Problem F. Two Decks Of Cards

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Given two decks of cards with n and m cards respectively. The i-th card of the first deck has two sides showing the numbers $a_{i,1}$ and $a_{i,2}$; similarly, the i-th card of the second deck has two sides showing the numbers $b_{i,1}$ and $b_{i,2}$.

You need to choose a face-up side for each card such that:

- When considering each deck separately, the face-up sides contain distinct values.
- The number of distinct values that appear in at least one deck is *minimized*.

Find the number of distinct values that appear in at least one deck, or indicate if there is no way to choose the face-up sides for the cards such that the first condition is satisfied.

Input

Each test consists of multiple test cases. The first line contains the number of test cases t ($1 \le t \le 1000$). The description of each test case is as follows.

The first line contains two integers n, m ($1 \le n, m \le 10\,000$) — the number of cards in the first deck and the second deck.

The i-th line in the next n lines contains two integers $a_{i,1}$ and $a_{i,2}$ ($1 \le a_{i,1}, a_{i,2} \le 2(n+m)$) — the values on both sides of the i-th card in the first deck.

The i-th line in the next m lines contains two integers $b_{i,1}$ and $b_{i,2}$ ($1 \le b_{i,1}, b_{i,2} \le 2(n+m)$) — the values on both sides of the i-th card in the second deck.

The sum of n and the sum of m over all test cases is guaranteed not to exceed 10 000.

Output

For each test case, if it is not possible to find a valid way to choose the face-up sides, print -1. Otherwise,

- The first line should print an integer — the minimum number of distinct values appearing in at least one deck of cards.
- The second line should print n positive integers, where the i-th number is the face-up value of the i-th card from the first deck.
- The third line should print m positive integers, where the i-th number is the face-up value of the i-th card from the second deck.

If there are multiple ways to choose the face-up values for the cards, print any valid way.

Scoring

Subtask	Score	Constraints
1	750	$a_{i,u} \ne b_{j,v}$ for all $1 \le i \le n, 1 \le j \le m, 1 \le u, v \le 2$
2	750	$a_{i,u} \ne a_{j,v}$ for all $1 \le i < j \le n, 1 \le u, v \le 2$ and $b_{i,u} \ne b_{j,v}$ for all $1 \le i < j \le m, 1 \le u, v \le 2$
3	1500	No additional constraints
Total	3000	

Example

standard input	standard output
5 3 4 1 2 2 3 3 1 1 4 4 5 5 7 4 6 3 4 1 2 2 3 3 1 1 4 4 5 5 1 1 5 1 2 1 1 2 2 2 3 2 3 2 3 4 5 1 2 3 4 5 6 5 3 10 4 9 1 7 8 2 7 6 5 6 7 9 6 5 3	6 1 2 3 1 5 7 6 -1 3 1 2 3 3 2 4 2 4 6 5 10 9 8 7 5 7 9 5

Note

In the first test case, the cards of the first deck are $[(1, 2), (2, 3), (3, 1)]$, while the cards of the second deck are $[(1, 4), (4, 5), (5, 7), (4, 6)]$. One way to minimize the number of values that appear in at least one deck is $[1, 2, 3]$ and $[1, 5, 7, 6]$ with 6 distinct values, which is outputted as the sample output. Another way is $[2, 3, 1]$ and $[1, 5, 7, 6]$.

In the second test case, it can be proven that there is no way to choose face-up cards so that the second deck has distinct values.

The third test case satisfies the constraints of subtask 1, and the fourth test case satisfies the constraints of subtask 2.