Specializing Villages

Problem

The countryside of Kickstartia consists of **V** villages, connected by **E** bidirectional roads. Because the citizens appreciate diversity in their road construction, no two roads have the same length. Each road connects exactly two villages, and no two roads connect the same two villages.

The new king, eager to show off his progressiveness, would like to create a plan in which each village will specialize in producing exactly one food: either fruit or vegetables. If a village produces fruit, then they will find a shortest path (perhaps using multiple roads) to some village that produces vegetables. Similarly, if a village produces vegetables, then they will find a shortest path to some village that produces fruit.

To keep things running smoothly, the king would like to minimize the average of the distances that each village needs to travel to get the food that it does not produce.

There could be many plans that minimize this average distance, so the king would like you to tell him how many there are. Two plans are different if there is a village that produces fruit in one plan, but vegetables in the other. The king guarantees that it is possible to find a plan which allows each village to get both fruit and vegetables.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case starts with one line, containing two integers **V** and **E**: the number of villages, and the number of roads, respectively. The villages are labeled from 1 to **V**. **E** lines follow; the i-th of these lines contains three integers A_i, B_i and L_i, indicating that the i-th road connects village A_i to village B_i and has length L_i.

Output

For each test case, output one line containing `Case #x: y`, where `x` is the test case number (starting from 1) and `y` is the answer the king wants, as described above.

Limits

$1 \le$ **T** ≤ 100.
Time limit: 20 seconds per test set.
Memory limit: 1 GB.
$1 \le$ **E** $\le \min(1000,$ **V** $*$ (**V** $- 1) / 2)$.
$0 \le L_i \le 10^5$, for all i.
$L_i \ne L_j$ for all $i \ne j$.
$1 \le A_i < B_i \le$ **V**, for all i.
$(A_i, B_i) \ne (A_j, B_j)$ for all $i \ne j$.
There is at least one plan that allows every village to get both types of food.

Small dataset (Test set 1 - Visible)

$2 \le$ **V** ≤ 10.

Large dataset (Test set 2 - Hidden)

$2 \le$ **V** ≤ 50.

Sample

Sample Input	Sample Output
2	Case #1: 2
3 3	Case #2: 16
1 2 1	
1 3 6	
2 3 4	
6 5	
1 2 6	
3 4 0	
5 6 7	
3 5 1	
4 6 2	

In Sample Case #1, one possible plan is to make villages 1 and 3 produce fruit and make village 2 produce vegetables. Village 1 and 2 can travel to each other to get to the food they lack, so they both have to travel distance of 1. Village 3 needs to travel to village 2 to get vegetables, traveling a distance of 4. In total, the average distance is (1 + 1 + 4)/3 = 2, which is the minimum possible. There is one other optimal plan (in which villages 1 and 3 produce vegetables and village 2 produces fruit), so the final answer is 2.

In Sample Case #2, there are 16 possible plans. One way is to have villages 1, 3 and 5 produce fruit while villages 2, 4 and 6 produce vegetables. Village 1 and 2 must travel to each other to get to the food they lack. Villages 3 and 5 can travel to village 4 to get vegetables, while villages 4 and 6 can travel to village 3 to get fruit. The average distance is (6 + 6 + 0 + 1 + 0 + 2)/6 = 2.5, which turns out to be the minimum possible. Even if two villages are connected by a road of length 0, they are considered to be distinct villages.