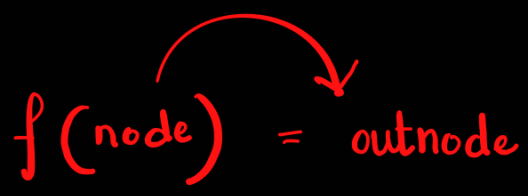
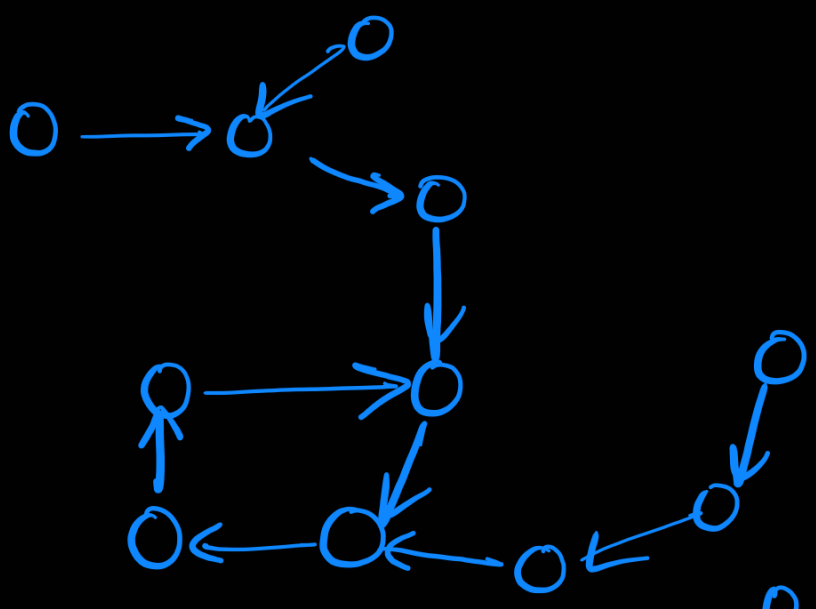
Functional Graph & SCC

Graph → all nodes outdeg = 1



imp to visualize

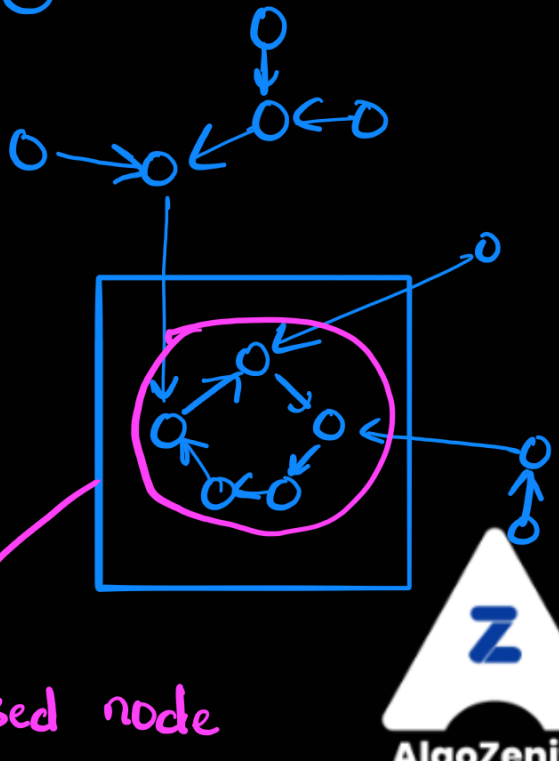


tree like

chain

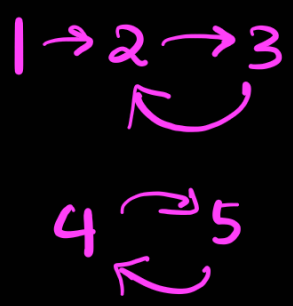
central cycle

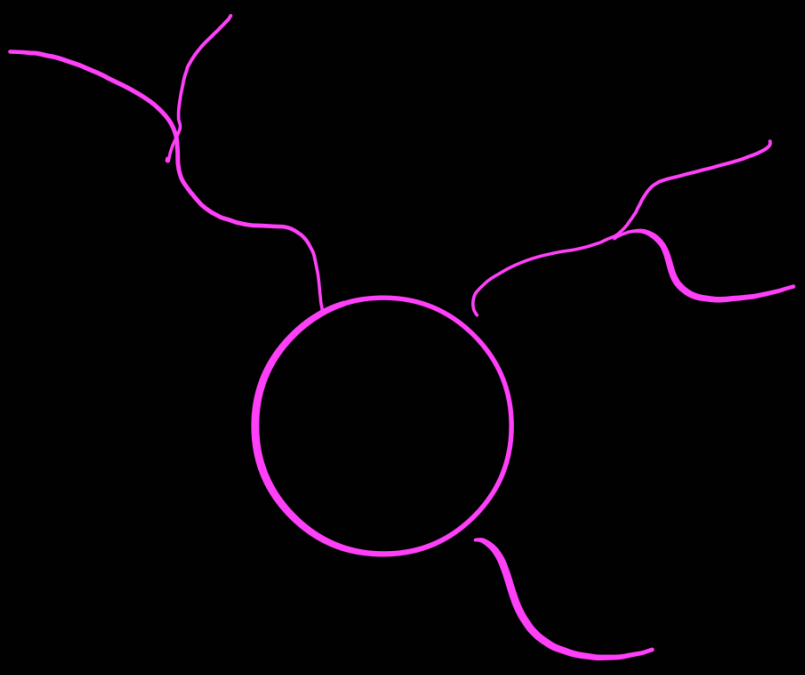
DAG to cycle



AlgoZenith

in each component, single cycle





outdegree always 1
forces a cycle to be present

Be aware of structure
Draw picture
Problems easy

AlgoZenith

Chef and Round Run

Read problems statements in Mandarin Chinese, Russian and Vietnamese as well.

