A. Dress'em in Vests!

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

The Two-dimensional kingdom is going through hard times... This morning the Three-Dimensional kingdom declared war on the Two-dimensional one. This (possibly armed) conflict will determine the ultimate owner of the straight line.

The Two-dimensional kingdom has a regular army of n people. Each soldier registered himself and indicated the desired size of the bulletproof vest: the i-th soldier indicated size a_i. The soldiers are known to be unpretentious, so the command staff assumes that the soldiers are comfortable in any vests with sizes from $a_i - x$ to $a_i + y$, inclusive (numbers $x, y \geq 0$ are specified).

The Two-dimensional kingdom has m vests at its disposal, the j-th vest's size equals b_j. Help mobilize the Two-dimensional kingdom's army: equip with vests as many soldiers as possible. Each vest can be used only once. The i-th soldier can put on the j-th vest, if $a_i - x \leq b_j \leq a_i + y$.

Input

The first input line contains four integers n, m, x and y ($1 \leq n, m \leq 10^5$, $0 \leq x, y \leq 10^9$) — the number of soldiers, the number of vests and two numbers that specify the soldiers' unpretentiousness, correspondingly.

The second line contains n integers $a_1, a_2, ..., a_n$ ($1 \leq a_i \leq 10^9$) in non-decreasing order, separated by single spaces — the desired sizes of vests.

The third line contains m integers $b_1, b_2, ..., b_m$ ($1 \leq b_j \leq 10^9$) in non-decreasing order, separated by single spaces — the sizes of the available vests.

Output

In the first line print a single integer k — the maximum number of soldiers equipped with bulletproof vests.

In the next k lines print k pairs, one pair per line, as "u_i v_i" (without the quotes). Pair (u_i, v_i) means that soldier number u_i must wear vest number v_i. Soldiers and vests are numbered starting from one in the order in which they are specified in the input. All numbers of soldiers in the pairs should be pairwise different, all numbers of vests in the pairs also should be pairwise different. You can print the pairs in any order.

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

Examples

input

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

output

```
2
1 1
3 2
```

input

```
3 3 2 2
1 5 9
3 5 7
```

output

```
3
1 1
```

```
2 2
3 3
```

Note

In the first sample you need the vests' sizes to match perfectly: the first soldier gets the first vest (size 1), the third soldier gets the second vest (size 3). This sample allows another answer, which gives the second vest to the fourth soldier instead of the third one.

In the second sample the vest size can differ from the desired size by at most 2 sizes, so all soldiers can be equipped.

```
2 2
3 3
```

Note

In the first sample you need the vests' sizes to match perfectly: the first soldier gets the first vest (size 1), the third soldier gets the second vest (size 3). This sample allows another answer, which gives the second vest to the fourth soldier instead of the third one.

In the second sample the vest size can differ from the desired size by at most 2 sizes, so all soldiers can be equipped.