P3. Crane

(Time Limit: 3 seconds)

There are n crates waiting to be loaded onto a ship. The crates are numbered 1, 2, . . . , n, the numbers determining the order of loading. Unfortunately, someone messed up the transit and the crates are standing in a row in an arbitrary order. As there is only limited space in the dock area, you must sort the crates by swapping some of them.

You are given a crane that works in the following way: you select a connected interval of crates of even length. The crane then exchanges the first half of the interval with the second half. The order inside both halves remains unchanged. Determine the sequence of crane moves that reorders the crates properly.

The crane's software has a bug: the move counter is a 9-based (not 10-based, as you might think) integer with at most 6 digits. Therefore, the crane stops working (and has to be serviced) after 9^6 = 531441 moves. Your solution must fit within this limit.

Input

The first line of input contains the number of test cases T. The descriptions of the test cases follow: Each test case starts with an integer n, $1 \le n \le 10000$, denoting the number of crates. In the next line a permutation of numbers {1, 2, . . . , n} follows.

Output

For each test case print a single line containing m — the number of swaps — followed by m lines describing the swaps in the order in which they should be performed. A single swap is described by two numbers — the indices of the first and the last element in the interval to be exchanged. Do not follow the crane's strange software design — use standard decimal numeral system.

Sample Input

```
2
6
5 4 6 3 2 1
5
1 2 3 4 5
```

Sample Input

5

1 2

4 5

5 6

4 5

1 6

0

P4. Blocks

(Time Limit: 3 seconds)

Some of you may have played a game called ‘Blocks’. There are n blocks in a row, each box has a color. Here is an example: Gold, Silver, Silver, Silver, Silver, Bronze, Bronze, Bronze, Gold.

The corresponding picture will be as shown below:

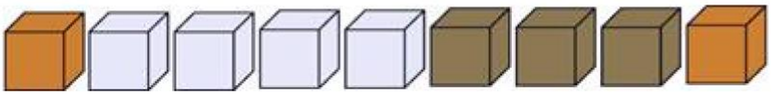

Figure 1

If some adjacent boxes are all of the same color, and both the box to its left (if it exists) and its right (if it exists) are of some other color, we call it a Žbox segment’. There are 4 box segments. That is : gold, silver, bronze, gold. There are 1, 4, 3, 1 box(es) in the segments respectively.

Every time, you can click a box, then the whole segment containing that box DISAPPEARS. If that segment is composed of k boxes, you will get k ∗ k points. for example, if you click on a silver box, the silver segment disappears, you got 4 ∗ 4 = 16 points.

Now lets look at the picture below:

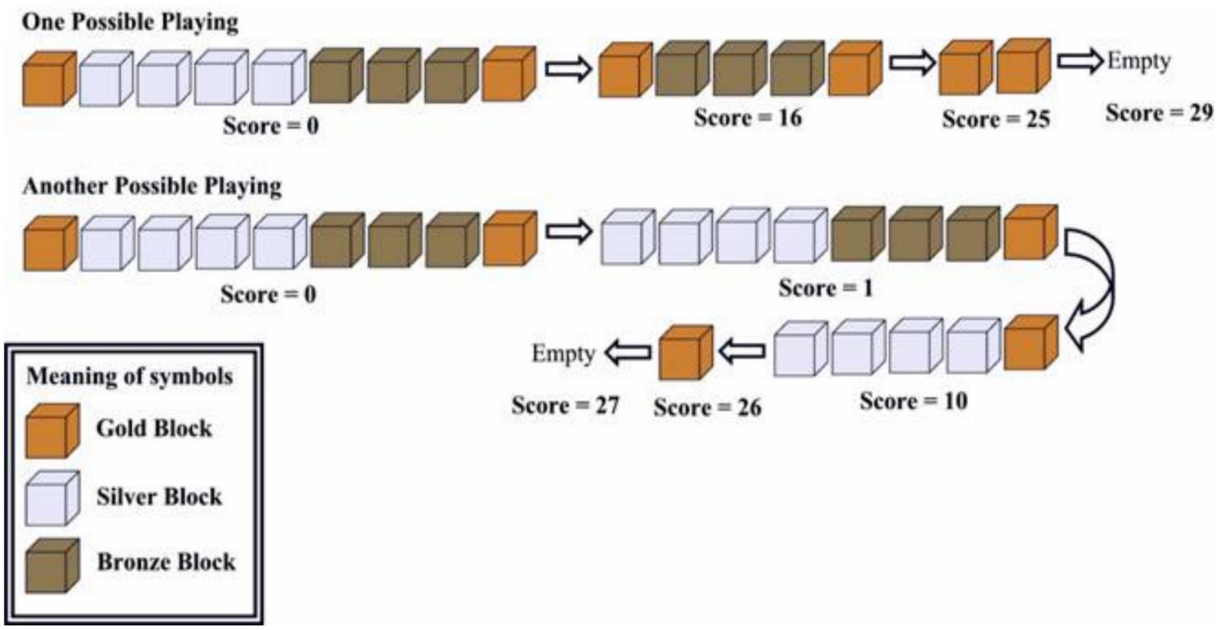

Figure 2

The first one is OPTIMAL.

Find the highest score you can get, given an initial state of this game.

Input

The first line contains the number of tests t (1 ≤ t ≤ 15). Each case contains two lines. The first line contains an integer n (1 ≤ n ≤ 200), the number of boxes. The second line contains n integers, representing the colors of each box. The integers are in the range 1 ∼ n.

Output

For each test case, print the case number and the highest possible score.

Sample Input

```
2
9
1 2 2 2 2 3 3 3 1
1
1
```

Sample Output

```
Case 1: 29
Case 2: 1
```