Problem A. Soccer Match

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

As a big sports fan, you, the primary leader of the Pigeon Kingdom, are organizing a soccer match! A total of N players signed up for the match, and you plan to divide them into three groups: Red team, Blue team, and spectators. The number of players in the Red team and the Blue team **can** be different.

There are M pairs of friends among the N participants, where $M \geq 2KN$ for some given constant $K \geq 1$. The friendship is mutual, which means that if a is a friend of b, then b is a friend of a, and vice versa. To make the match more exciting, you want to make sure that each player in the Red team has at least $K + 1$ friends in the Blue team, and each player in the Blue team has at least $K + 1$ friends in the Red team. Can you find an arrangement satisfying such constraints?

Input

The first line contains one integer T ($1 \leq T \leq 50\,000$), denoting the number of test cases. For each test case:

The first line contains three integers, N, M, and K ($1 \leq N, M, K \leq 50\,000$ and $M \geq 2KN$), denoting the number of players, the number of pairs of friends, and the given constant, respectively.

Then M lines follow, each containing two integers u and v ($1 \leq u < v \leq N$), denoting that u and v are friends.

It is guaranteed that, in each test case, each pair of (u, v) appears at most once, and the sum of M over all test cases does not exceed $50\,000$.

Output

For each test case, output two lines:

The first line begins with one integer R, denoting the number of players in the Red team. Then R space-separated integers follow, each denoting the index of a player in the Red team.

The second line follows the same format. It begins with an integer B, denoting the number of players in the Blue team. Then B space-separated integers follow, each denoting the index of a player in the Blue team.

If there are multiple solutions, you can output any one of them. It can be shown that, under such constraints, a solution always exists.

Example

standard input	*standard output*
2 5 10 1 1 2 1 3 1 4 1 5 2 3 2 4 2 5 3 4 3 5 4 5 10 20 1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 1 10 1 4 4 7 7 10 3 10 3 6 6 9 2 9 2 5 5 8 1 8	3 2 3 4 2 1 5 3 2 8 10 2 1 9