Chef cooks nice receipes in the cafeteria of his company. The cafe contains **N** boxes with food enumerated from **1** to **N** and are placed in a circle in clocwise order (boxes **1** and **N** are adjacent). Each box has unlimited amount of food with a tastyness level of A_i. Chef invented a definition of a magic box!

- Chef picks a box **i** and stays in front of it.
- Now Chef eats food from box **i** and **skips** next A_i boxes.
- Now Chef is staying at some other (probably even the same!) box and repeats.
- Box **i** is a magic box if at some point of such game started from box **i**, Chef will find himself staying in front of it again.

When Chef came home, Chef's dog Tommy asked him about how many magic boxes were in the cafe? Help Chef to in finding that!

Input

The first line of the input contains an integer **T** denoting the number of test cases. The description of **T** test cases follows.

The first line of each test case contains a single integer **N** denoting the number of boxes.

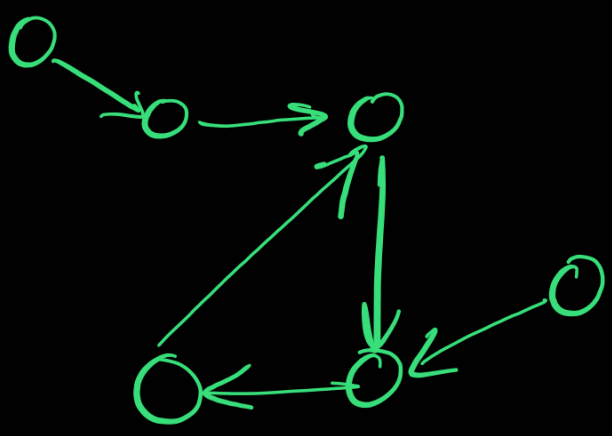
The second line contains **N** space-separated integers A_1, A_2, ..., A_N denoting the tastyness levels of each box.

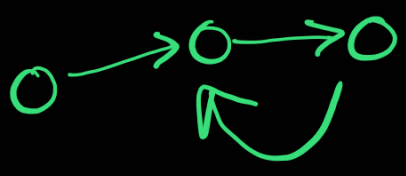
Output

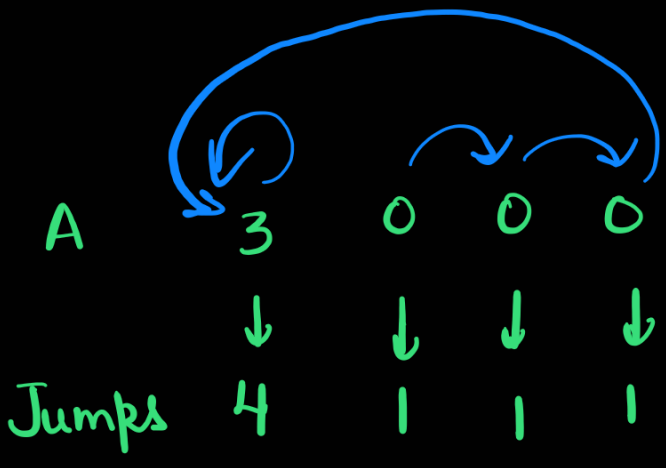
For each test case, output a single line containing number of magical boxes.

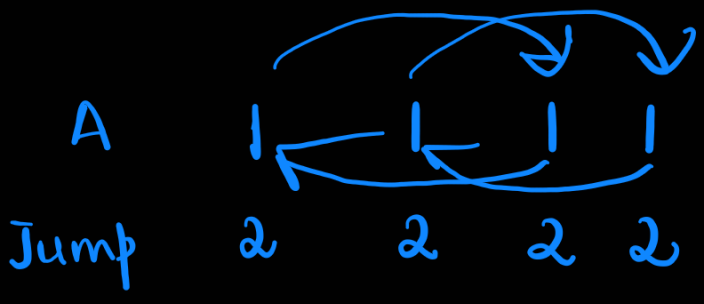
Constraints

- $1 \le$ sum of all **N** over all the test cases in a single test file $\le 10^6$
- $0 \le A_i \le 10^9$

