

### E. One-Way Reform

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

There are  $n$  cities and  $m$  two-way roads in Berland, each road connects two cities. It is known that there is no more than one road connecting each pair of cities, and there is no road which connects the city with itself. It is possible that there is no way to get from one city to some other city using only these roads.

The road minister decided to make a reform in Berland and to orient all roads in the country, i.e. to make each road one-way. The minister wants to **maximize** the number of cities, for which the number of roads that begins in the city **equals** to the number of roads that ends in it.

#### Input

The first line contains a positive integer  $t$  ( $1 \leq t \leq 200$ ) — the number of testsets in the input.

Each of the testsets is given in the following way. The first line contains two integers  $n$  and  $m$  ( $1 \leq n \leq 200, 0 \leq m \leq n \cdot (n - 1) / 2$ ) — the number of cities and the number of roads in Berland.

The next  $m$  lines contain the description of roads in Berland. Each line contains two integers  $u$  and  $v$  ( $1 \leq u, v \leq n$ ) — the cities the corresponding road connects. It's guaranteed that there are no self-loops and multiple roads. It is possible that there is no way along roads between a pair of cities.

It is guaranteed that the total number of cities in all testset of input data doesn't exceed 200.

Pay attention that for **hacks**, you can only use tests consisting of **one testset**, so  $t$  should be equal to one.

#### Output

For each testset print the maximum number of such cities that the number of roads that begins in the city, is equal to the number of roads that ends in it.

In the next  $m$  lines print oriented roads. First print the number of the city where the road begins and then the number of the city where the road ends. If there are several answers, print any of them. It is allowed to print roads in each test in arbitrary order. Each road should be printed exactly once.

#### Example

input	Copy
<pre>2 5 5 2 1 4 5 2 3 1 3 3 5 7 2 3 7 4 2</pre>	
output	Copy
<pre>3 1 3 3 5 5 4 3 2 2 1 3 2 4 3 7</pre>	

#### Codeforces Round #375 (Div. 2)

Finished

#### → Practice?

Want to solve the contest problems after the official contest ends? Just register for practice and you will be able to submit solutions.

Register for practice

#### → Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

#### → Problem tags

constructive algorithmsdfs and similar

flowsgraphsgreedy\*2300

No tag edit access

#### → Contest materials

- Announcement #1 (en) ×
- Announcement #2 (ru) ×
- Tutorial #1 (en) ×
- Tutorial #2 (ru) ×

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov  
The only programming contests Web 2.0 platform  
Server time: Jul/15/2019 14:27:09<sup>UTC+8</sup> (e3).  
Desktop version, switch to [mobile version](#).  
[Privacy Policy](#).

Supported by



ITMO UNIVERSITY