Problem 1: Sinthia's Game Adventure (I Dont Like This Game)

Problem Statement:

Sinthia, with her classmates, is playing a game in the classroom called 'I Dont Like This Game'.

In this game, N players are given a number between 1 to N, and placed in a line in a specific order. For example, for 3 people, [3, 2, 1], [2, 3, 1], [1, 2, 3], and so on are the valid order of lines. Whenever a player is at the front of the line, it is his/her turn. When a player's turn comes, if his/her number is the smallest in the line, he/she will scream, "The wait is over!!" and leave the line. Otherwise, he/she will scream, "I don't like this game" and then move to the end of the line. Then the next player's turn will come. The game will end when there is no one remaining in the line.

Your task is to find the number of times the phrase, "I don't like this game" will be screamed in the classroom before the game ends.

Note: See sample input-output for better understanding.

Input:
- The first line of the input will have an integer 'N' (1 <=N <= 1000): the number of players
- The second line of the input will be a list of 'N' space-separated integers a[i] (1 <=a[i] <= N): the numbers given to the players

Output:

The output will be a single integer, the number of times the phrase, "I don't like this game" screamed in the classroom.

Examples:

Input	Output
2 1 2	0

3 3 2 1	3
3 2 1 3	2
8 3 2 4 7 8 1 6 5	18
5 1 2 3 5 4	1

Explanation:

In the second example:

Line: [3 2 1]->The player in the front has the number 3, which is not the smallest number in the line, so he/she will scream the phrase and move to the end of the line. phraseScream = 1.

Line: [2 1 3]->Then the player with the number 2 will be in the front, which is still not the smallest in the line, so he/she will do the same, and move to the end of the line. phraseScream = 2.

Line: [1 3 2]->Now, the smallest number player is in the front, so he/she will scream, "The wait is over!!", and leave the line. phraseScream = 2.

Line: [3 2]->Now, again, the player with the number 3 is in the front and not the smallest in the line. So, he/she will again scream the phrase "I don't like this game" and move to the end of the line. phraseScream = 3.

Line: [2 3]->Now, the player in front has the number 2, which is the smallest in the line, so he/she will scream, "The wait is over!!" and leave the line. phraseScream = 3.

Line: [3]->Now again, the player with the number 3 is in the front, but this time, it is the smallest in the line, so he/she will scream, "The wait is over!!" and leave the line.

Line: []->As no players are remaining in the line, the game is over.

Problem 2: Sinthia's Game Adventure (It Was You)

Problem Statement:

Sinthia and her classmates did not like the previous game that much. They all are tired of screaming and moving in the game. Now all of them are sitting at their desks and do not want to move anymore. They do not know for sure whose idea it was to play the game, but they want to blame it on someone. So they point the finger to the next player and say "It was you!". For example, if 3 players are given numbers, [2 1 3] in the previous game, in that order, i.e. player 1 has number 2, player 2 has number 1, and player 3 has number 3, player 1 points to player 2, player 2 points to player 3 and player 3 points to no one.

In this way, the last player always ends up pointing to no one. This makes him/her sad.

For this reason, he/she was given the power to choose any player from the line with whom he/she can swap his/her pointings(points to and pointed by) without changing his/her sitting position, i.e. the last player will now point to the player that the chosen player was pointing, and the chosen player will point to no one.

Your task is to use a data structure that will help them to do as described above and show their pointing order in the end.

Note 1: They can not change their places, but only their pointings.
Note 2: See sample input-output for better understanding.

Input:
- The first line of the input will have an integer 'N' (1 <=N <= 1000): the number of players
- The second line of the input will be a list of 'N' space-separated integers a[i] (1 <=a[i] <= N): the numbers given to the players
- The third line of the input will be an integer 'B' ((1 <=B <= N): The chosen player by the last player.

Output:

The output will be a single line containing the order in which the players are pointing at each other after the last player chooses a player to swap his/her pointings.

