Problem G. Grand Test

Input file:	grand.in	Time limit:	3 seconds
Output file:	grand.out	Memory limit:	512 megabytes

Jeremy, Richard and James like to test cars. It is always hard for them to decide where they should do it. Usually car test looks like this. They choose a country and examine its cities and two-way roads that connect them. To perform a test, they need to choose two different cities S and F, such that there exist three routes between them. Moreover, each city except S and F should be visited by at most one route, and none of the roads may be used twice.

Then each of them takes a car in city S, drives along one of those routes and tries to get to city F faster than others.

You are given a description of multiple countries. For each country you should decide if it is possible to choose two cities and three routes between them in a way described above.

Input

The first line of the input contains a single integer T — number of countries ($1 \le T \le 100\,000$). It is followed by T country descriptions.

The first line of each country description contains two integers n and m — the number of its cities and roads ($1 \le n, m \le 100\,000$). The following m lines contain two integer numbers each: u_i and v_i — the cities at the ends of the road ($1 \le u_i < v_i \le n$). All roads are two-way. Each pair of cities is connected by at most one road.

Both the total number of cities and roads in all countries does not exceed 100 000.

Output

Output the answer for each country in the order they are given in the input.

If it is not possible to test cars in this country, the answer is -1. Otherwise the first line of the answer should contain two integers S and F — start and finish cities. The next three lines should contain three distinct routes. Each route is described by an integer k — the number of cities it visits, and k numbers $v_1, v_2, ..., v_k$ — the cities, where $v_1 = S$, $v_k = F$, and there is a road between cities v_i and v_{i+1} for all $1 \le i \le k - 1$.

Example

grand.in	grand.out
2 6 6 3 6 3 4 1 4 1 2 1 3 2 3 3 1 1 2	1 3 3 1 2 3 2 1 3 3 1 4 3 -1