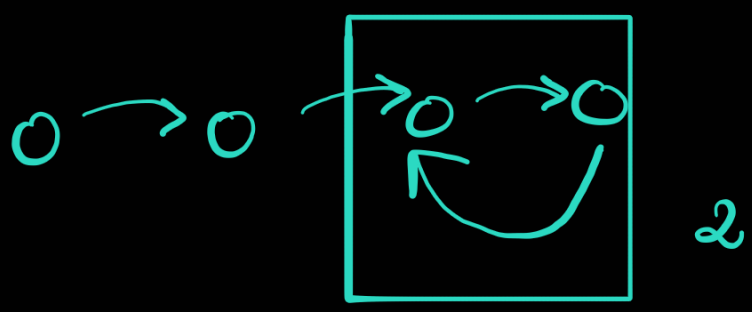






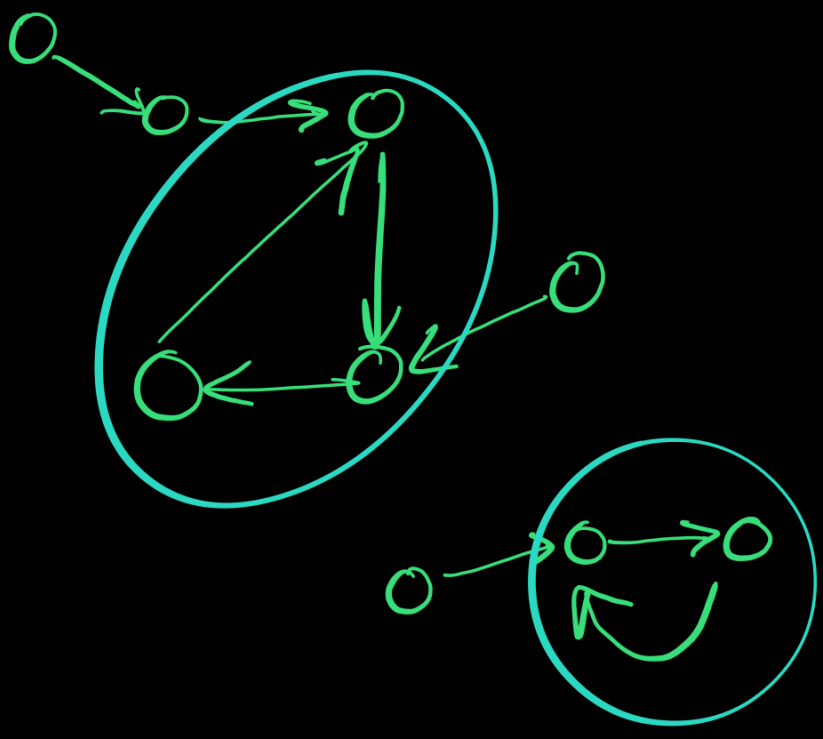


4 0 0 0 2



AlgoZenith

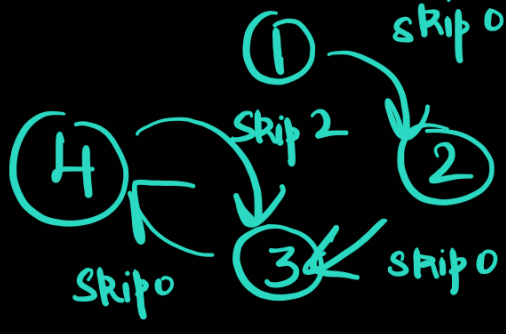
No. of nodes part of cycle is answer



5

4

0 0 0 2



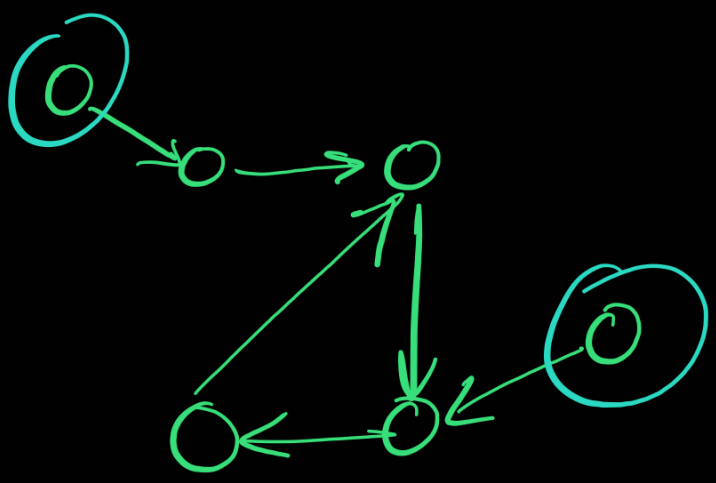
Top ordering does not help with cycle detection

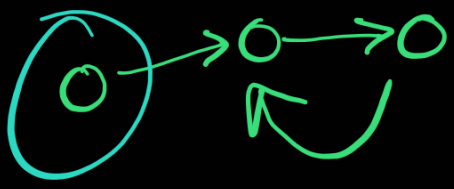
AlgoZenith

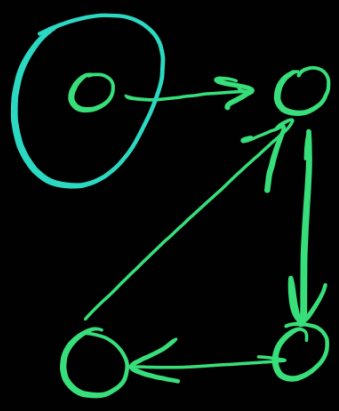
Can only help detect if cycle present or not

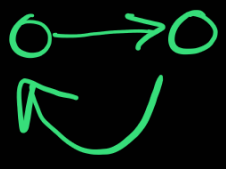
General cycle — Topo → if cycle exists or not

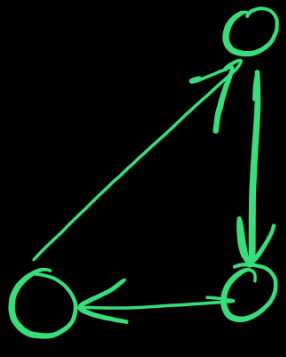
But in functional graph we can use keep deleting nodes with indegree = 0

