The Greatest Graph Ever

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Ambar Pal just managed to hack into Facebook's Social Graph. There exists an edge between two people on the graph if they are in each other's friend list. Since friendship in Facebook is bidirectional, the graph is essentially undirected.

He thinks it is the biggest graph ever! However, instead of stalking people, he and Rounaq switch to another problem. They want to divide the people on Facebook amongst themselves, each having a few users in their set. However, they are not fine with any ordinary division.

They have recently learnt about vertex covers. Suppose the graph G is given. Subset A of its vertices is called a vertex cover of this graph if for each edge (u, v) there is at least one endpoint of it in this set, i.e. $u \in A$ or $v \in A$ (or both).

Both of them wish that their part is a vertex cover. However, they suddenly feel tired and decide to give over the job to you. Given the graph, you have to give them two disjoint sets of vertices A and B, such that both A and B are vertex covers, or claim that the task is impossible?

Input

The first line consists of 2 numbers, $N(2 \leq N \leq 10^5)$ and $M(1 \leq M \leq \frac{N(N-1)}{2})$, denoting the number of vertices and Edges. The next M lines, consist edges of the Graph.

Output

If it's impossible, print `-1` .

If there are two disjoint sets of vertices, such that both sets are vertex cover, print their descriptions. Each description must contain two lines. The first line contains a single integer k denoting the number of vertices in that vertex cover, and the second line contains k integers — the indices of vertices. Note that because of $M \geq 1$, vertex cover cannot be empty.

Examples

standard input	standard output

5 7
3 2
5 4
3 4
1 3
1 5
1 4
2 5
	-1

10 16
6 10
5 2
6 4
6 8
5 3
5 4
6 2
5 9
5 7
5 1
6 9
5 8
5 10
6 1
6 7
6 3
	8
	1 2 3 4 9 7 8 10
	2
	5 6