Teleportatinator

SEASON 2021/2022 – FINAL ROUND

Sashka is going on an excursion. She has prepared K backpacks, each of which has a given capacity v_j. Some of them already contain items left over from the previous trip. There are N types of items, with an unlimited amount of each type. Item of type i $(1 \le i \le N)$ brings $s_{i,j}$ pleasure if it is placed in backpack j $(1 \le j \le K)$ and takes up volume w_i (the pleasure of a given item depends on the backpack the item is in). The pleasure of an item may be negative, and it means that it removes $|s_{i,j}|$ pleasure.

It wasn't long before Sashka got exhausted from too much packing, and then, by pure chance, Kyusho appeared out of nowhere with a special device for just such occasions – a teleportatinator. It supports 4 functions, each having a given cost:

1. Buying an item of type i and adding it to backpack j for a cost a.
2. Removing an item of type i from backpack j and discarding it for a cost b.
3. Swapping two items, one of type i from backpack j, the other of type x from backpack y for a cost c. The swap takes place immediately - there isn't a moment when both items are in the same bag together.
4. Taking t items of type i from backpack x and placing them in backpack y for cost $d \times \lfloor\sqrt{t}\rfloor$, where $\lfloor x \rfloor$ denotes the biggest integer smaller or equal to x.

At any time, the sum of the volumes occupied by the items in a backpack should not exceed its capacity, and the items being removed from a backpack should be there in advance. If any of these conditions is not met, the teleportatinator self-destructs and you get Wrong answer.

Let S denote the sum of the pleasures that the items from all backpacks bring (for items that bring pleasures 3,5 and -2, S = 3 + 5 + (- 2) = 6) after the teleportatinator executes all functions; B denotes the sum of pleasures in the beginning; and T denotes the sum of the costs of the functions used. Sashka wants to maximize the difference $(S - B) - T$. As a good friend of Sashka's, Harry decides to help her pack her backpacks by writing a program that finds the optimal strategy, but as you may have guessed, his laziness got in the way. Help Harry by writing a program `teleportatinator` that helps Sashka pack her luggage in an optimal way.

Input

The first line of `teleportatinator.in` contains the integers N, K, a, b, c, d – the number of item types, the number of backpacks and the costs of the functions. The second line of the file contains K numbers v_j – the capacities of the backpacks. The next line contains N numbers w_i – the volumes of the items. Next, there are N lines containing K integers each, where the j-th number on the i-th row is $s_{i,j}$ – the pleasure received by placing the i-th item type in the j-th backpack. Each of the last K lines of the file contains a number p_j followed by p_j

numbers with values from 1 to N – the item types in the backpacks left from the previous trip. It is guaranteed that the volumes they occupy do not exceed the capacities of the backpacks.

Output

On the first line of `teleportatinator.out` print an integer M – the number of functions the teleportatinator must execute. On the next M lines you should print 3 or 5 numbers, depending on the function type – if it's the first or second type, the line should be in the 1 i j or 2 i j format. If the function is of the third type, the format must be 3 i j x y. And if the function is of the fourth type, the format should be 4 t i x y.

Constraints

$2 \leq N \leq 10^4$

$1 \leq K \leq 50$

$0 \leq M \leq 10^6$

$0 \leq a, b \leq 10^6$

$0 \leq c, d \leq 10^5$

$0 \leq |s_{i,j}| \leq 10^6$

$1 \leq v_j, w_i \leq 10^5$

$0 \leq p_1 + p_2 + \ldots + p_K \leq 10^5$

Scoring

The result of the participant is the final difference $(S - B) - T$. Let $yourScore$ be your result and $maxScore$ be the maximum result among all participants. If $yourScore < 0$, you will receive 0 points for the current test. Otherwise, your result for the test will be $\frac{yourScore+1}{maxScore+1} \times points\_for\_the\_test$.

Subtasks

Percent of tests	N	K
20%	**≤ 20**	**≤ 5**
20%	**≤ 200**	**≤ 10**
20%	**≤ 10^3**	**≤ 20**
40%	**≤ 10^4**	**≤ 50**

Fraction of test cases in each subtask	Constraints
1/3	**a=0, b=0**
1/3	**c=0**
1/3	**None**

Teleportatinator

Time Limit: 5.0 sec.
Memory Limit: 1024 MB.

Sample test case

Input (teleportatinator.in)	**Output** (teleportatinator.out)
5 2 6 2 2 3 21 30 6 9 4 13 9 -6 11 -3 8 4 -5 1 8 17 -70 3 2 1 1 1 5	4 3 2 1 5 2 4 2 1 1 2 1 5 1 1 1 2

Explanation: In the end, the first backpack contains two items of type 5, and the second backpack - one of type 2 and three items of type 1. Thus $S = 2 \times 17 + 8 + 3 \times 11 = 75$, $B = -3 + 2 \times (-6) + (-70) = -85$, $T = 2 \times 6 + 2 + 3 \times \lfloor\sqrt{2}\rfloor = 17$. The final result is $(S - B) - T = 143$.