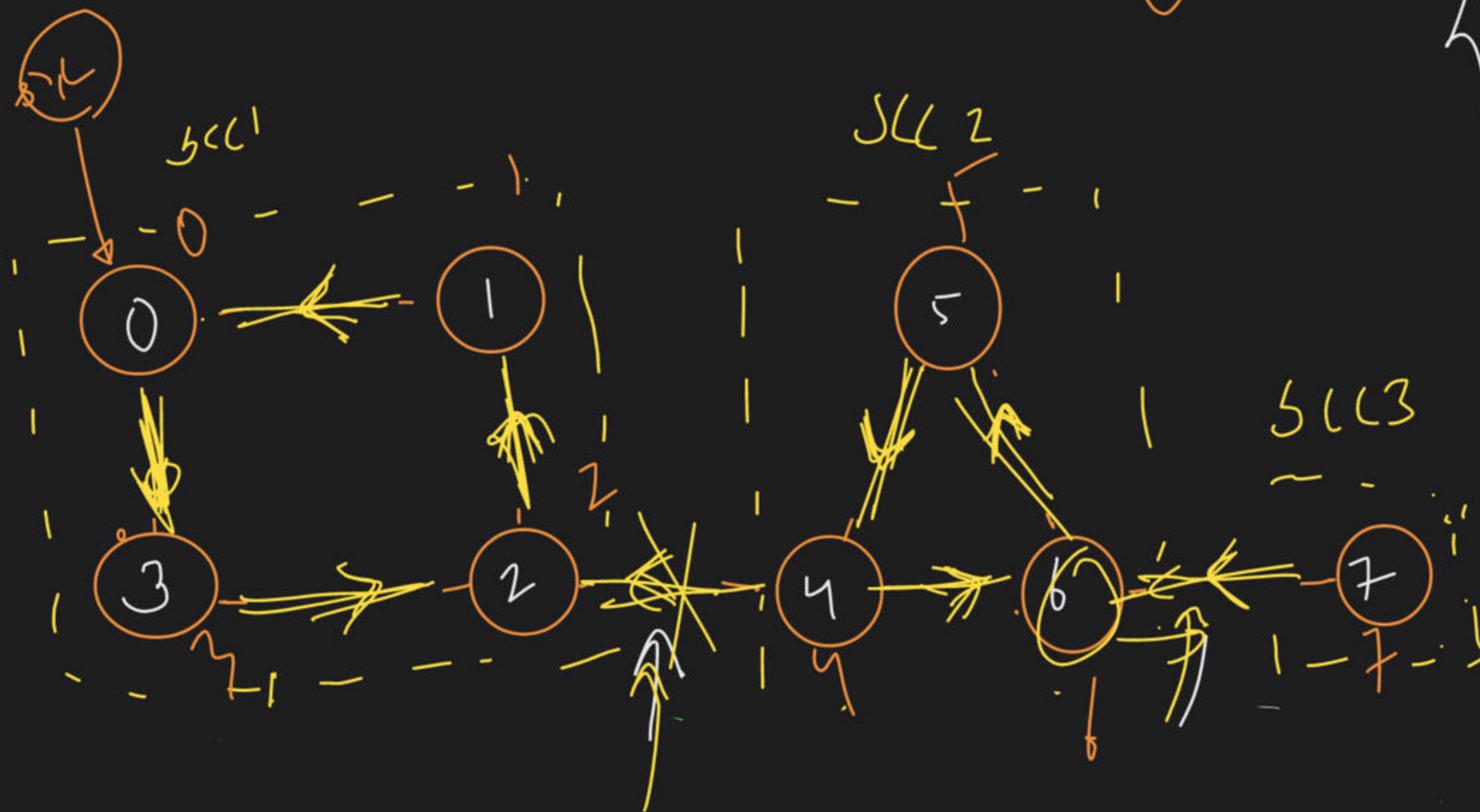
Graph Class - 8

