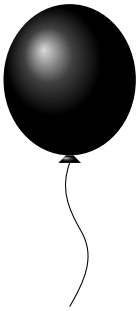
C Diligent Johnny



Time limit: 1.5s
Memory limit: 512MB

It is Little Johnny's birthday. And this is serious algorithmic problem, so the poor kid received no toys, games or a computer for his birthday present. Rather, he was presented with long arrays filled with numbers, trees, maps of strange lands rife with roads that lead through numerous tunnels and overpasses, lengthy tapes filled with 1048576 symbols long prefixes of Fibonacci and Thue-Morse words, etc. Of all these educational gifts, he likes an array holding a permutation of the first n positive integers the most. Soon, Johnny started wondering what is the lexicographic predecessor of the one he was given. Having figured that out rather quickly, Johnny immediately asked himself how could he write this predecessor permutation in his array. The only operation that the array supports is selecting two cells and swapping the contents of those cells. Fortunately, Johnny was smart enough to transform the initial permutation into its predecessor in the minimum number of swaps. He found this task so captivating, that he kept on transforming each successive permutation into its predecessor.

In his permutation madness, Johnny is ignoring all his birthday party guests, which they find amusing enough but also a little rude. One of them soon realized that Johnny will stop once he gets down to the identity permutation $1, 2, \ldots, n$, which is lexicographically smallest. The question is, how long will this take?

Help them answer this question, knowing that every swap takes Johnny exactly one second. As this might take a while (diligent is Johnny's middle name), the guests will be happy enough to know the remainder of division by $10^9 + 7$. After all, they can check back on Johnny every $10^9 + 7$ seconds to see if he is finally done.

A permutation of the numbers from 1 to n is a sequence of pairwise different integers $p_1, \ldots, p_n$ satisfying $1 \leq p_i \leq n$ (i.e., every integer from 1 to n appears exactly once in a permutation).

The permutation $P = (p_1, \ldots, p_n)$ is lexicographically smaller than the permutation $Q = (q_1, \ldots, q_n)$ (which we denote $P < Q$) if $p_j < q_j$, where j is the smallest index such that $p_j \neq q_j$. The permutation P is the lexicographic predecessor of Q if $P < Q$ and there exists no permutation R such that $P < R < Q$.

Input

In the first line of the input, there is a single integer n — specifying the length of the permutation that Johnny got for his birthday ($1 \leq n \leq 1\,000\,000$).

In the second line, the permutation itself is given, as a sequence of n distinct integers $p_1, p_2, \ldots, p_n$, separated by single spaces ($1 \leq p_i \leq n$).

Output

Print the remainder of division by $10^9 + 7$ of the number of swaps that Johnny will make before he stops.

Samples

Sample input 1	Sample output 1
3 3 1 2	6