

G	Spying				
	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; padding: 2px 5px;">Time Limit</td><td style="padding: 2px 5px;">1 second</td></tr> <tr> <td style="padding: 2px 5px;">Memory Limit</td><td style="padding: 2px 5px;">32 MB</td></tr> </table>	Time Limit	1 second	Memory Limit	32 MB
Time Limit	1 second				
Memory Limit	32 MB				

There's a war going on in country called Puripuri between the Kingdom and the Commonwealth, the two factions trying to monopolize a mysterious ore mine which started a war and the country was split in half. You're a spy assigned by the Commonwealth and your mission is to infiltrate a school in the Kingdom to observe a teacher name Mr. X, who is thought to be an outstanding weapon researcher in the Kingdom, over the course of **W** weeks.

Mr. X has **N** apartments that labeled from **1** to **N** where he usually meets with Kingdom's generals in secret. Each apartment is connected via underground passage (there might be multiple different routes connecting the same pair of apartments), but routes in the underground passage are full of traps against spies like yourself. So, you have to follow Mr. X trail to avoid traps and setup hidden cameras along the path so that you will know when they set new traps to avoid activating them. Mr. X has a set of routes he can use to travel between apartments and every Saturday of each week he will come up with a new route which you have not known before (this new route might connect two of the apartments which already have routes connecting them) and add it to his set of routes.

You want to be able to move to every apartment via the underground passage and want to minimize the number of hidden cameras inside to avoid the risk of Mr. X finding out (no hidden cameras are set in any unused routes). So, you want to maintain a set of safe routes by setting up hidden cameras which you can use to travel between apartments. Each of the routes need different number of hidden cameras and you only know how many to set once you follow Mr. X. Since each week new traps are set. Meaning that to avoid traps you must travel only via the **safe routes** you already set hidden cameras, or the seven routes Mr. X uses during the week. You have a choice of maintaining the same set of safe routes or remove some routes and adding the routes Mr. X uses through the week. The first week of your mission is special since you took over the last spy assigned to this mission and she already set up **M** hidden cameras over all the routes Mr. X



have taken but you don't really need all of them, do you? It is guaranteed that the M safe routes provided by the last spy allow you to travel to every apartment.

The only difficult thing is that after setting hidden cameras at the end of each week on Saturday (since you are a student), you have to change the hidden cameras' battery. So, you would like to send message to Commonwealth HQ to request for batteries equals to the number of hidden cameras you have set up. Your task is to output how many batteries to request at the end of each week.

INPUT

The first line of the input contains one integer T ($1 \leq T \leq 5$) which denotes the number of tests in the input file. Then T tests follow on separate lines, where the format of each test is as follows:

Each test case starts with a line of two integers: N and M . ($2 \leq N \leq 4,000$; $N \leq M \leq 2N$)

Each of the next M lines contains three integers: A , B and C where A and B denote the apartments label and C denotes the number of hidden cameras already setup in the safe routes between them.

Next line containing one integer: W ($1 \leq W \leq 8,000$)

Each of the next W group of lines each contains seven lines and each line in the group contains three integers: X , Y , and Z where X and Y denote the apartments label and Z denotes the number of hidden cameras needed to setup between X and Y to make this route a safe route. The i^{th} group of line represents the i^{th} week you are assign to the mission. And the $7 * (i - 1) + j^{\text{th}}$ line represents the j^{th} day of the i^{th} starting from Sunday to Saturday.

OUTPUT

For each test case, output one line containing "Case #x:" where x is the test case number starting from 1 and W following lines containing one integer b where b is the number of batteries needed each week.

EXAMPLE

Sample Input	Sample Output
<pre> 2 3 4 1 2 1 1 3 4 1 2 3 2 3 5 2 1 2 1 1 3 4 1 2 1 2 3 5 2 3 5 1 2 1 1 2 1 1 2 2 2 1 2 1 2 1 2 3 5 1 2 1 1 3 4 1 3 4 3 2 5 2 3 2 3 2 1 2 8 1 3 9 3 1 2 8 1 3 9 2 1 8 1 2 8 3 1 9 1 3 9 1 2 7 1 2 8 2 1 8 1 2 7 2 1 7 3 1 9 2 1 8 2 1 8 1 2 6 1 2 8 1 2 8 2 1 8 2 1 6 1 3 9 2 1 6 3 2 1 </pre>	<pre> Case #1: 5 3 Case #2: 16 15 7 </pre>