Sending Money

Lea and her friends often go out together and usually one person pays the bill for the rest. Since it is extremely tedious to balance the resulting debts instantly, Lea quickly wrote a program she called DebtGraph to record all payments. Now, the end of the month is approaching and it is time to settle all debts. Of course, it is of fundamental importance that as little money as necessary is sent around. Unfortunately, Lea is busy with competing in the "Fundamentally Awesome Universe"-contest and thus can't compute the exact transactions necessary to settle all debts this way. Can you help her find an optimal set of transactions settling all debts fairly, i.e. after these transactions everybody has exactly the amount of money which he would have if the debts were settled instantly?

Hint: You do not need to minimize the number of transactions, only the total sum of money sent with these transactions.

Input

The first line of the input contains an integer t. t test cases follow.

Each test case starts with two integers n and m, the number of firends and recorded debts between them, respectively. m lines follow, describing the debts. The i-th line contains three integers x_i, y_i, and c_i, denoting that a debt of x_i owing c_i money to y_i is recorded.

Output

For each test case, output a line containing "Case #i: a" where i is its number, starting at 1, and a is the minimal amount of money that needs to be sent in order to settle all debts. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n, m \leq 10^5$
- $1 \leq a \leq 10^8$

Sample Input 1

```
1
3 2
1 2 5
2 3 5
```

Sample Output 1

```
Case #1: 5
1 3 5
```

Sample Input 2

```
1
3 3
1 2 5
2 3 4
3 1 3
```

Sample Output 2

```
Case #1: 2
1 2 1
1 3 1
```