

CSE221 Assignment 06 Spring 2025

A. Advising

1 second, 1024 megabytes

In this problem, there are  $N$  courses in the curriculum and  $M$  requirements of the form "Course  $A$  has to be completed before course  $B$ ".

Your task is to find an order in which you can complete the courses. If there are multiple valid order, you may print any of them. If no such sequence exists, then print  $-1$ .

Input

The first line contains two integers  $N, M$  ( $1 \leq N \leq 2 \times 10^5, 1 \leq M \leq 3 \times 10^5$ ) — the number of courses and total requirements.

The next  $M$  lines will contain two integers  $A_i, B_i (1 \leq A_i, B_i \leq N)$  — Course  $A$  has to be completed before course  $B$ .

Output

Print an order in which you can complete the courses. Please note, that there could be multiple correct sequences. You can print any valid order that includes all the courses.

If there is no valid sequence, print  $-1$ .

input
5 4 2 4 2 5 4 3 1 5
output
2 4 3 1 5

input
8 8 6 4 6 2 4 2 2 1 1 7 7 5 5 8 8 3
output
6 4 2 1 7 5 8 3

input
2 1 1 2
output
1 2

input
4 6 1 2 1 3 4 1 2 3 2 4 4 3

output
-1

B. A Football Match

2 seconds, 1024 megabytes

There is an intense football match going on between Robots and Humans. However, things aren't as simple as they seem — the Robots have disguised themselves to look exactly like Humans! From the outside, it's impossible to tell who is a Robot and who is a Human.

The audience know only one important information — the Robots tackles only the Humans, and the Humans tackles only the Robots.

Now, you are given a list of tackles, each involving two players. Based on this information, find the maximum possible number of Robots or Humans.

Input

The first line contains two integers  $N$  and  $M$  ( $1 \leq N \leq 2 \times 10^5, 1 \leq M \leq 3 \times 10^5$ ) — the number of players in the match and the number of tackles occurred during the match respectively.

The next  $M$  lines will contain two integers  $u_i, v_i (1 \leq u_i, v_i \leq N)$  — player  $u_i$  tackled player  $v_i$ . Each tackle between two players will be reported at most once.

Output

Print the maximum possible number of Robots or Humans.

input
5 6 3 4 3 2 5 4 5 2 4 1 1 2
output
3

input
5 4 4 3 1 3 3 2 3 5
output
4

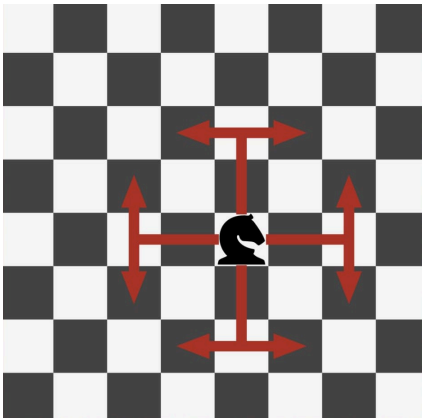
input
4 1 1 3
output
3

input
6 6 1 3 1 4 3 6 4 6 4 5 6 2
output
3

C. The Knight of Königsberg

1 second, 256 megabytes

You are given an  $N \times N$  chessboard and the initial position  $(x_1, y_1)$  of a Knight piece. You need to find the minimum number of moves the Knight needs to reach the target position  $(x_2, y_2)$ . If it is not possible to reach the target, print  $-1$ .



Moves of a Knight in Chess

The Knight can move one step in any of the 8 possible directions as shown in the picture.

Input

The first line contains an integer  $(1 \leq N \leq 2 \times 10^3)$  — the size of the chessboard.

The second line contains four integers  $(1 \leq x_1, y_1, x_2, y_2 \leq N)$  — the initial position  $(x_1, y_1)$  and the target position  $(x_2, y_2)$  of the Knight on the chessboard.

Output

Print the minimum number of moves the Knight needs to reach the target position. If it's not possible, print  $-1$ .

input
3 1 2 1 3
output
3

input
3 1 1 2 2
output
-1

input
10 8 4 3 1
output
4

D. Easy Tree Queries

1 second, 1024 megabytes

There is a tree with  $N$  nodes. The tree is rooted at a given node  $R$ .