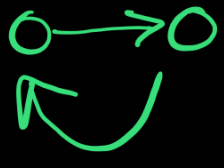










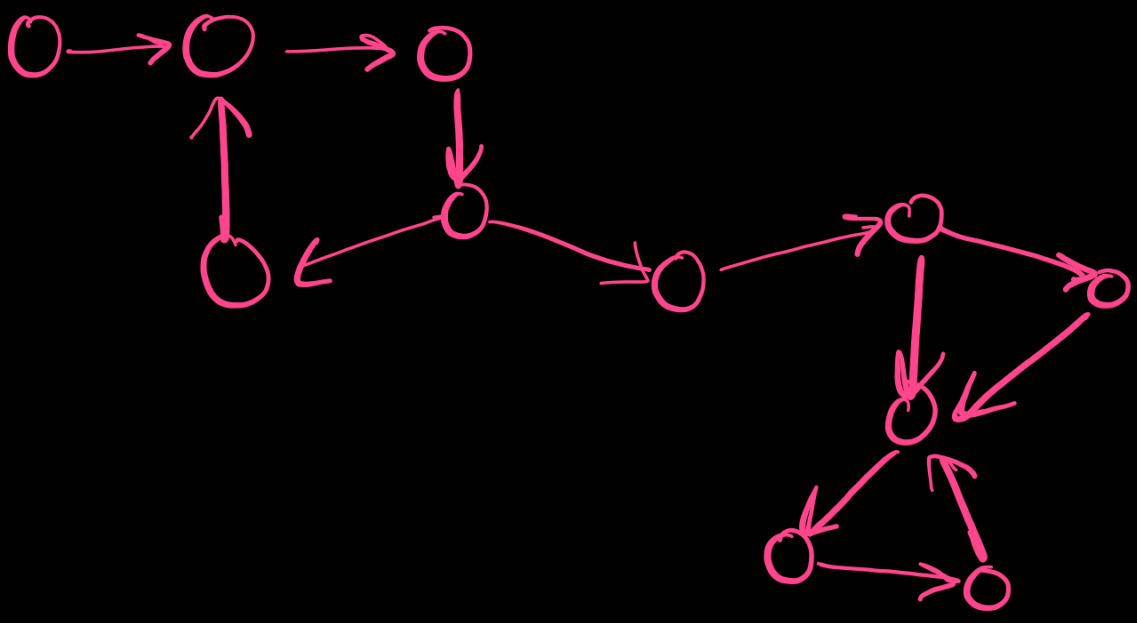


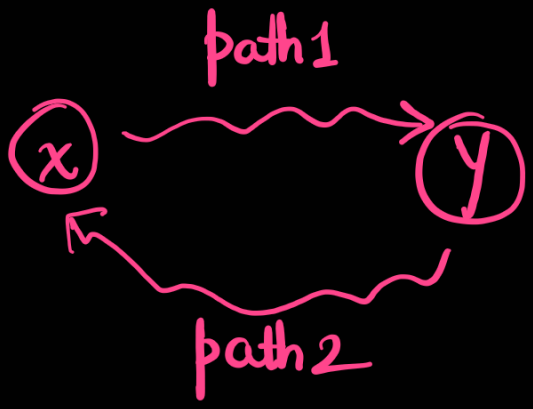
Strongly Connected Components

Tarjan Algo

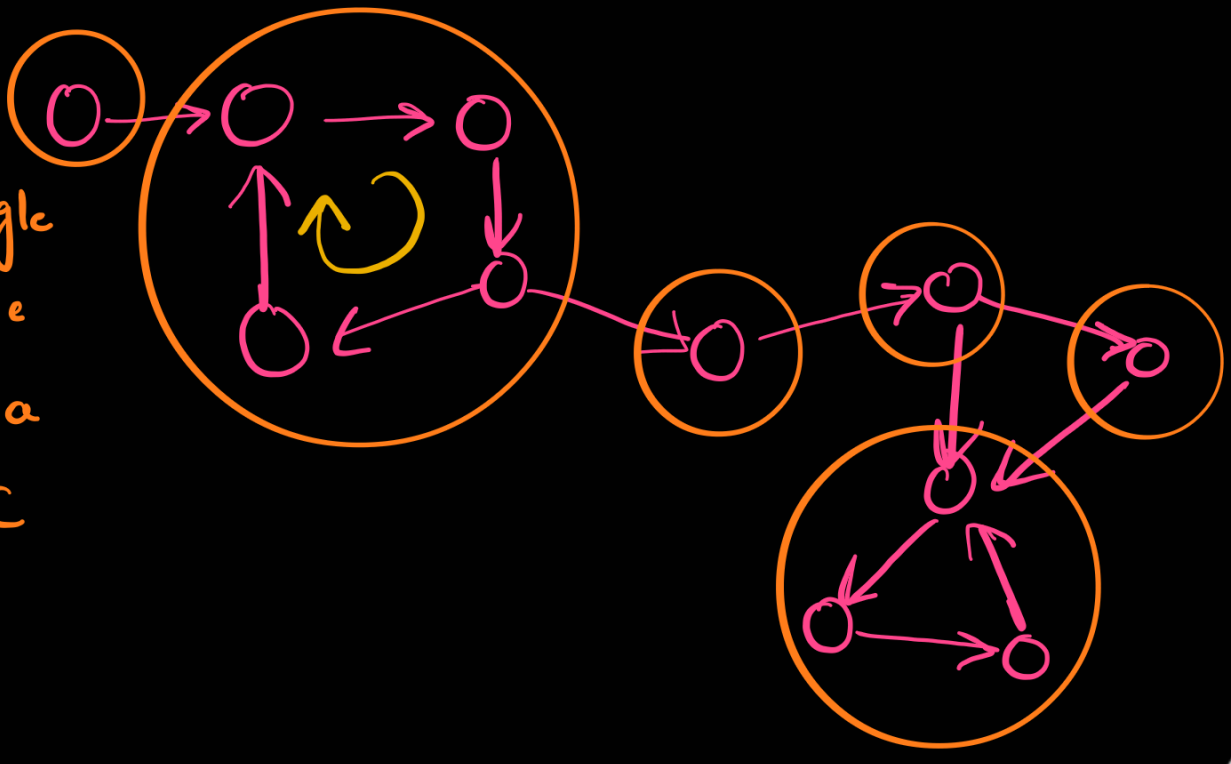
General directed graph

AlgoZenith





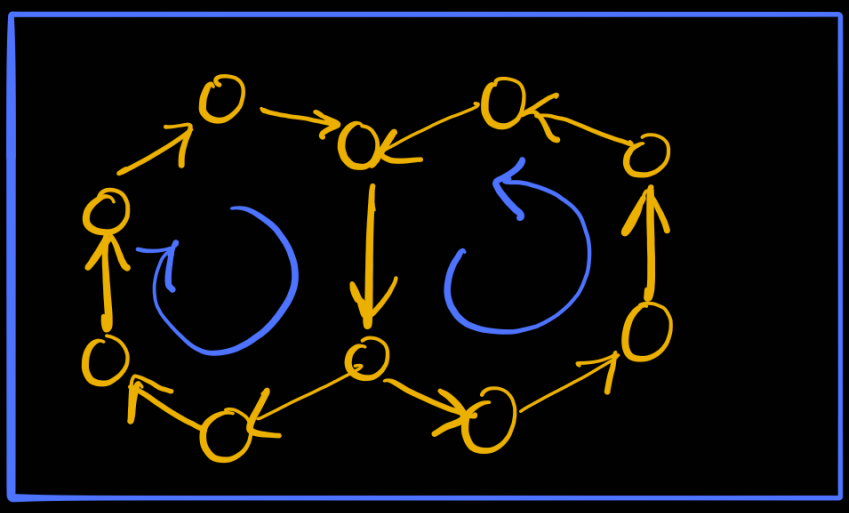
x, y are in same CC



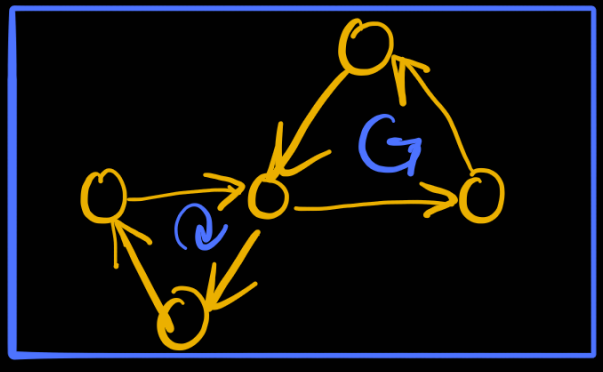
Single node is a CC

6 SCCs

A SCC is a group of nodes where every node reaches each other

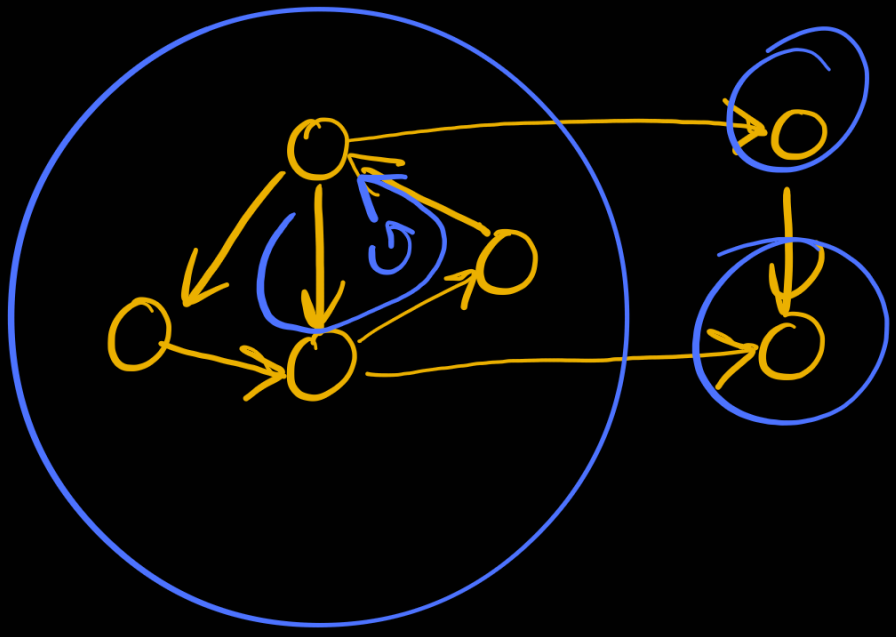


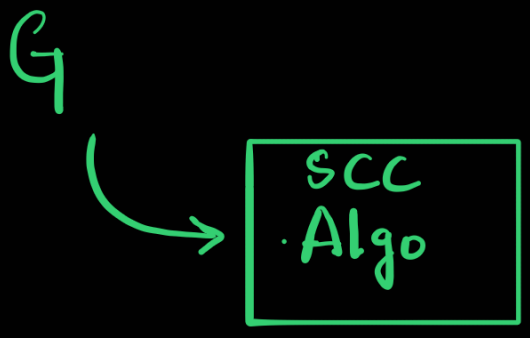
1 SCC



1 SCC

because of common nodes, they get merged

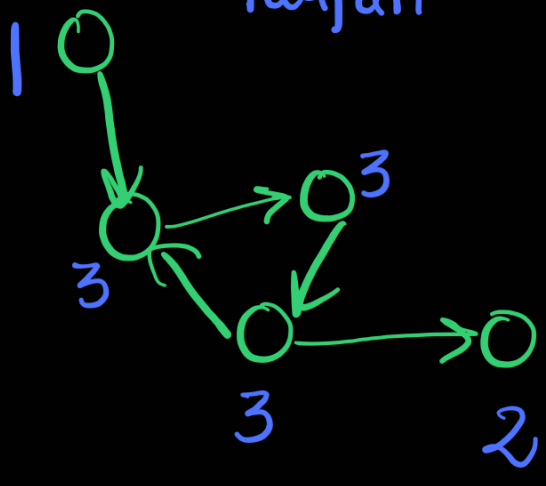




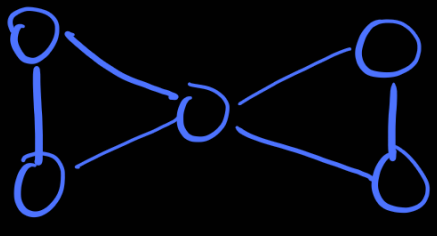
→ Label nodes with SCC numbers

Kosaraju

Tarjan → Low Linking
- ↳ Articulation
- ↳ Bridges