Special class

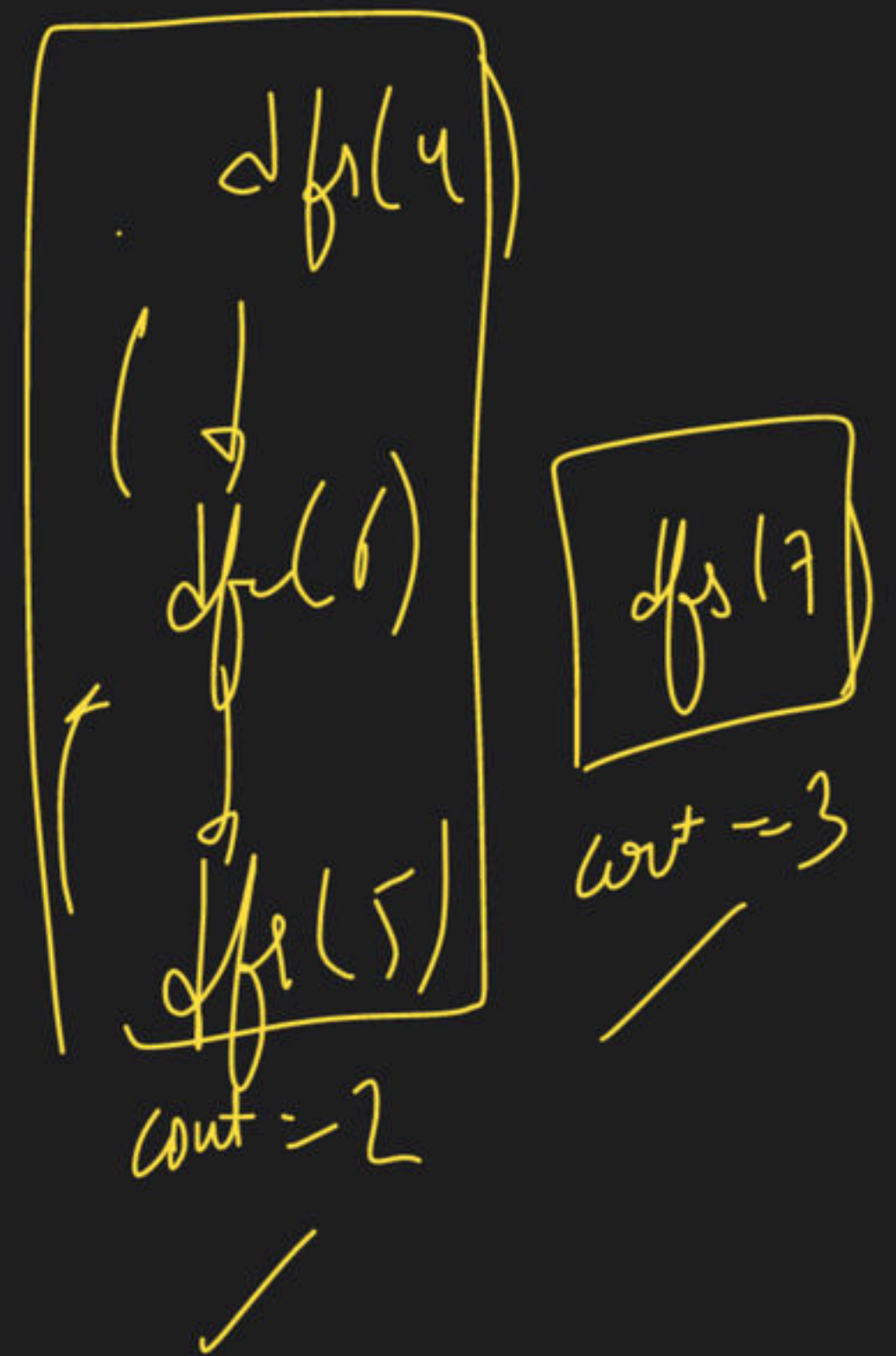
