D. Shop

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Vasya plays one very well-known and extremely popular MMORPG game. His game character has k skill; currently the i-th of them equals to a_i. Also this game has a common rating table in which the participants are ranked according to the **product** of all the skills of a hero in the descending order.

Vasya decided to 'upgrade' his character via the game store. This store offers n possible ways to improve the hero's skills; Each of these ways belongs to one of three types:

1. assign the i-th skill to b;
2. add b to the i-th skill;
3. multiply the i-th skill by b.

Unfortunately, a) every improvement can only be used once; b) the money on Vasya's card is enough only to purchase not more than m of the n improvements. Help Vasya to reach the highest ranking in the game. To do this tell Vasya which of improvements he has to purchase and in what order he should use them to make his rating become as high as possible. If there are several ways to achieve it, print any of them.

Input

The first line contains three numbers — k, n, m ($1 \le k \le 10^5$, $0 \le m \le n \le 10^5$) — the number of skills, the number of improvements on sale and the number of them Vasya can afford.

The second line contains k space-separated numbers a_i ($1 \le a_i \le 10^6$), the initial values of skills.

Next n lines contain 3 space-separated numbers t_j, i_j, b_j ($1 \le t_j \le 3$, $1 \le i_j \le k$, $1 \le b_j \le 10^6$) — the type of the j-th improvement (1 for assigning, 2 for adding, 3 for multiplying), the skill to which it can be applied and the value of b for this improvement.

Output

The first line should contain a number l ($0 \le l \le m$) — the number of improvements you should use.

The second line should contain l distinct space-separated numbers v_i ($1 \le v_i \le n$) — the indices of improvements in the order in which they should be applied. The improvements are numbered starting from 1, in the order in which they appear in the input.

Examples

input

```
2 4 3
13 20
1 1 14
1 2 30
2 1 6
3 2 2
```

output

```
3
2 3 4
```