- ↳ SCC
- ↳ Biconnected comp

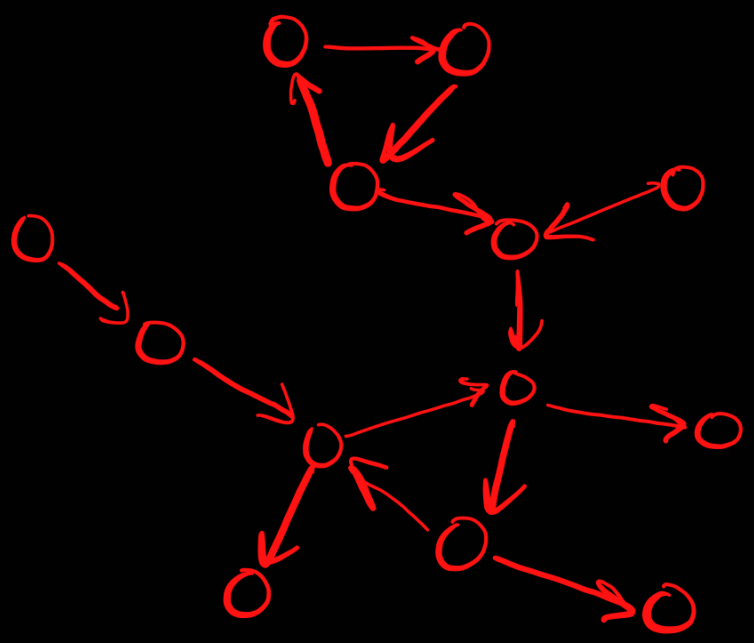


After deleting node find no. of CC



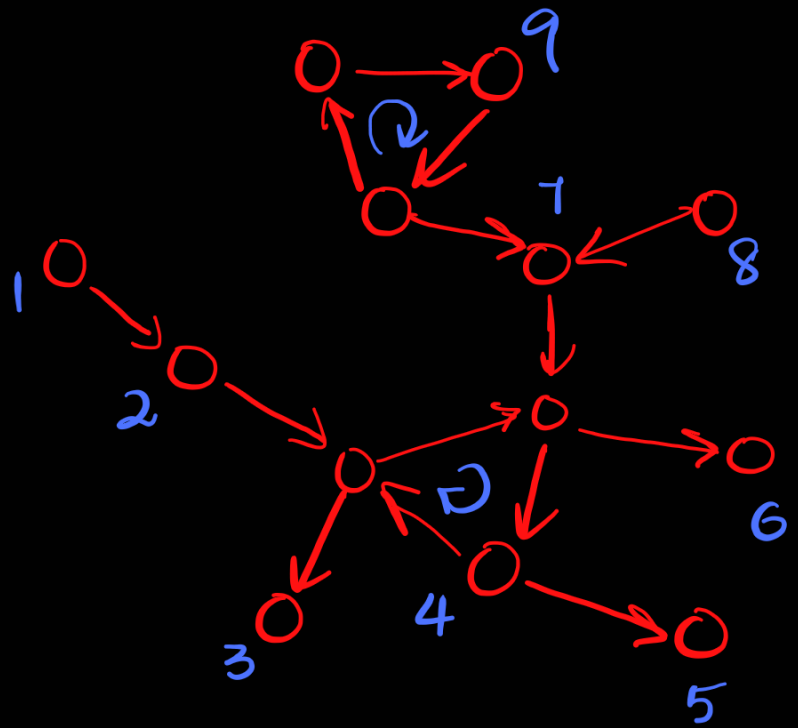
sum of values of components

① College Lab

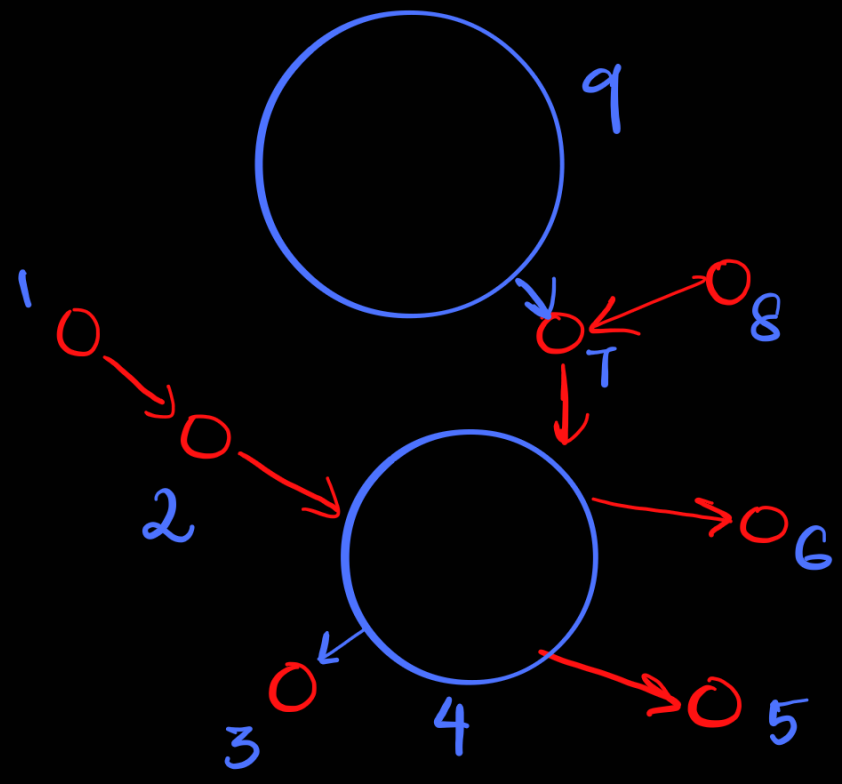


Virus → infection spreads

Find min no. of nodes to infect so that whole graph is infected



AlgoZenith



Compress / Condense SCC

No cycles in condensed SCC graph

Min nodes to infect → nodes with indegree 0

any other nodes with in >0 will be infected by someone else

2

It happens every year

"Chotu and BornConfused are friends."

"Live for each other,die for each other."

There are N restaurant in a city .Each restaurant is connected to every other by a pair of directed road.