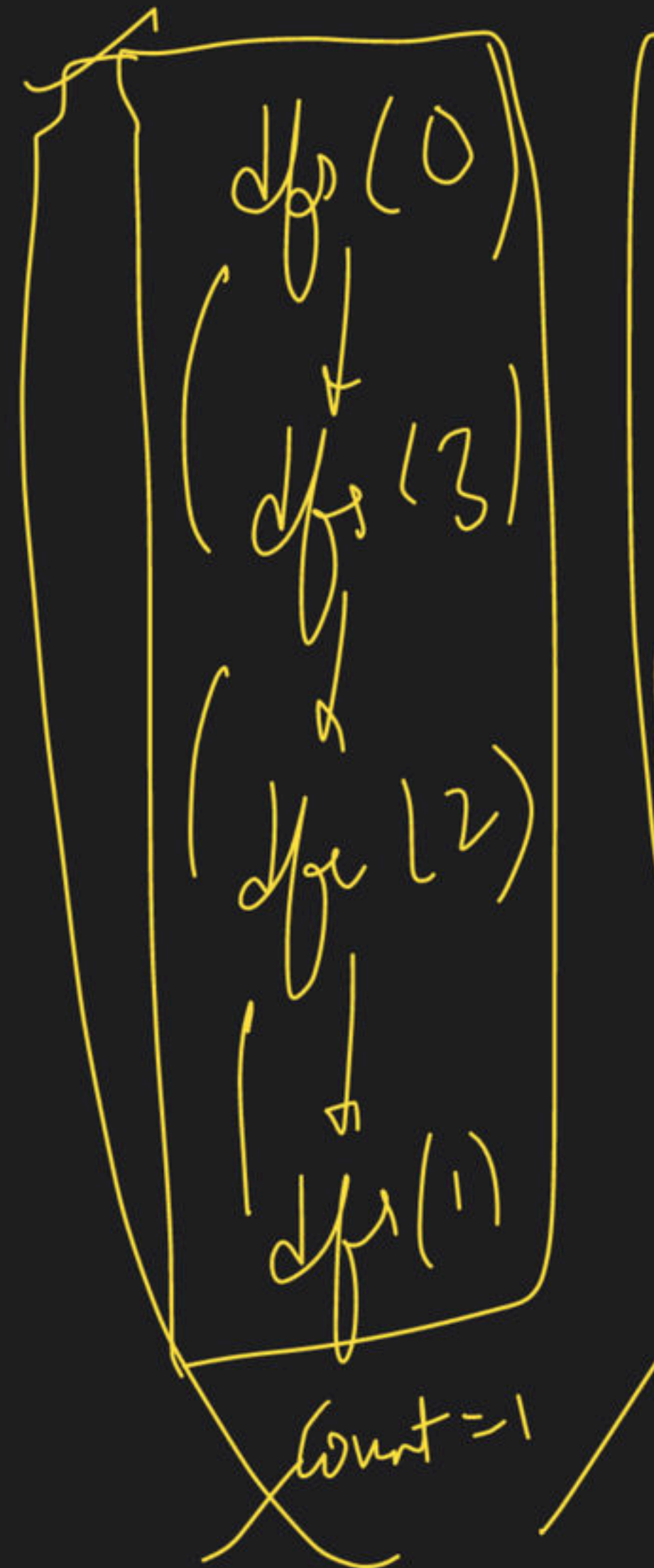




