

## Problem F. Bike Sharing

Program:            **bike.(cpp|java)**  
Input:              **bike.in**  
Balloon Color:     **Purple**

The city of Sharm El Sheikh is planning to open a bike sharing service in the city. They will build  $N$  bike stations at strategic points in the city. Each station will have a capacity  $C$ , meaning that at most  $C$  commuters may take a bike from a station in the morning, and at most  $C$  commuters may return a bike to a station in the evening. These numbers are independent (i.e. a station can still accept  $C$  bikes in the evening even if no bikes were taken from it in the morning).

You are in the planning department of this service, and you decided to do some market research to make sure the service is profitable. Here is what you found out: There are exactly  $M$  different groups of possible commuters that would like to use your service. Each group consists of  $P$  individuals, each individual in the same group wants to use the service in the same way: they all want to grab a bike from the a certain start station  $St$  in the morning, and return it to a certain (possibly the same) end station  $En$  in the evening, and each individual is willing to pay exactly  $X$  in return for this service. Since commuters need their bike for the whole day, no bike can be used by multiple commuters.

The number of bikes in a station may not satisfy all commuters. In this case you are free to pick the subset of commuters that you can satisfy that will pay the biggest amount in total (you don't need to pick the whole group, it's okay to pick a subset of the same group).

Now armed with the above information, you have to choose the best possible value for  $C$  (possibly zero). Of course having a larger capacity comes with a cost. The cost of having a capacity  $C$  is  $D \times C$ . Can you figure out the maximum profit (total revenue – total cost) this service can generate?

### Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer  $T$  ( $1 \leq T \leq 50$ ) representing the number of test cases. Followed by  $T$  test cases. Each test case starts with a line containing 3 integers separated by a single space  $N$ ,  $M$  and  $D$  ( $1 \leq N \leq 50$ ) ( $1 \leq M \leq 250$ ) ( $1 \leq D \leq 100,000$ ) representing the number of stations, the number of groups of possible commuters, and the cost per extra bike in all stations, respectively. Followed by  $M$  lines, each line will contain 4 integers  $P$ ,  $St$ ,  $En$  and  $X$  ( $1 \leq P \leq 100,000$ ) ( $1 \leq St, En \leq N$ ) ( $1 \leq X \leq 100,000$ ) representing the number of individuals in the group, the start and end of their commute, and the amount each individual is willing to pay, respectively.

### Output

For each test case print a single line containing "Case n:" (without quotes) where n is the test case number (starting from 1) followed by a space then the maximum profit.

### Examples

<b>bike.in</b>	<b>stdout</b>
2	Case 1: 10
2 3 3	Case 2: 50
10 1 2 2	
10 1 1 2	
10 2 2 2	
2 3 5	
10 1 2 10	
10 1 1 2	
10 2 2 2	