



Organizing Containers of Balls

by [ma5termind](#)

Problem

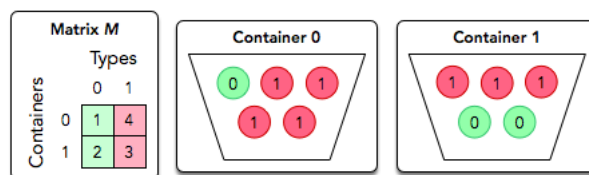
Submissions

Leaderboard

Discussions

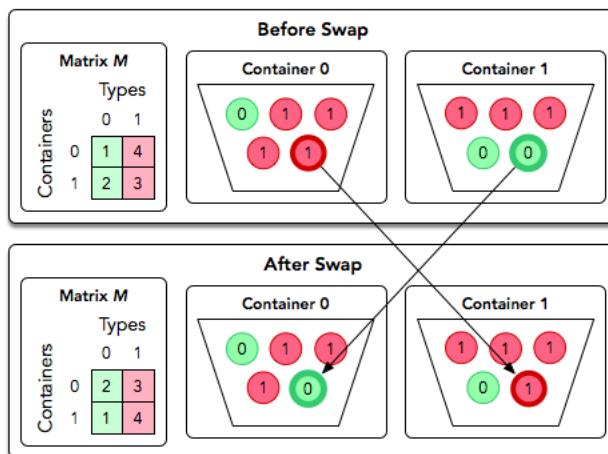
Editorial

David has n containers and n different types of balls, both of which are numbered from 0 to $n - 1$. The distribution of ball types per container are described by an $n \times n$ matrix of integers, M , where each $M_{c,t}$ is the number of balls of type t in container c . For example, consider the following diagram for $M = \begin{bmatrix} 1 & 4 \\ 2 & 3 \end{bmatrix}$:



In a single operation, David can *swap* two balls located in different containers (i.e., one ball is moved from container c_a to c_b and the other ball is moved from c_b to c_a).

For example, the diagram below depicts a single swap operation:



David wants to perform some number of swap operations such that both of the following conditions are satisfied:

- Each container contains only balls of the same type.
- No two balls of the same type are located in different containers.

You must perform q queries where each query is in the form of a matrix, M . For each query, print `Possible` on a new line if David can satisfy the conditions above for the given matrix; otherwise, print `Impossible` instead.

Input Format

The first line contains an integer denoting q (the number of queries). The subsequent lines describe each query in the following format:

- The first line contains an integer denoting n (the number of containers and ball types).
- Each line i of the n subsequent lines contains n space-separated integers describing row i in matrix M .

Constraints

- $1 \leq q \leq 10$
- $1 \leq n \leq 100$
- $0 \leq M_{c,t} \leq 10^9$

Scoring

- For 33% of score, $1 \leq n \leq 10$.
- For 100% of score, $1 \leq n \leq 100$.

Output Format

For each query, print `Possible` on a new line if David can satisfy the conditions above for the given matrix; otherwise, print `Impossible` instead.

Sample Input 0

```
2
2
1 1
1 1
2
0 2
1 1
```

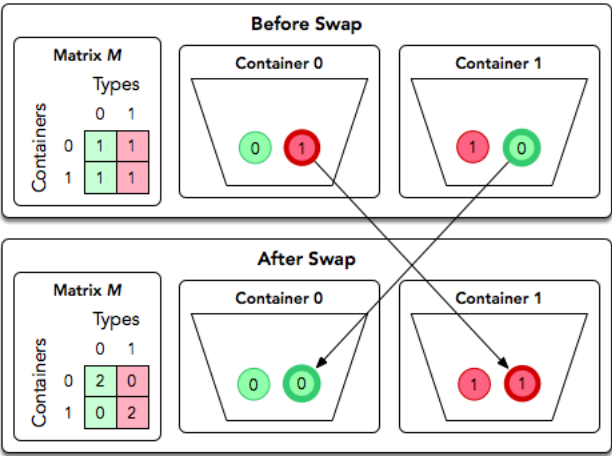
Sample Output 0

```
Possible
Impossible
```

Explanation 0

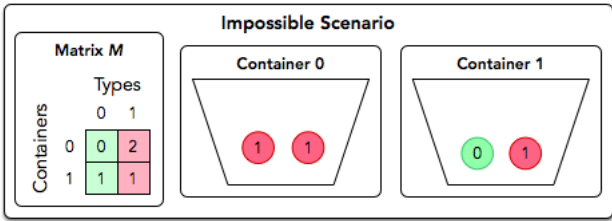
We perform the following $q = 2$ queries:

1. The diagram below depicts one possible way to satisfy David's requirements for the first query:



Thus, we print `Possible` on a new line.

2. The diagram below depicts the matrix for the second query:



No matter how many times we swap balls of type t_0 and t_1 between the two containers, we'll never end up with one container only containing type

t_0 and the other container only containing type t_1 . Thus, we print `Impossible` on a new line.



[f](#) [t](#) [in](#)Submissions: [5137](#)

Max Score: 30

Difficulty: Medium

Rate This Challenge:

☆☆☆☆☆

[More](#)Current Buffer (saved locally, editable)  

Java 7



```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8
9     public static void main(String[] args) {
10         Scanner in = new Scanner(System.in);
11         int q = in.nextInt();
12         for(int a0 = 0; a0 < q; a0++){
13             int n = in.nextInt();
14             int[][] M = new int[n][n];
15             for(int M_i=0; M_i < n; M_i++){
16                 for(int M_j=0; M_j < n; M_j++){
17                     M[M_i][M_j] = in.nextInt();
18                 }
19             }
20             // your code goes here
21         }
22     }
23 }
24
```

Line: 1 Col: 1

 [Upload Code as File](#)☐ Test against custom input[Run Code](#)[Submit Code](#)

Join us on IRC at [#hackerrank](#) on freenode for hugs or bugs.

[Contest Calendar](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) | [Request a Feature](#)