Strongly Connected Components

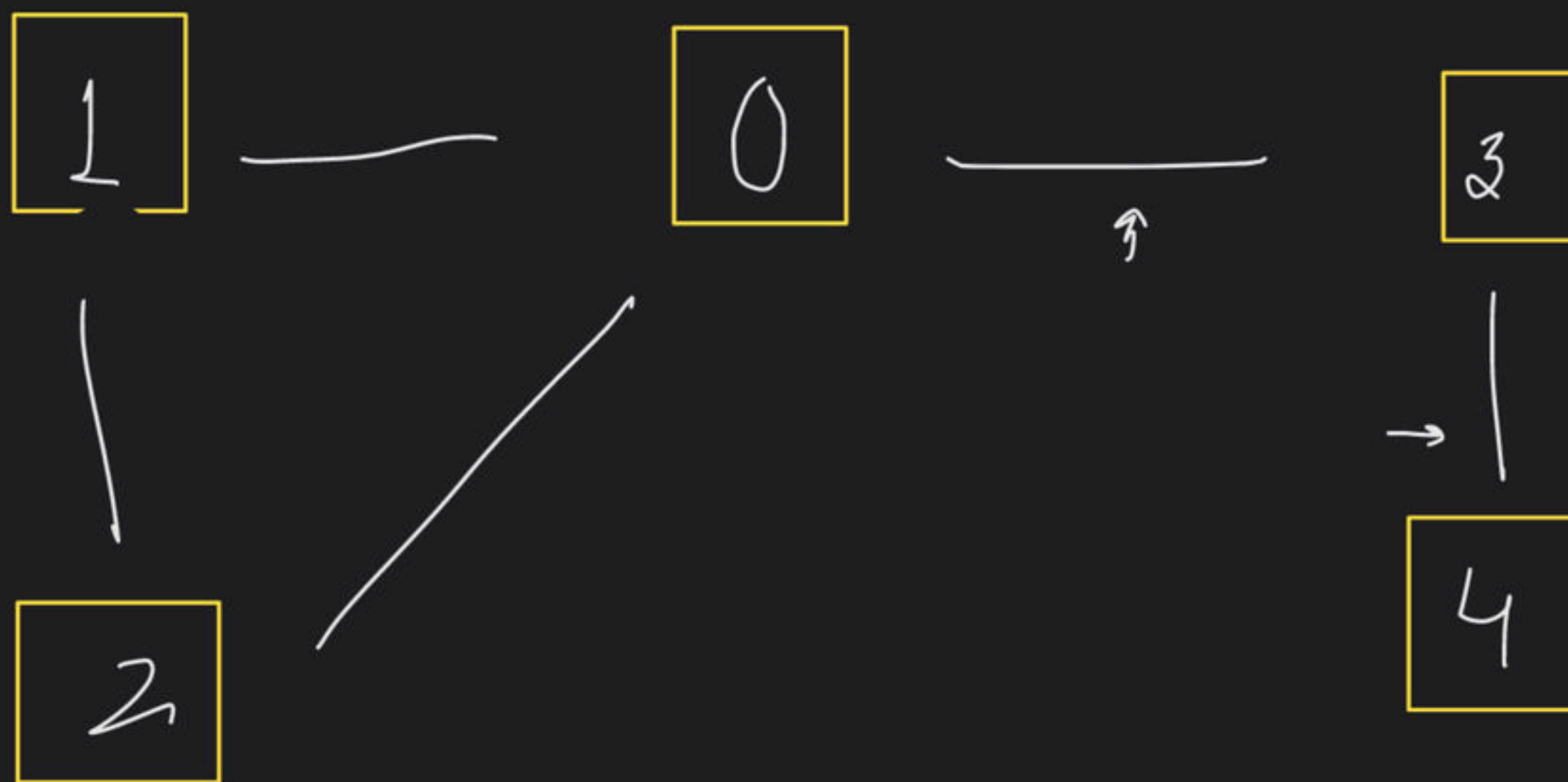


- ① find Ordering in which we should traverse the graph
- ② reverse the edges
all
- ③ Count components



