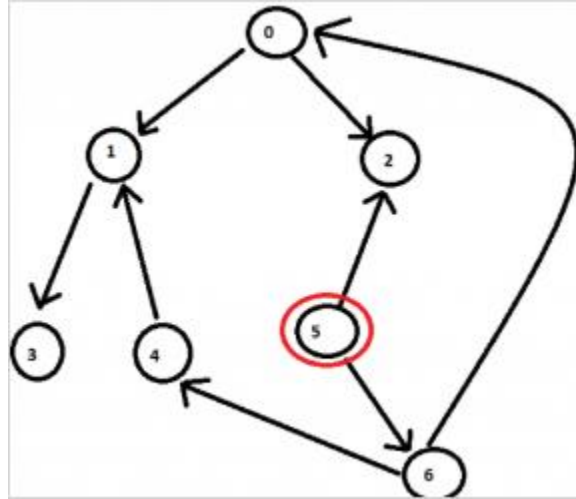


LAB Assignment 8
CS205
Topic: Elementary Graph Search
Lab Assignment: a,b Home Assignment : c

- a. In a directed graph, a vertex will be called core vertex if we can reach all other nodes from that vertex. For example in the below graph, node 5 is a core vertex. Please write an **efficient** algorithm to find out core vertex if it is there in the graph otherwise output of your algorithm will be “None of the vertices is core vertex”.



Upload Ass8a.c

- b. There are many islands that are connected by one-way bridges, that is, if a bridge connects islands a and b , then you can only use the bridge to go from a to b but you cannot travel back by using the same. If you are on island a , then you select (uniformly and randomly) one of the islands that are directly reachable from a through the one-way bridge and move to that island. You are stuck on an island if you cannot move any further. It is guaranteed that after leaving any island it is not possible to come back to that island.

Find the island that you are most likely to get stuck on. Two islands are considered equally likely if the absolute difference of the probabilities of ending up on them is $\leq 10^{-9}$.

Input format

- First line: Three integers n (the number of islands), m (the number of one-way bridges), and r (the index of the island you are initially on)
- Next m lines: Two integers u_i and v_i representing a one-way bridge from island u_i to v_i .

Output format

Print the index of the island that you are most likely to get stuck on. If there are multiple islands, then print them in the increasing order of indices (space separated values in a single line).

Input Constraints

$1 \leq n \leq 200000$

$1 \leq m \leq 500000$

$1 \leq u_i, v_i, r \leq n$

Sample input	Sample output
5 7 1 1 2 1 3 1 4 1 5 2 4 2 5 3 4	4

Explanation

There are two islands on which you could get stuck- 4 and 5 with 4 being more probable.

Upload Ass8b.c

- c. Given an array of pairs A of size N which represents a game situation where the first player wins against the second player. Given multiple queries, each query contains two numbers, the task is to determine which one of them will win if they compete with each other.

NOTE:

If A wins over B and B wins over C, then A will always win over C.

If A wins over B and A wins over C, if there is a match against B and C and if we couldn't determine the winner then the player with smaller number wins

Examples:

Input : `arr[] = {{0, 1}, {0, 2}, {0, 3}, {1, 5}, {2, 5}, {3, 4}, {4, 5}, {6, 0}}`

`query[] = {{3, 5}, {1, 2}}`

Output : 5 1

Explanation : 4 wins over 3 and 5 wins over 4. So, 5 is the winner in the first match.

We can't determine the winner between 1 and 2. So, the player with a smaller number is the winner i.e., 1

Input : `arr[] = {{0, 1}, {0, 2}, {0, 3}, {1, 5}, {2, 5}, {3, 4}, {4, 5}, {6, 0}}`

`query[] = {{0, 5}, {0, 6}}`

Output : 5 0

Upload Ass8c.c