Problem E. United States of Byteland

Input file:	`scc.in`
Output file:	`scc.out`
Time limit:	2 seconds
Memory limit:	256 mebibytes

In Byteland, there are N cities connected by two-way roads. King of Byteland decided to change the political system to federal constitutional republic with exactly K states (each state consists of several cities).

King doesn't want a revolution, so he asked his political scientists for help. They reported that the most simple way to reform the political system is to make each road a one-way road and divide cities into states according to the following simple rule: for each pair of cities A and B, they belong to the same state if and only if there are paths from A to B and from B to A. You are the best of the best programmers of the kingdom, so you should investigate if is it possible to reform without pain and horror.

Input

The first line of input contains three integers: N, the number of cities, M, the number of roads, and K, the required number of states. ($1 \leqslant N \leqslant 16$, $0 \leqslant M \leqslant 10^5$, $1 \leqslant K \leqslant N$). The next M lines contain two integers u_i and v_i — the numbers of cities connected by i-th road ($1 \leqslant u_i, v_i \leqslant N$).

Output

If the reform is impossible, print "`NO`". Other, on the first line, print "`YES`", and on the following M lines output the plan of the reform. Each of these M lines corresponds to one edge in the input. Edges should be output in the same order as in the input. The i-th of these lines should contain integers u_i and v_i in some order separated by a space. The order is important: if you output u_i v_i, i-th road is directed from u_i-th city to v_i-th city; if you output v_i u_i, it is directed from v_i-th city to u_i-th city instead.

Examples

scc.in	scc.out

5 6 3
1 2
2 3
3 1
1 4
2 5
4 5
	YES
	1 2
	2 3
	3 1
	4 1
	5 2
	5 4

5 6 4
1 2
2 3
3 1
1 4
2 5
4 5
	NO

16 0 16	YES