

Free Tour

In order to celebrate the 2nd anniversary of Travel Agent SPOJ (Safe – Professional – hospitable – Joyful), the management intend to hold free tours around cities for clients to make them more satisfied with SPOJ.

A tour is a simple cycle, starting at any city (called a source-city) visits some other cities (each city must be visited at most once) and then returns to the source-city. The number of roads in the tour should be an even number because we are celebrating a 2nd anniversary, and 2 is even! Since many tours in different areas of the country are planned, the cost of organising them could turn out quite high. Hence, the management of SPOJ hope to find at least one 'reasonable' tour, which should have as small a number of roads as possible.

You're given maps of the areas where SPOJ wants to hold free tours. For each map, help them figure out a reasonable tour.

Input

The first line of input contains an integer **t**, the number of maps ($t \leq 5$). **t** maps follow.

For each map:

- In the first line there are 2 integers **N** – number of cities in that area, **M** – number of roads ($1 \leq N \leq 8000, 0 \leq M \leq 10000$)
- In the next **M** lines, the *i*-th line describes the *i*-th road: a line with two integers **a b** denotes a bidirectional road between city **a** and city **b**

There is one blank line between successive tests.

Output

For each map, if there is no tour satisfying the conditions, write "-1" (without quotes). Otherwise, write one integer representing the number of roads in a reasonable tour, and in the next line show out the tour with form "source-city a b c ... source-city", that means the tour is source-city → city a → city b → ... → source-city. If there are many tours satisfy in each map, any of them will be accepted.

Example

Input:

2

3 3

1 2

2 3

3 1

4 4

1 2

2 3

3 4

4 1

Output:

-1

4

1 2 3 4 1