Like every year it's time coming when BornConfused will meet his new crush. She do not like cheap restaurants.Out of N restaurants at least one is not cheap. Chotu knows everything. He knows that which restaurants are cheap for her and which are not. Whereas BornConfused can be at any restaurant in the city. Chotu heard a news that some roads will be blocked on the day of their meeting and got the list of M unblocked roads. Chotu does not want to see his friend BornConfused to be sad. So, he decided to find out minimum number of roads which should be added to list of M unblocked roads such that BornConfused surely can reach to his girlfriend(he thinks ;P) in any situation.

Your task is to find out minimum number of roads which should be added to list of unblocked roads, while Chotu is busy talking to Amit about the problem.

It is guaranteed that all roads, from and to, cheap restaurants are blocked.

###Input:

- First line contain two integers N and M, number of restaurants and number of unblocked roads
- Second line contains N integers denoting that ith restaurant is cheap"type 0" or not "type 1".
- Next M lines contains two integers u and v. Denoting a directed road from restaurant u to restaurant v.

###Output: Required single integer.

###Constraints

- $1 \leq N, M \leq 200000$
- $0 \leq TypeOfRestaurant \leq 1$
- $1 \leq u, v \leq N$
- It is guaranteed that all roads, from and to, type "0"(cheap) restaurants are blocked.

###Sample Input: 4 1 1 1 0 0 1 2

###Sample Output: 3

###Explanation: List of roads which are added to list of unblocked roads: 4 → 3 3 → 2 2 → 1