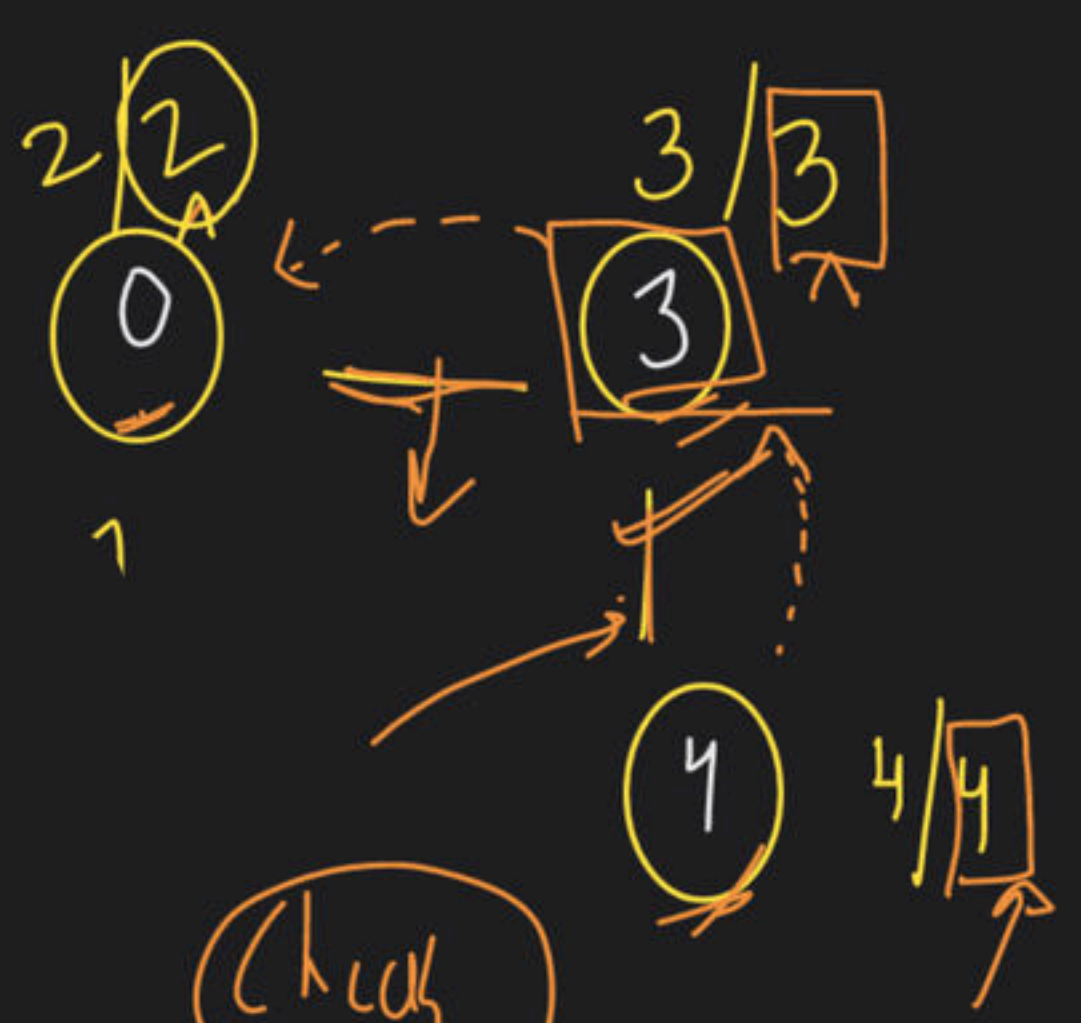
~~Q~~ Bridge in a Graph:-

starty | finishing time



Back
force

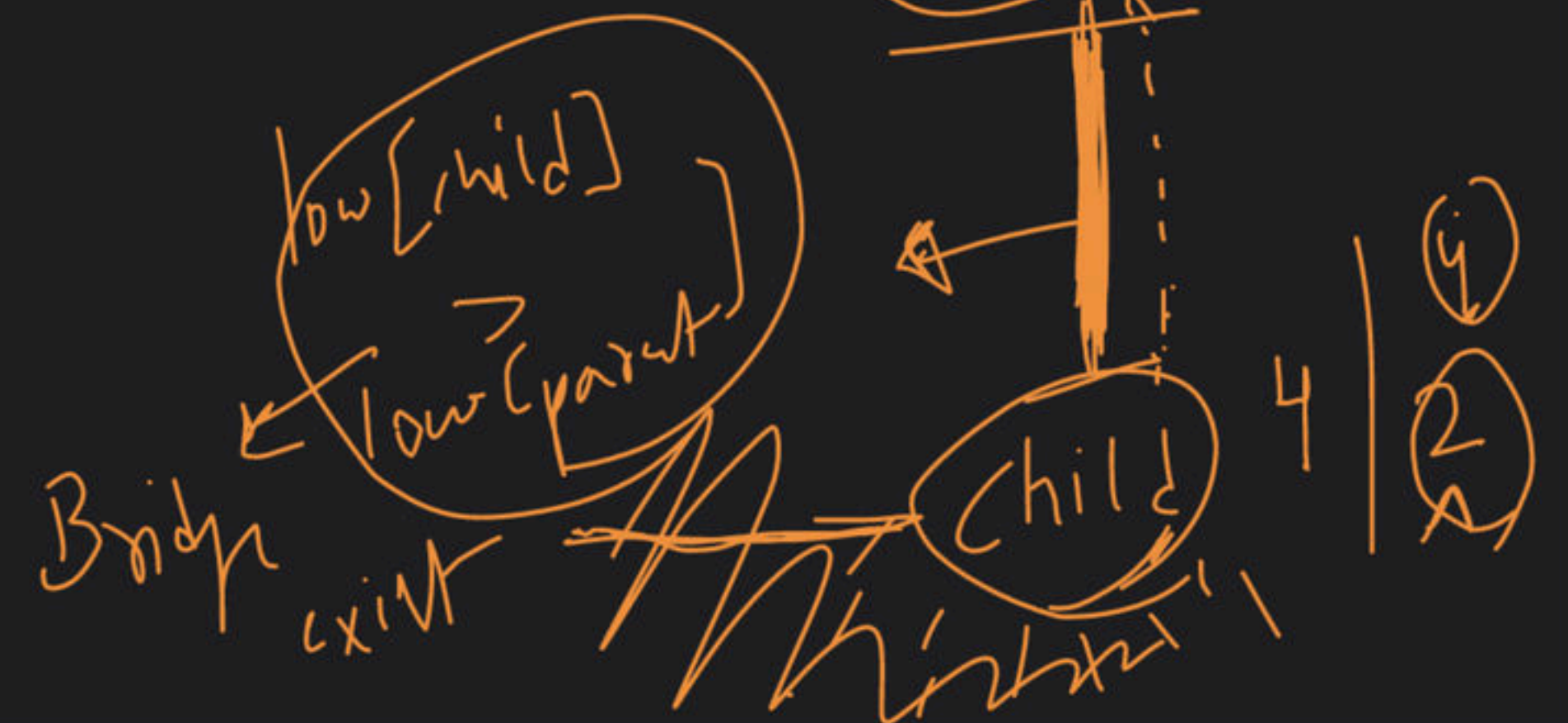
Tarjan's
Algo



\leftarrow as
 $\text{tin} \rightarrow$ insertion time
 $\text{low} \rightarrow$ lowest insertion time

path parent
