Problem H. Classical Maximization Problem

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

You are given $2n$ distinct points on a plane. Point i has integer coordinates (x_i, y_i).

Points i and j are a *friendly pair* if either $x_i = x_j$ or $y_i = y_j$.

Form n pairs of points. Every point must belong to exactly one pair. The number of friendly pairs among your n pairs must be maximized.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 10^5$).

The i-th of the next $2n$ lines contains two integers x_i and y_i, denoting the coordinates of the i-th point ($-10^9 \le x_i, y_i \le 10^9$). All points are distinct.

It is guaranteed that the sum of n over all test cases does not exceed 10^5.

Output

For each test case, print a non-negative integer k, denoting the maximum possible number of friendly pairs.

In the i-th of the next n lines, print two integers a_i and b_i, denoting a pair formed by points a_i and b_i ($1 \le a_i, b_i \le 2n$; $a_i \ne b_i$).

Every integer from 1 to $2n$ must appear among a_i and b_i exactly once. The number of indices i such that points a_i and b_i are a friendly pair must be equal to k.

Example

standard input	standard output
3	2
2	2 4
0 0	3 1
0 1	2
1 0	4 3
1 1	2 1
2	0
0 0	1 2
0 1	3 4
0 2	
0 3	
2	
0 0	
1 1	
2 2	
3 3	