

Picky Eater

time limit per test: 1 second (1 second for Python)
memory limit per test: 976.6 megabytes
input: standard input
output: standard output
points: 100 · problem code: pickyeater · author: glipR

Jasmine the Cat is a very picky eater. Some days she wants wet food, and other days she only has eyes for the dry stuff.



Over a long enough time period, however, Jackson has determined that Jasmine will eat exactly a portions of wet food and b portions of dry food.

Jackson has just received an email detailing all of the different discount bundles on offer at his nearest pet store. For bundle i , he can purchase x_i portions of wet food and y_i portions of dry food for c_i dollars.

Jackson hates food wastage, so he wants to buy a combination of these bundles to receive exactly a portions of wet food, and b portions of dry food.

Jackson naturally also hates money wastage, so he would like to select these bundles to spend the least dollars possible.

Input

Input will begin with three integers n , a and b . These are the number of bundles, the amount of wet food, and dry food that Jasmine will eat respectively.

The next n lines contain the bundle information. Each line contains 3 integers x_i , the amount of wet food in the bundle, y_i , the amount of dry food in the bundle, and c_i , the cost of the bundle. Each bundle can be purchased multiple times, as the pet store has near-unlimited stock.

Output

Output should begin with a single integer, the total amount of dollars spent, C . You should then output n integers on separate lines, corresponding to the amount of times you will use each bundle. If multiple solutions exist, you can output any of them, as long as it buys exactly a wet food, b dry food, and minimises cost.

If it is not possible to purchase exactly a wet food and b dry food, your program should output only the integer `-1`.

Constraints

- All inputs are integers,
- $1 \leq n \leq 100$,
- $0 \leq a, b \leq 200$,
- $0 \leq x_i, y_i \leq 200$,
- $0 \leq c_i \leq 10^9$.

Example 1

Input

```
3 10 10
2 2 20
5 2 0
1 2 12
```

Output

```
56
1
1
3
```

Explanation

In this example, there are multiple ways to purchase the required 10 wet and 10 dry, but the cheapest way involves buying 1 of the first and second bundles, and 3 of the final. Total cost is $20+0+36 = 56$.

Example 2

Input

```
2 0 0
1 2 4
2 1 4
```

Output

```
0
0
0
```

Explanation

In this example, Jasmine doesn't need any food, so it is cheapest not to buy anything!

Example 3

Input

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

Output

```
-1
```

Explanation

In this example, it is impossible to buy exactly 0 wet food and 4 dry food, so we output -1.