

## Problem B. The Witcher

Input file: *standard input*  
Output file: *standard output*  
Time limit: 6 seconds  
Memory limit: 512 mebibytes

The Witcher is now in some serious trouble! He needs to prepare a pocket map for his upcoming journey. He is now in a tavern and here there is a map of the whole Continent. For simplicity, we assume that the map is just an undirected graph without loops, but it may contain multiple edges between the same pair of nodes. There is an unwritten rule in the Witcher's world, that each graph placed in a pocket map should satisfy the following condition: the degree of each vertex should be even. The Witcher doesn't want to break the rule and also the graph in the pocket map should have the same set of nodes as the graph from the map in the tavern and the edges of this graph should form a subset of the edges from the graph in the tavern. Of course, there are some edges, which are necessary for the Witcher to complete his upcoming adventure, so he would also like to place them in his pocket map. The Witcher would like to know, if he can construct a valid graph, fulfilling the given conditions and if he can, then you should tell him, which edges he should take into his pocket map.

### Input

In the first line one integer  $Z \leq 100$  is given, denoting number of testcases described in following lines.

The first line of the standard input contains two integers  $n, m$  meaning the number of nodes and the number of edges in the graph from the map in the tavern. Each of the next  $m$  lines contains the description of one edge in the graph.  $i$ -th of them contains three integers  $a_i, b_i, x_i$  meaning that the  $i$ -th edge is connecting the nodes with numbers  $a_i$  and  $b_i$ , and if  $x_i = 1$ , then the Witcher needs this edge in his pocket map, otherwise you can choose it, but it is not necessary.

$(n \in [1, 5 \cdot 10^5], m \in [0, 5 \cdot 10^5]), (a_i, b_i \in [1, n], a_i \neq b_i, x_i \in \{0, 1\})$ .

### Output

For each test case, the first line of the standard output should contain one word "TAK", if the Witcher can prepare the pocket map fulfilling all the conditions, or "NIE" otherwise. If the first line contains the word "TAK", then in the next  $m$  lines you should print the numbers  $y_i \in \{0, 1\}$ , such that  $y_i \geq x_i$  and the graph formed from the edges with  $y_i = 1$  should satisfy all the Witcher's conditions.

### Example

standard input	standard output
2	TAK
4 5	1
1 2 0	0
2 3 0	0
3 1 0	1
1 4 1	1
2 4 1	NIE
5 4	
1 2 0	
2 3 0	
3 4 1	
4 5 0	