

BCC Biconnected Component



Topological Sort

DAG : Directed Acyclic Graph 방향이 있고 사이클이 없는 그래프

모든 노드를 Sort 해서 한 줄로 세워서 오른쪽으로만 가게 정렬

사이클이 없으면 \leftrightarrow Sort 가능

$O(n \cdot m)$

그래프 중에 indegree가 0인 노드가 있다. \rightarrow 없다면 계속 따라가면 사이클이 생김.

indegree가 0인 노드를 본다. \rightarrow 남은 그래프는 DAG.

$O(n+m + n+m)$

Event Queue 사용.

Indegree가 0인 노드 넣기.

거의 노드와 연결된 애들리 indegree 계산.

다시 indegree가 0인 노드 다시 넣기

더 빠르게

DFS를 하면서 post #를 계산.

post #가 감소하는 순서로 출력

post #가 더 큼.



실제 사용

reverse 그래프로 DFS 하기

Post # 오름차순으로, 또는 post #가 계산되는 순서대로 sort 된다.

일반

Shortest path : Dijkstra

Longest path : NP-complete

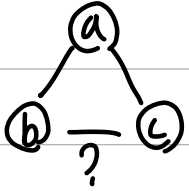
bag

순서대로 min 계산

max 계산

무작정거방

삼각형을 이루는 노드 중 degree가 2인 노드 제거. 최소 점점만 남기기.



Event Queue 활용.

제거하는 순서에 따라 최종그래프의 모양의 개수? 그리고 그 노드 개수는 항상 같은가?

제거하는 개수를 센다.

Seq a

Seq b

del a_1

del b_1

$\rightarrow a_1 = b_1$ ok.

a_2

b_2

$a_1 = b_k$ b_k 를 먼저 지우면 ok.

a_3

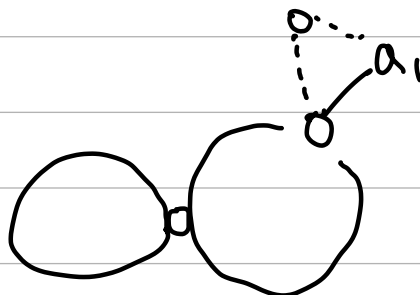
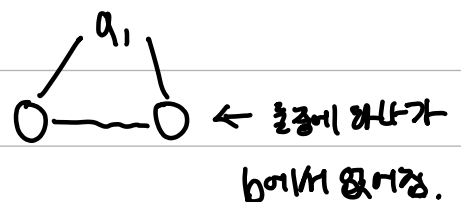
b_3

$a_1 \neq b_k$ b에서 a_1 를 못지우는 상태.

\vdots

\vdots

왜?



Cut Vertex가 되면
끊어지거나 생기지 않음.

다 풀었을 때 1개 짜리로 바뀌어도 $a_1 \neq b_k$ 가 될 수 있음.

즉, triangle tree 여야 함.