Chef recorded a video explaining his favorite recipe. However, the size of the video is too large to upload on the internet. He wants to compress the video so that it has the minimum size possible.

Chef's video has *N* frames initially. The value of the ��ℎ*ith* frame is ��*Ai*​. Chef can do the following type of operation **any** number of times:

* Choose an index *i* (1≤�≤�)(1≤*i*≤*N*) such that the value of the ��ℎ*ith* frame is **equal** to the value of **either** of its neighbors and **remove** the ��ℎ*ith* frame.

Find the **minimum** number of frames Chef can achieve.

**Input Format**

* First line will contain �*T*, the number of test cases. Then the test cases follow.
* The first line of each test case contains a single integer �*N* - the number of frames initially.
* The second line contains �*N* space-separated integers, �1,�2,…,��*A*1​,*A*2​,…,*AN*​ - the values of the frames.

**Output Format**

For each test case, output in a single line the **minimum** number of frames Chef can achieve.

**Sample 1:**

Input

Output

4

1

5

2

1 1

3

1 2 3

4

2 1 2 2

1

1

3

3

**Explanation:**

**Test case 11:** There is only one frame with value 55. Since there are no neighbors, Chef won't remove any frame and the minimum number of frames Chef can achieve is 11.

**Test case 22:** There are two frames where both frames have value 11. Chef can remove the first frame as the value of the first frame is equal to that of the second frame. The remaining frames have values [1][1]. The minimum number of frames Chef can achieve is 11.

**Test case 33:** There are 33 frames. All frames have distinct values. Thus, the minimum number of frames Chef can achieve is 33.

**Test case 44:** Chef can remove the fourth frame as the value of the fourth frame is equal to that of the third frame. The remaining frames have values [2,1,2][2,1,2]. Thus, the minimum number of frames Chef can achieve is 33.