Calculate SCC first

All nodes of same comp. are given a number

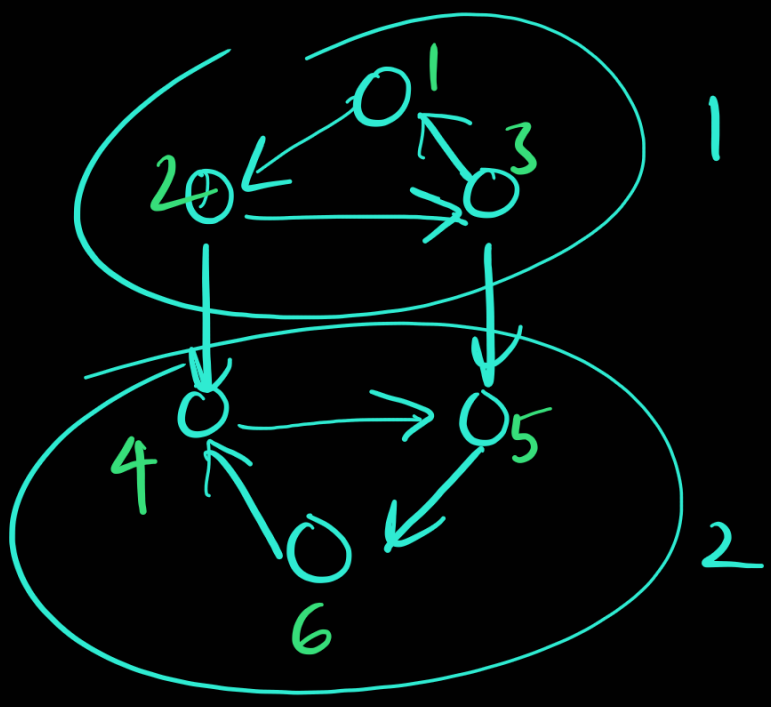
$O(V+M)$

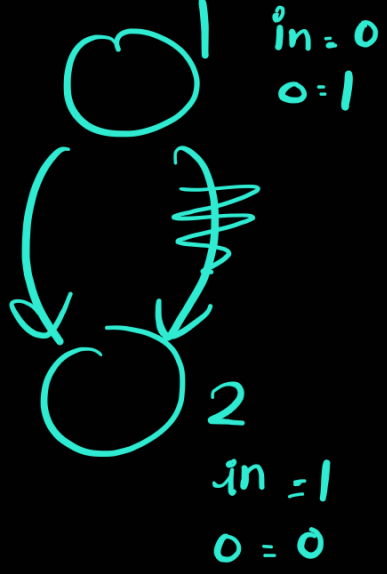
Construct the DAG for a condensed graph

i ⟶ V

outdeg — indeg

scc ++ — scc --





use a set to avoid duplicates

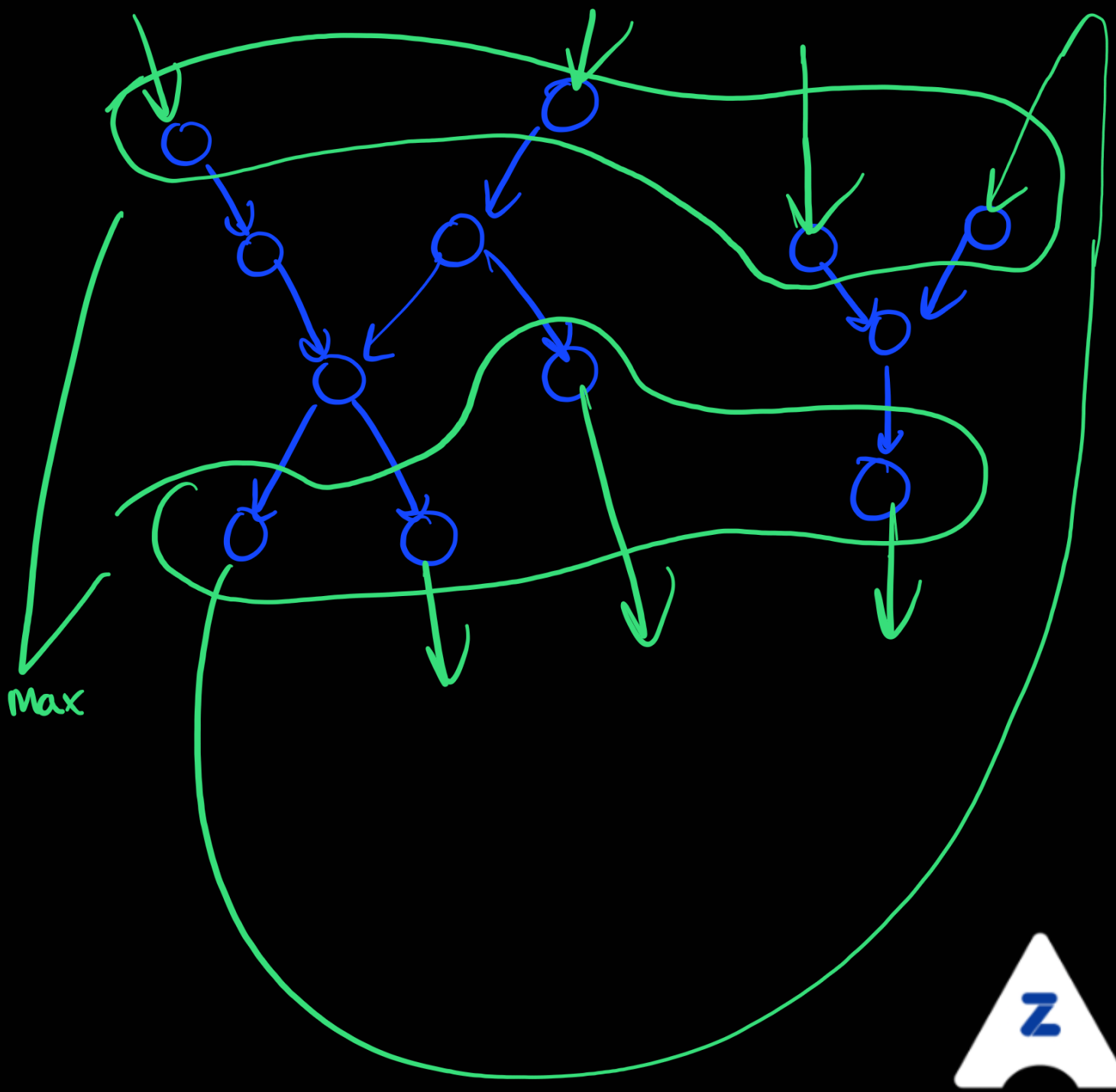
getconut

1 : [1 2 3]

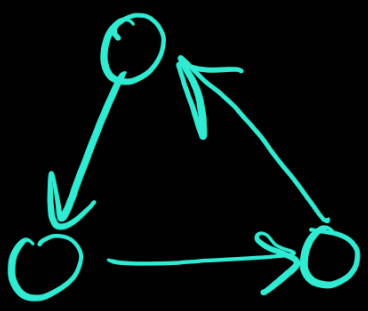
2 : [4 5 6]

find SCC nodes of in = 0 / out = 0

ans max of indeg / outdeg



AlgoZenith



Condense

Answer