 low visited
 low update

Check if bridge
 $\text{low}[\text{child}] < \text{low}[\text{parent}] \rightarrow \text{child} \rightarrow$
 this is a bridge
 low update
 parent



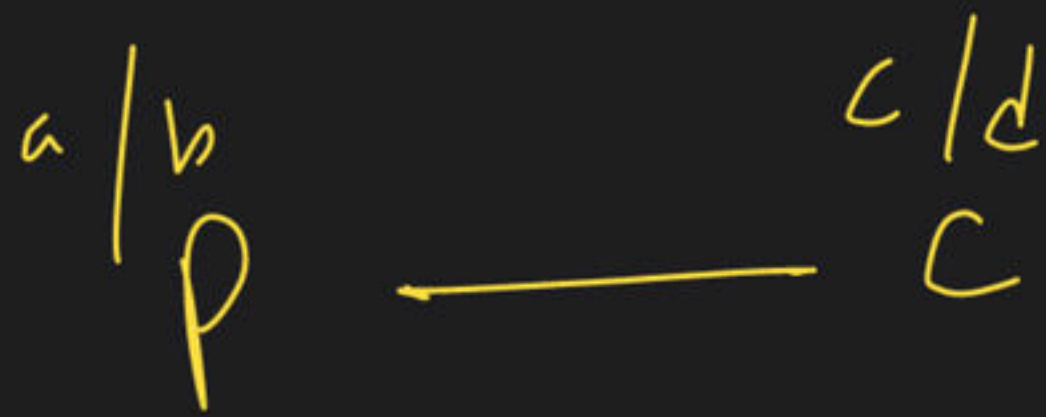
① $\text{low}[\text{child}] > \text{low}[\text{parent}]$ → wrong

