10660   Citizen attention offices

The Mayor of a city wants to improve the attentions of the local government to the citizens. For that, five offices to attend to consults will be open in the city. The Mayor thinks that the attentions would be better if the offices are close to where the people live. To decide where to open the offices, the city has been divided in 25 areas, in a square:

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

and the quantity of people living in each square is counted, thus assigning to each area the number (an integer number) of thousands of people living there. For example:

0	0	2	3	1
5	2	2	2	0
1	1	2	3	0
0	0	2	1	5
2	1	2	0	4

The number of people in each area is less or equal to 10 million.

For each area, the minimum distance to the five offices is obtained as follows: the distance from one area to an office is obtained as the number of movements to go from the area to the area where the office will be installed (the movements are in horizontal and vertical; for example, from (1,1) to (2,3) three movements are necessary).

And we want to minimize the sum of the minimum distances from all the areas, but having into account the quantity of people living in each area. Thus, for each area the distance to an office is multiplied by the value associated to the area.

Input

The input begins with a line where the number of test cases (t) is indicated. The data for each test case appear in sucessive lines. In each test case the first line contains the number of areas in the city with non null population (n) and in each one of the n following lines three numbers appear, the first and the second indicate the row and the column of the area (they are numbered from 0 to 4) and the third the number of thousand of people living in the area.

Output

The output consists of a line for each problem, with five numbers with a space between each two numbers, and no space after the last number. The numbers in the solutions are written in increasing order. If there is more than one optimum solution, you must output the solution with lowest first value. If the first value coincides, you must output the solution with lowest second value, and so on.

Sample Input

```
4
1
2 2 1
4
0 0 1
4 4 1
0 4 1
4 0 1
5
0 0 1
1 1 1
2 2 1
3 3 1
4 4 1
7
4 2 2
3 3 1
2 4 3
2 1 1
1 3 4
1 2 2
1 0 1
```

Sample Output

```
0 1 2 3 12
0 1 4 20 24
0 6 12 18 24
5 7 8 14 22
```