Examples:

Input	Output
5 3 2 1 5 4 3	3 2 4 5 1
4 1 2 3 4 1	4 2 3 1
10 8 7 6 2 3 4 1 9 10 5 4	8 7 6 5 3 4 1 9 10 2

Explanation:

In the first example:

Player	1	2	3	4	5
Number	3	2	1	5	4

So the pointing is like below:

      3->2->1->5->4->

The last player, player 5, who has the number 4, is pointing to no one. So he chooses the third player to swap.
After swapping, the pointing is like below:

      3->2->4->5->1->

So they swapped their pointings.

Problem 3: Sinthia's Game Adventure (One Rod Hanoi)

Problem Statement:

Sinthia likes playing games that consist of logical thinking. She recently learned about The Tower of Hanoi game.

The Tower of Hanoi is a mathematical game or puzzle consisting of three rods and some disks of various diameters, which can slide onto any rod. The puzzle begins with the disks stacked on one rod in order of decreasing size, the smallest at the top, thus approximating a conical shape. The objective of the puzzle is to move the entire stack to one of the other rods, obeying the following rules:

1. Only one disk may be moved at a time.
2. Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack or an empty rod.
3. No disk may be placed on top of a disk that is smaller than it.

She played that game a number of times with 3 rods and 3 disks. With three disks, the puzzle can be solved in seven moves. The minimal number of moves required to solve a Tower of Hanoi puzzle is $2^n - 1$, where n is the number of disks.

Now she wants to create a modified version of the game. She named the game 'One Rod Hanoi'. Rules of the games are given below:

1. There is only one rod.
2. There are **'n'** numbers of disks and will be given one by one.
3. In one move, a disk can be placed or taken out.
4. Whenever a disk is given, it has to be placed in the rod in a way that all the disks are in decreasing order, the largest at the bottom and the smallest at the top.
5. If it is not possible to place a disk because of the size difference, the required number of disks can be taken out and placed back again.
6. All the previously given disks have to be in order before a new disk is given.

Your task is to help Sinthia find the number of moves required to place **'n'** number of disks of different diameters in the rod and show the order of the disks from bottom to top at the end of the game.

Note: See sample input-output for better understanding.

Input:
- The first line of the input will have an integer 'N' (1 <=N <= 1000): the number of disks.
- The second line of the input will be a list of 'N' space-separated integers a[i] (1 <=a[i] <= 1000): representing the diameters of the disks..

Output:
The output will consist of two lines. The first line will be a single integer, the number of moves required to place all the disks in one rod. The second line will be the order of the disks after their placement in the rod.

Examples:

Input	Output
3 2 1 3	7 3 2 1
3 100 4 1	3 100 4 1
3 1 4 100	9 100 4 1
6 3 29 13 2 4 7	20 29 13 7 4 3 2
10 98 99 5 72 79 52 55 27 60 93	48 99 98 93 79 72 60 55 52 27 5
5 5 4 3 2 1	5 5 4 3 2 1

Explanation:

In the first example:

There are 3 disks given one by one.

The first disk's diameter is 2. As there are no disks in the rod. So, it will be placed. Moves=1, Disk's Order: 2.

The second disk's diameter is 1. There is a disk in the rod with a diameter of 2, as the size of the current disk is smaller, it can be placed. Moves=2, Disk's Order: 1, 2.

The third disk's diameter is 3. The top disk's diameter in the rod is 1, which is smaller than the current disk, so it needs to be taken out. Moves=3, Disk's Order: 2.

Now, the top disk in the tower is of diameter 2, which is still smaller than 3 and needs to be taken out. Moves=4, Disk's Order: .

Now, there is no disk larger than 3 in the rod, so it can be placed. Moves=5, Disk's Order: 3.

Now, we can place 2 again. Moves=6. Disk's Order: 2, 3.

Now, we can place 1 again. Moves=7. Disk's Order: 1, 2, 3.