② ↓ → Absent

③ ~~if nbr == parent → ignore~~

④ !vis[nbr] → ~~dfs call~~
→ ~~low update~~
→ check for bridge

⑤ ~~Already vis → low update~~
 ~~$\text{low}[\text{nbr}] = \min(\text{low}[\text{nbr}], \text{low}[\text{nbr}])$~~

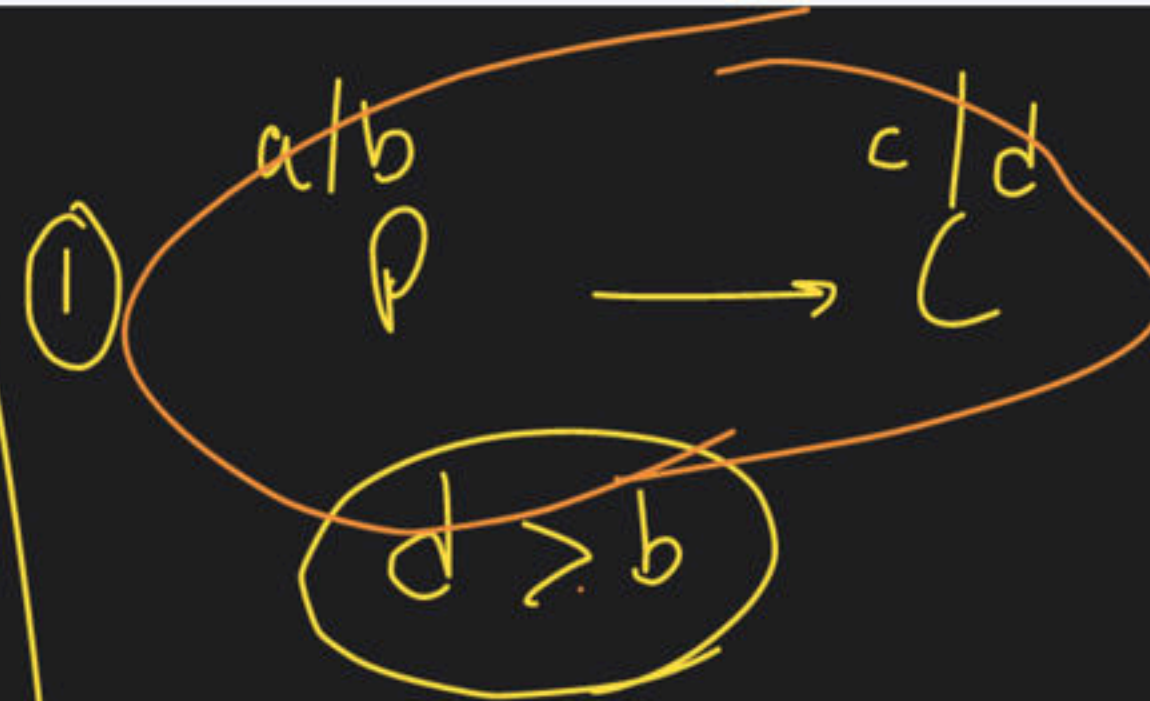


$$d > b \quad \times$$

\hookrightarrow 1, $(low[child] > low[parent])$

\hookrightarrow c tak kahi or me
bhi pohuta jaa
sakte h

\hookrightarrow ~~$c-p$~~ \rightarrow Not a
bridge



$low[child] > low[parent]$

\hookrightarrow c tak sirf p se
pohuta sakte h

\hookrightarrow $c-p$ is a
bridge

for (nbr → adj[i])

