B. Discounts

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

One day Polycarpus stopped by a supermarket on his way home. It turns out that the supermarket is having a special offer for stools. The offer is as follows: if a customer's shopping cart contains at least one stool, the customer gets a 50% discount on the cheapest item in the cart (that is, it becomes two times cheaper). If there are several items with the same minimum price, the discount is available for only one of them!

Polycarpus has k carts, and he wants to buy up all stools and pencils from the supermarket. Help him distribute the stools and the pencils among the shopping carts, so that the items' total price (including the discounts) is the least possible.

Polycarpus must use all k carts to purchase the items, no shopping cart can remain empty. Each shopping cart can contain an arbitrary number of stools and/or pencils.

Input

The first input line contains two integers n and k ($1 \le k \le n \le 10^3$) — the number of items in the supermarket and the number of carts, correspondingly. Next n lines describe the items as "c_i t_i" (without the quotes), where c_i ($1 \le c_i \le 10^9$) is an integer denoting the price of the i-th item, t_i ($1 \le t_i \le 2$) is an integer representing the type of item i (1 for a stool and 2 for a pencil). The numbers in the lines are separated by single spaces.

Output

In the first line print a single real number **with exactly one** decimal place — the minimum total price of the items, including the discounts.

In the following k lines print the descriptions of the items in the carts. In the i-th line print the description of the i-th cart as "t b_1 b_2 ... b_t" (without the quotes), where t is the number of items in the i-th cart, and the sequence $b_1, b_2, ..., b_t$ ($1 \le b_j \le n$) gives the indices of items to put in this cart in the optimal distribution. All indices of items in all carts should be pairwise different, each item must belong to exactly one cart. You can print the items in carts and the carts themselves in any order. The items are numbered from 1 to n in the order in which they are specified in the input.

If there are multiple optimal distributions, you are allowed to print any of them.

Examples

input

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

output

```
5.5
2 1 2
1 3
```

input

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

output

```
8.0
1 1
2 4 2
1 3
```

Note

In the first sample case the first cart should contain the 1st and 2nd items, and the second cart should contain the 3rd item. This way each cart has a stool and each cart has a 50% discount for the cheapest item. The total price of all items will be: $2 \cdot 0.5 + (3 + 3 \cdot 0.5) = 1 + 4.5 = 5.5$.