D. Professor GukiZ and Two Arrays

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

Professor GukiZ has two arrays of integers, a and b. Professor wants to make the sum of the elements in the array a s_a as close as possible to the sum of the elements in the array b s_b. So he wants to minimize the value $v = |s_a - s_b|$.

In one operation professor can swap some element from the array a and some element from the array b. For example if the array a is $[5, 1, 3, 2, 4]$ and the array b is $[3, 3, 2]$ professor can swap the element 5 from the array a and the element 2 from the array b and get the new array a $[2, 1, 3, 2, 4]$ and the new array b $[3, 3, 5]$.

Professor doesn't want to make more than two swaps. Find the minimal value v and some sequence of no more than two swaps that will lead to the such value v. Professor makes swaps one by one, each new swap he makes with the new arrays a and b.

Input

The first line contains integer n ($1 \le n \le 2000$) — the number of elements in the array a.

The second line contains n integers a_i ($-10^9 \le a_i \le 10^9$) — the elements of the array a.

The third line contains integer m ($1 \le m \le 2000$) — the number of elements in the array b.

The fourth line contains m integers b_j ($-10^9 \le b_j \le 10^9$) — the elements of the array b.

Output

In the first line print the minimal value $v = |s_a - s_b|$ that can be got with no more than two swaps.

The second line should contain the number of swaps k ($0 \le k \le 2$).

Each of the next k lines should contain two integers x_p, y_p ($1 \le x_p \le n$, $1 \le y_p \le m$) — the index of the element in the array a and the index of the element in the array b in the p-th swap.

If there are several optimal solutions print any of them. Print the swaps in order the professor did them.

Examples

input
5 5 4 3 2 1 4 1 1 1 1

output
1 2 1 1 4 2

input
5 1 2 3 4 5 1 15

output

```
0
0
```

input

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

output

```
1
1
3 1
```