Zombie Apocalypse

Do not watch too many horror movies! If you are asking why, read what happened to Lea last week:

Lea loves horror movies and watches vast numbers of them. But last week she exaggerated by watching a 24 hours marathon of zombie apocalypse movies. Watching all these movies without sleeping left her dazed. After the last movie ended (there was no happy end...) she decided to prepare for the zombie apocalypse since it must be coming if there are that many movies about it. She rushed to the closest supermarket and began to buy huge amounts of food.

Now, she realises that she does not know what to buy to be well-prepared. Although in confusion, she still wants to do this right, but the weight she may carry is limited and she fears to go to the supermarket a second time since it may be too late by then. Can you help her once more?

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case starts with a line containing two integers m and n, where m is the weight Lea may carry in grams and n is the number of groceries available. n lines describing the groceries follow. The i-th line contains three integers p_i, l_i and s_i where p_i is the number of packets of this grocery available in the supermarket, l_i is the weight of a packet of this grocery in grams and s_i is the amount of calories per packet.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is a space-separated list of groceries to buy (grocery i may appear at most p_i times in this list). The sum of their weights should be at most m and the sum of their calories should be as big as possible. If there are multiple optimal solutions, any of them will be accepted.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 100$
- $1 \leq m \leq 3 \cdot 10^3$
- $1 \leq p_i \leq 100$ for all $1 \leq i \leq n$
- $1 \leq l_i \leq 100$ for all $1 \leq i \leq n$
- $1 \leq s_i \leq 10^4$ for all $1 \leq i \leq n$

Sample Input 1

```
2
10 2
1 3 5
6 1 1

10 3
2 3 7
1 2 8
3 7 5
```

Sample Output 1

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

Sample Input 2

```
9
11 4
1 1 91
1 3 29
1 3 47
2 1 44

11 3
1 3 80
1 1 7
2 2 83

27 5
2 1 47
2 2 37
1 1 66
1 3 30
1 1 12

29 3
1 2 38
1 3 16
2 3 60

21 5
1 2 86
2 2 89
1 3 3
2 3 51
2 2 20

30 5
2 3 97
1 2 14
2 1 53
2 1 89
1 3 28

21 3
1 3 82
2 1 36
2 3 96

21 5
1 1 3
1 2 11
2 3 15
1 1 44
2 3 2

12 5
1 3 16
1 2 89
1 2 68
2 3 79
1 2 74
```

Sample Output 2

```
Case #1: 1 2 3 4 4
Case #2: 1 2 3 3
Case #3: 1 1 2 2 3 4 5
Case #4: 1 2 3 3
Case #5: 1 2 2 3 4 4 5 5
Case #6: 1 1 2 3 3 4 4 5
Case #7: 1 2 2 3 3
Case #8: 1 2 3 3 4 5 5
Case #9: 2 3 4 4 5
```