

Problem B. Bipartite graph coloring

Input file: `stdin`
Output file: `stdout`
Time limit: 5 seconds
Memory limit: 512 megabytes

bobo gets a bipartite graph with n vertices (that is, a graph without odd cycles).

He colors each vertex into black or white, and then calculates the product of each edge's value. The value of an edge is determined by the colors of its two end points. Thus, there can be $2 \times 2 = 4$ different values associated to a given edge.

Now bobo would like to know the sum of products of all 2^n possible coloring, modulo $(10^9 + 7)$.

Input

The first line contains 2 integers n, m which denotes the number of vertices and edges ($2 \leq n \leq 40$, $1 \leq m \leq 100$).

Vertices are numbered by $1, 2, \dots, n$ for convenience.

Each of the following m lines contains 6 integers $a_i, b_i, v_{i,00}, v_{i,01}, v_{i,10}, v_{i,11}$, which denotes an edge between vertices a_i and b_i ($1 \leq a_i, b_i \leq n, 0 \leq v_{i,00}, v_{i,01}, v_{i,10}, v_{i,11} \leq 10^9$).

- If vertices a_i and b_i are both white, the i -th edge's value is $v_{i,00}$.
- If vertex a_i is white and b_i is black, the value is $v_{i,01}$.
- If vertex a_i is black and b_i is white, the value is $v_{i,10}$.
- If vertices a_i and b_i are both black, the value is $v_{i,11}$.

Output

A single integer denotes the sum.

Sample input and output

stdin	stdout
2 1 1 2 1 2 3 4	10
3 2 1 2 1 0 0 1 2 3 1 0 0 1	2