

## D. Xor Spanning Tree

time limit per test: 2 seconds  
 memory limit per test: 128 megabytes  
 input: standard input  
 output: standard output

In the galaxy far far away is the ancient interplanetary republic of Bubbleland, consisting of  $N$  planets. Between them, there are  $M$  bidirectional wormholes, each connecting a pair of planets. Bubbleland is a very centralized republic, having a capital planet Whiteplanet, from which any another planet can be reached using these wormholes. It is also guaranteed that no wormhole connects planet to itself and that no two different wormholes connect same pair of planets.

We call a path that begins at one planet, visits other planets and each of them at most once and returns to starting point a *tour*. Interplanetary Safety Regulations guarantee that each planet belongs to at most one *tour* and that there are at most 42 *tour*s.

After many eons of usage, wormholes need to be repaired and each wormhole has the cost  $W_i$  which needs to be paid for reparation. Unfortunately, the Senate of Bubbleland is short on budget. Therefore, they have decided only to fix as many wormholes as they need in order to have all planets reachable from capital and to pay as little money as they have to for this repair. However the way in which the Senate calculates the cost is different. Cost of the set of reparations is binary xor of costs of each individual reparation, that is if reparations to be made have costs  $A_1, A_2, \dots, A_k$ , the cost of entire set is  $A_1 \oplus A_2 \oplus \dots \oplus A_k$ .

Now the Senate would like to know how much money do they have to pay and also the number of different ways to achieve that cost **modulo** 1000000007.

### Input

First line of input contains two numbers  $N(1 \leq N \leq 100.000)$ , the number of planets and  $M(1 \leq M \leq 100.041)$ , the number of wormholes. Following  $M$  lines contain three numbers  $U, V(1 \leq U \neq V \leq N)$  and  $W(1 \leq W \leq 100.000)$ , meaning that there exists a wormhole connecting planets  $U$  and  $V$ , with repair cost of  $W$ .

### Output

Output two numbers, the smallest possible cost of entire reparation and the number of different valid reparations with that cost **modulo** 1000000007.

### Example

input	Copy
6 6 4 1 5 5 2 1 6 3 2 1 2 6 1 3 3 2 3 4	
output	Copy
1 1	

### Note

We can repair wormholes 1,2,3,5 and 6, paying  $5 \oplus 1 \oplus 2 \oplus 3 \oplus 4 = 1$ , one can check that this is the cheapest repair in which all of the planets are connected and the only valid repair with that cost.

### Bubble Cup 12 - Finals [Online Mirror, unrated, Div. 1]

Finished

Practice



### → Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

### → Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

### → Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

### → Submit?

Language: GNU G++11 5.1.0

Choose file:  未选择任何文件

### → Problem tags

divide and conquer fft graphs \*2600  
 No tag edit access

### → Contest materials

- Announcement (en) ✕
- Statements (en) ✕
- Tutorial (en) ✕

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov  
The only programming contests Web 2.0 platform  
Server time: Sep/18/2019 20:14:57<sup>UTC+8</sup> (e2).  
Desktop version, switch to [mobile version](#).  
[Privacy Policy](#)

Supported by



ITMO UNIVERSITY