B. Dreamoon Likes Permutations

time limit per test: 2 seconds
memory limit per test: 256 megabytes

The sequence of m integers is called the *permutation* if it contains all integers from 1 to m exactly once. The number m is called the length of the permutation.

Dreamoon has two permutations p_1 and p_2 of non-zero lengths l_1 and l_2.

Now Dreamoon concatenates these two permutations into another sequence a of length $l_1 + l_2$. First l_1 elements of a is the permutation p_1 and next l_2 elements of a is the permutation p_2.

You are given the sequence a, and you need to find two permutations p_1 and p_2. If there are several possible ways to restore them, you should find all of them. (Note that it is also possible that there will be no ways.)

Input

The first line contains an integer t ($1 \le t \le 10\,000$) denoting the number of test cases in the input.

Each test case contains two lines. The first line contains one integer n ($2 \le n \le 200\,000$): the length of a. The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n - 1$).

The total sum of n is less than $200\,000$.

Output

For each test case, the first line of output should contain one integer k: the number of ways to divide a into permutations p_1 and p_2.

Each of the next k lines should contain two integers l_1 and l_2 ($1 \le l_1, l_2 \le n, l_1 + l_2 = n$), denoting, that it is possible to divide a into two permutations of length l_1 and l_2 (p_1 is the first l_1 elements of a, and p_2 is the last l_2 elements of a). You can print solutions in any order.

Example

input Copy

```
6
5
1 4 3 2 1
6
2 4 1 3 2 1
4
2 1 1 3
4
1 3 3 1
12
2 1 3 4 5 6 7 8 9 1 10 2
3
1 1 1
```

output Copy

```
2
1 4
4 1
1
4 2
0
0
1
2 10
0
```

Note

In the first example, two possible ways to divide a into permutations are $\{1\} + \{4, 3, 2, 1\}$ and $\{1, 4, 3, 2\} + \{1\}$.

In the second example, the only way to divide a into permutations is $\{2, 4, 1, 3\} + \{2, 1\}$.

In the third example, there are no possible ways.

Codeforces Round 631 (Div. 2) - Thanks, Denis aramis Shitov!

Finished

Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++20 13.2 (64 bit, win

Choose file: Choose File No file chosen

Submit

→ Last submissions

Submission	Time	Verdict
281379473	Sep/15/2024 18:55	Accepted
281301578	Sep/15/2024 07:07	Accepted
281301429	Sep/15/2024 07:05	Accepted
281300586	Sep/15/2024 06:52	Wrong answer on test 2
281300521	Sep/15/2024 06:50	Wrong answer on test 2

→ Problem tags

implementation math *1400

No tag edit access

→ Contest materials

- Announcement (en)
- Tutorial (en)