No comparison to ~~parent~~ ^{parent} parent

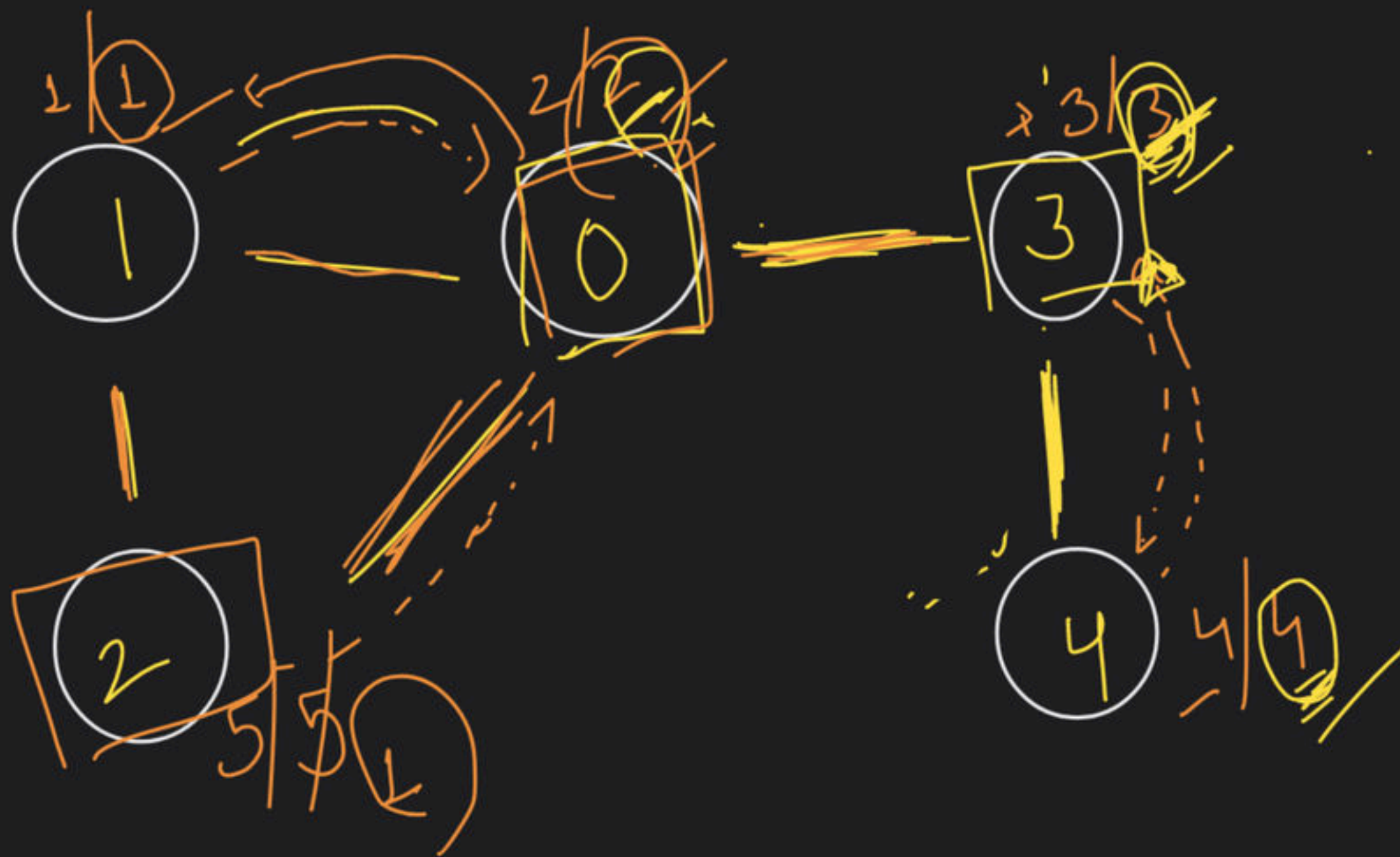
nbr → !visited

call dfs()

if (nbr → visited) {
 nbr → not parent
}

low → update

low update
check for bridge



timely-1

$$\gg 2\pi n$$

$4 > 3 \rightarrow \text{True}$

wapal Jane

- low update
- check for Bridge

$\text{low}[c] > \text{low}[p]$



$\text{low}[c] > \text{tin}[p]$



