**Problem 2. Permutation**

Ivancho has already understood how important for a future programmer is to know combinatorics. He has been solving problems using combinatorics for weeks. Now he wants to improve his algorithmic background too, especially algorithms for sorting. The basic sort algorithms aren’t hard for Ivancho but when sorting is mixed with combinatorics your friend has big issues. Therefore, he needs you to help him.

Here is the problem. You got an array of numbers, which is a permutation of the numbers from 1 to N. The operation you can use is as follows: you take an element from the array and place it at the end of the array. What is the minimum number of operations needed to sort the array in increasing order.

 **Input**: On the first line of the input file **permutation.in** will be written N. Follow N integers – the members of the array.

**Output**: The output file **permutation.out** must contain a single integer – the minimum number of operations needed to sort the array.

**Constraints:**

1$\leq $ N$\leq $1000

**TIME LIMIT – 2 sec**

 **Note**: You can scan from a file and print into a file by using an operator for that. You can use operator freopen including fstream library and adding the following two lines in the beginning of your main function:

freopen ( "permutation.in", "r", stdin );

 freopen ( “permutation.out”, “w” , stdout );

**Example:**

|  |  |
| --- | --- |
| **permutation.in** | **permutation.out** |
| 101 8 2 4 6 3 7 5 9 10 | 7 |