Sashka loves to regularly visit your class in informatics on Thursdays, where you always cover interesting topics. The last class was with a deputy teacher, who taught the whole class about bitwise operations. They emphasized on the *[excluding](https://en.wikipedia.org/wiki/XOR_gate" \l ":~:text=XOR%20gate%20(sometimes%20EOR%2C%20or,to%20the%20gate%20is%20true.) or* (XOR) operation. Sashka, impressed by the capabilites of the operation, decided to open her older brother’s notebook, full of interesting tasks, searching for XOR ones. Initially, she solved the easier problems, but now she stumbled upon the following exotic task:

You are gived a tree of vertices, with weighted edges. You are allowed to make changes of the values of the edges in order to minimize the count of *unpleasant* simple paths in the tree. A simple path is a path, which doesn’t contain a vertex more than once. A path is *unpleasant*, when the *excluding or* of the values of the edges it contains, has an odd parity of set bits in its bitwise representation. Find the minimal possible count of *unpleasant* paths, after optimal use of operations.

Sashka used all of her knowedge in informatics, and was able to solve the task! At least that’s what she thought, until next Thursday, when you said that she always uses operations. Now she is very unhappy, which gives you the opportunity to cheer her up, by writing a program xorfun.cpp, which solves the mentioned task.

**Input**

The first line of xorfun.in constains two natural numbers and *.* The next lines contain natural numbers – the two endpoints of an edge и , and its weight – .

**Output**

On one line in xorfun.out you should print one number – the minimal count of *unpleasant* paths.

**Constaints**

**Time Limit: 3 sec.**

**Memory Limit: 256 MB.**

**Sample testcases**

|  |  |
| --- | --- |
| **Input (xorfun.in)** | **Output (xorfun.out)** |
| 9 3  1 2 6  1 7 4  2 3 7  3 4 12  3 5 14  3 6 13  5 8 2  5 9 11 | 8 |
| 10 3  5 2 938707311  6 5 312182765  7 5 952433887  1 5 1630822531  3 5 1850473008  9 1 510193547  8 6 1465047925  10 8 1237069467  4 9 1320045850 | 9 |
| 15 4  1 3 1936267205  3 10 1050608599  15 1 1084713226  15 6 327822164  10 8 1085192654  1 9 479650095  12 9 16193632  12 11 1114880438  8 5 1011320449  8 13 2012210084  14 3 47220444  7 15 1406284003  4 1 16038788  13 2 733939625 | 36 |