Summer Trip

After the whole year of hard studies and competing in the High School Programming League, Julia and Robert are going to spend some time on vacation in the Tricity area (one of the places they want to visit is Pomeranian Science and Technology Park, where the SPOJ servers are located). They have already bought their plane tickets, and everything is planned, but one problem remains: their luggage is too heavy. Due to airline restrictions they can only take 25 kg per person (otherwise there are additional costs).

After some discussion, they have decided to approach the problem as follows: first each of them prepared a separate list of personal items, which must be taken, and then they prepared (together) a list of stuff they would like to take, writing the weight of each item (in kilograms with 0.1 kg precision) and the importance of the item (from their point of view). The sum of the importance of all items will be the criterion based on which they will decide which items are going to be taken.

This way they have an instance of standard 0-1 knapsack problem... well, nearly: they have two knapsacks! Are you able to find an optimal solution?

Input

In the first line, three numbers: $10 < J$, $R < 25$ - the weights of Julia's and Robert's personal effects (floating point numbers with one digit precision), and $0 < N < 100$ - the number of items on the list of things which they would additionally like to take.

N lines follow, with three numbers in each: the number i of the item, its weight $0 < m_i < 15$ expressed in kilograms (a floating point number with one digit precision), and an integer $0 < v_i$ <100 describing its importance.

Output

First S, the optimum total value (according to of Julia and Robert's criterion) of things which can be taken, and in the next two lines, a possible realization of such a scenario according to the following format: in the first line an integer j, the number of items taken by Julia, followed by their numbers, and in the second line an integer r, the number of items taken by Robert, followed by their numbers.

Example

Input:

```
24.7 22.0 4
1 0.1 3
2 0.6 2
3 3.2 12
4 2.4 7
```

Output:

```
12
1 1
2 2 4
```

Scoring

For every correctly solved data set you will receive 2 points, summing to a total of 10 points.