Problem
Strange Sorting 3
Time limit: 4 seconds

Little Cyan Fish has a permutation $p_1, p_2, \ldots, p_n$ of the integers from 1 to n, and, as the title of the problem suggests, he wants to sort it in ascending order. This time, the operation he can perform is as follows.

- In one operation, Little Cyan Fish may choose two integers i and j such that $1 \leq i < j \leq n$ and swap p_i and p_j.

But, come on, everyone knows how to sort a permutation with simple swaps, so obviously this is not what the task asks you to do. Instead, Little Cyan Fish defines the *distance* of an operation to be $j - i$. For a sequence of operations that sorts the permutation, let a_d denote the number of operations of distance d, for every $1 \leq d < n$.

Among all finite sequences of operations that sort the permutation, Little Cyan Fish wants you to find the lexicographically smallest sequence $(a_1, a_2, \ldots, a_{n-1})$, and construct a sequence of **at most** $5n$ operations that achieves it.

Input

There are multiple test cases. The first line of the input contains a single integer T ($1 \leq T \leq 10^6$), indicating the number of test cases.

For each test case, the first line contains a single integer n ($2 \leq n \leq 10^6$), indicating the length of the permutation. The second line contains n integers $p_1, p_2, \ldots, p_n$ ($1 \leq p_i \leq n$), indicating the permutation. It is guaranteed that p is a permutation of length n.

It is guaranteed that the sum of n over all test cases does not exceed 10^6.

Output

For each test case, output the answer in the following format.

The first line for each test case should contain $n - 1$ integers $a_1, a_2, \ldots, a_{n-1}$, the lexicographically smallest sequence.

The second line should contain a single integer k ($0 \leq k \leq 5n$), the number of operations in your sequence.

Each of the following k lines should contain two integers i and j ($1 \leq i < j \leq n$), the indices chosen in the corresponding operation. The operations, performed in the given order, must sort the input permutation, and for every $1 \leq d < n$, exactly a_d of the operations must have distance d.

It can be shown that an optimal sequence of at most $5n$ operations always exists. If there are multiple valid sequences of operations, you may output any of them.

Sample Input 1	Sample Output 1
3 3 3 2 1 4 2 1 3 4 4 1 3 2 4	0 1 1 1 3 0 1 2 3 1 4 2 4 1 4 0 3 2 5 1 3 1 4 2 4 1 4 1 3

In the first sample test case, Little Cyan Fish swaps positions 1 and 3. This sorts the permutation using one operation of distance 2.

In the second sample test case, one optimal sequence is $(1, 4)$, $(2, 4)$, $(1, 4)$. The distances are 3, 2, and 3, so the resulting sequence is $(0, 1, 2)$.

In the third sample test case, one optimal sequence is $(2, 4)$, $(1, 4)$, $(1, 3)$, $(1, 4)$, $(2, 4)$. Its distance counts are $(0, 3, 2)$.