Problem C
A Sorting Problem

Time limit: 3 seconds

Memory limit: 2048 megabytes

Problem Description

You are given an array $[p[1], p[2], ..., p[n]]$ where all the numbers in the array are distinct. In addition, the numbers are positive integers between 1 and n. You can only perform the following operations on the array: Pick two indices x and y such that $|p[x] - p[y]| = 1$, and then swap the values of $p[x]$ and $p[y]$. We now want to sort this array in ascending order. That is, to make $p[i] = i$ for all $i \in \{1, 2, \ldots, n\}$. For example, we can sort the array $[p[1] = 2, p[2] = 3, p[3] = 1]$ in two operations:

1. Swap $p[1]$ and $p[3]$. The array becomes $[p[1] = 1, p[2] = 3, p[3] = 2]$.

2. Swap $p[2]$ and $p[3]$. The array becomes $[p[1] = 1, p[2] = 2, p[3] = 3]$ which is sorted in ascending order.

Please write a program to compute the minimum number of operations to sort a given array in ascending order.

Input Format

The input contain two lines. The first line contains one integer n. The second lines contain n space-saparated numbers $p[1], p[2], \ldots, p[n]$ representing the array $[p[1], p[2], \ldots, p[n]]$

Output Format

Output only one number that denotes the minimum number of operations required to sort the given array.

Technical Specification

- $1 < n \leq 200000$.
- $1 \leq p[i] \leq n$.
- All $p[i]$ are distinct.

Sample Input 1

```
3
1  3  2
```

Sample Output 1

```
1
```

Sample Input 2

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

Sample Output 2

```
7
```