

Wesley's Anger Contest 5 Problem 3 - Super Squirrel Sisters

Time limit: 1.5s **Memory limit:** 256M

Java: 2.5s

Python: 2.5s

Wesley is playing Super Squirrel Sisters! In this game, Wesley has to play through N levels in order, and completing the i^{th} level rewards you with the b_i bonus which are all labelled from 1 to N . Wesley has already beaten this game multiple times, so his main objective now is to get the rumoured secret item.

The secret item is said to only appear if the player earns a **non-empty contiguous subsequence** of bonuses where the number of distinct bonuses earned is equal to the number of occurrences of every bonus in the subsequence. An example of a valid sequence of bonuses earned is `1 2 1 1 2 3 3 2 3`, where the player earns all three bonuses 1, 2, and 3 three times each. An invalid sequence of bonuses earned is `2 1 1 2 2`, where the player earned one too many bonuses labelled as 2.

Wanting to play the game in every possible way, Wesley wants to find how many valid contiguous subsequences there are in this game. Can you help him out?

For this problem, Python users are recommended to use PyPy over CPython.

Constraints

For this problem, you will NOT be required to pass all the samples in order to receive points. In addition, all subtasks are disjoint, and you are NOT required to pass previous subtasks to earn points for a specific subtask.

For all subtasks:

$$1 \leq N \leq 200\,000$$

$$1 \leq b_i \leq N \text{ for all } 1 \leq i \leq N$$

Subtask 1 [16%]

$$1 \leq N \leq 100$$

Subtask 2 [17%]

$$1 \leq N \leq 5\,000$$

Subtask 3 [25%]

$$1 \leq b_i \leq \min(N, 2) \text{ for all } 1 \leq i \leq N$$

Subtask 4 [42%]

No additional constraints.

Input Specification

The first line contains an integer N indicating the number of levels in the game.

The next line contains N integers separated with spaces indicating the b_i bonus earned by completing the i^{th} level.

Output Specification

This problem is graded with an `identical` checker. This includes whitespace characters. Ensure that every line of output is terminated with a `\n` character and that there are no trailing spaces.

Output, on a single line, the number of valid contiguous subsequences.

Sample Input 1

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

Sample Output 1

```
12
```

Sample Explanation 1

From the 12 possible subsequences, 9 of them have a size of 1. The other three are `2 1 1 2`, `2 3 3 2`, and `1 2 1 1 2 3 3 2 3`.

Sample Input 2

```
6
2 2 1 2 1 1
```

Sample Output 2

```
7
```

Sample Explanation 2

From the 7 possible subsequences, 6 of them have a size of 1. The other one is `2 1 2 1`.