You will be given  $Q$  queries. In each query, you are asked to find the size of the subtree of a given node  $X$ .

Input

The first line contains two integers  $N, R$   
 $(1 \leq N \leq 2 \times 10^5, 1 \leq R \leq N)$  — the number of nodes and the root of the tree.

The next  $N - 1$  lines each contain two integers  $u_i, v_i (1 \leq u_i, v_i \leq N)$  — representing an bidirectional edge between nodes  $u_i$  and  $v_i$ .

The next line contains an integer  $Q (1 \leq Q \leq 2 \times 10^5)$  — the number of queries.

The next  $Q$  lines each contain a single integer  $X (1 \leq X \leq N)$  — the node whose subtree size you need to compute.

Output

For each query, print a single integer — the size of the subtree of node  $X$ .

input
4 1 3 1 1 2 4 2 3 1 4 2
output
4 1 2

input
5 3 1 2 5 3 3 2 2 4 5 3 5 4 2 1
output
5 1 1 3 1

input
8 2 1 7 7 3 3 6 6 5 5 2 2 8 8 4 8 6 4 2 1 7 5 8 3
output
4 1 8 1 2 5 2 3

input
1 1 1 1
output
1

E. What's the Diameter?

1 second🕒, 1024 megabytes

You are given an **undirected connected** graph with  $N$  nodes and  $N - 1$  edges. Your task is to find two nodes such that the path between those two nodes is the longest possible in the graph.

Input

The first line contains one integer  $N$  ( $2 \leq N \leq 200000$ ) — the number of nodes.

The next  $N - 1$  lines will contain two integers  $u_i, v_i$  ( $1 \leq u_i, v_i \leq N$ ) — denoting there is a bidirectional road between  $u_i$  and  $v_i$ .

Output

On the first line, print a single integer — the length of the longest path.

On the second line, print two integers A and B — the nodes that form this longest path. If multiple pairs exist, you may print any one.

input
5 5 1 1 4 4 2 3 2
output
4 3 5

input
5 1 2 5 3 3 2 2 4
output
3 5 1

input
8 1 7 7 3 3 6 6 5 5 2 2 8 8 4
output
7 4 1

input
7 7 5 5 6 6 1 1 3 3 4 4 2
output
6 7 2

F. An Ancient Ordering

1 second🕒, 256 megabytes

You have found an old dictionary containing  $N$  words. The words are stored in an order that is different from the regular Latin lexicographic order.

Your task is to determine the order of the alphabet that satisfies the lexicographic order of this dictionary. If there are multiple valid orders, print the **lexicographically smallest** one. For example, the sequence  $S_1 = "d x i k"$  is lexicographically smaller than the sequence  $S_2 = "d x p a k"$ .

If no such valid sequence exists, print  $-1$ . A valid ordering is not possible if the characters create cyclic dependencies or if a longer word appears before a shorter word that is a prefix of it.

Input

The first line contains an integer  $N$  ( $1 \leq N \leq 1000$ ) — the number of words in the dictionary.

The next  $N$  line contains a string  $S$  ( $1 \leq |S| \leq 100$ ). Each word consists of only lowercase Latin letters  $a - z$ .

Output

Find out the order of the alphabets that satisfy the sorting order of the words in the given dictionary. If there are multiple valid orders, print the **lexicographically smallest** one. If no such valid sequence exists, print  $-1$ .

<b>input</b>
3 eat tea ate
<b>output</b>
eta

<b>input</b>
9 error tooth tot teeth their there thi tie hit
<b>output</b>
oethir

<b>input</b>
6 gef gie hf hd hc ha
<b>output</b>
efdcaghi

<b>input</b>
5 cmwage yent jtdgx wlp xufjpf
<b>output</b>
acdefglmnpqtuyjwx

<b>input</b>
6 abc ab p pq pqr pqrs
<b>output</b>
-1

<b>input</b>
2 pigeon pigeons
<b>output</b>
eginops

<b>input</b>
4 ab bc ca ac
<b>output</b>
-1