

Infinite Market

Input file: **standard input**
Output file: **standard output**
Time limit: 4 seconds
Memory limit: 256 megabytes

Busy Beaver is at the farmers' market! There are N stalls, numbered from 1 to N . There are also M directed paths between stalls. The i -th path goes from stall a_i to stall b_i , where $a_i \neq b_i$. It is guaranteed that no paths leave stall N , at least one path leaves every stall other than stall N , no two paths have the same starting and ending stalls, and for every path going from stall $r_1 \neq N$ to $r_2 \neq N$, there is another path going from r_2 to r_1 . Each path i that does not end at stall N has a unique successor path s_i . It is guaranteed that path s_i can be used after path i . In other words, $a_{s_i} = b_i$. Each stall also has an integer counter x_i .

Busy Beaver chooses a stall to start his shopping. First, Busy Beaver travels along any path leaving his starting stall. Then, every minute, supposing Busy Beaver has just traveled along path p from a_p to b_p , he performs the following two actions:

1. He reaches b_p and increments x_{b_p} by 1.
2. If x_{b_p} is a multiple of some positive integer K , Busy Beaver will go along path s_p next. Otherwise, Busy Beaver travels along any path leaving b_p .

If Busy Beaver ever reaches stall N , he will leave the farmers' market. Given the map of the farmers' market, does there exist a positive integer K , initial integer values for all x_i , and a starting stall for Busy Beaver, so that he can stay in the market forever?

Input

The first line contains a single integer T ($1 \leq T \leq 10^4$) — the number of test cases.

The first line of each test case contains two space-separated integers N and M ($2 \leq N \leq 2 \cdot 10^5$, $1 \leq M \leq 4 \cdot 10^5$) — the total number of stalls and the number of directed paths in the farmers' market.

The i -th of the next M lines of each test case contains three space-separated integers a_i , b_i , and s_i ($1 \leq a_i, b_i \leq N$, $a_i \neq b_i$, $1 \leq s_i \leq M$) — indicating that the i -th path goes from stall a_i to stall b_i , and its unique successor path is s_i . If $b_i = N$, then $s_i = -1$, indicating the path has no successor.

It is guaranteed that the sum of N over all test cases does not exceed $2 \cdot 10^5$ and the sum of M over all test cases does not exceed $4 \cdot 10^5$.

Output

For each test case, if there exists a value for K , initial values for all x_i , and a starting stall such that Busy Beaver can stay in the market forever without ever reaching stall N , print "YES". Otherwise, print "NO".

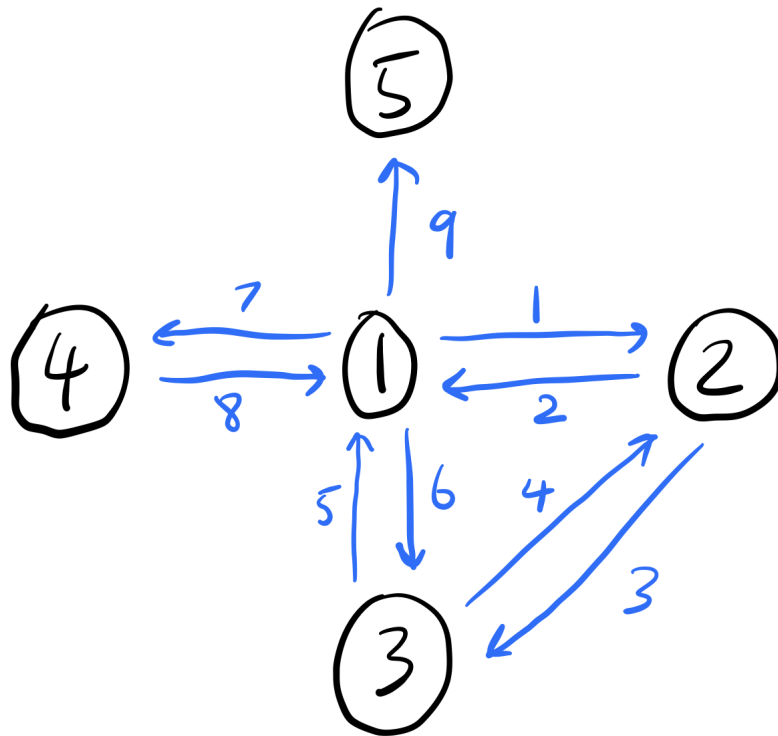
You can output the answer in any case (upper or lower). For example, the strings "yEs", "yes", "Yes", and "YES" will be recognized as positive responses.

Example

standard input	standard output
2	YES
5 9	NO
1 2 3	
2 1 7	
2 3 5	
3 2 2	
3 1 9	
1 3 4	
1 4 8	
4 1 1	
1 5 -1	
4 9	
1 2 8	
2 1 7	
2 3 9	
3 2 8	
3 1 7	
1 3 9	
1 4 -1	
2 4 -1	
3 4 -1	

Note

The market for test case 1 is shown below. The stalls are circled and the paths are in blue. It is possible for Busy Beaver to stay in the market forever. A solution is to set $K = 2$, $x = [0, 0, 0, 0, 0]$, and initially put Busy Beaver at stall 1. Busy Beaver will then travel along the closed path through the stalls $1 \rightarrow 2 \rightarrow 3 \rightarrow 1 \rightarrow 4 \rightarrow 1$, repeating forever. When Busy Beaver reaches stall 1 with path 5, x_1 becomes odd, and when Busy Beaver reaches stall 1 with path 8, x_1 becomes even. This ensures that Busy Beaver is never forced to take path 9 (which leads to stall N).



The market for test case 2 is shown below. It can be shown that it is impossible for Busy Beaver to stay in the market forever.

