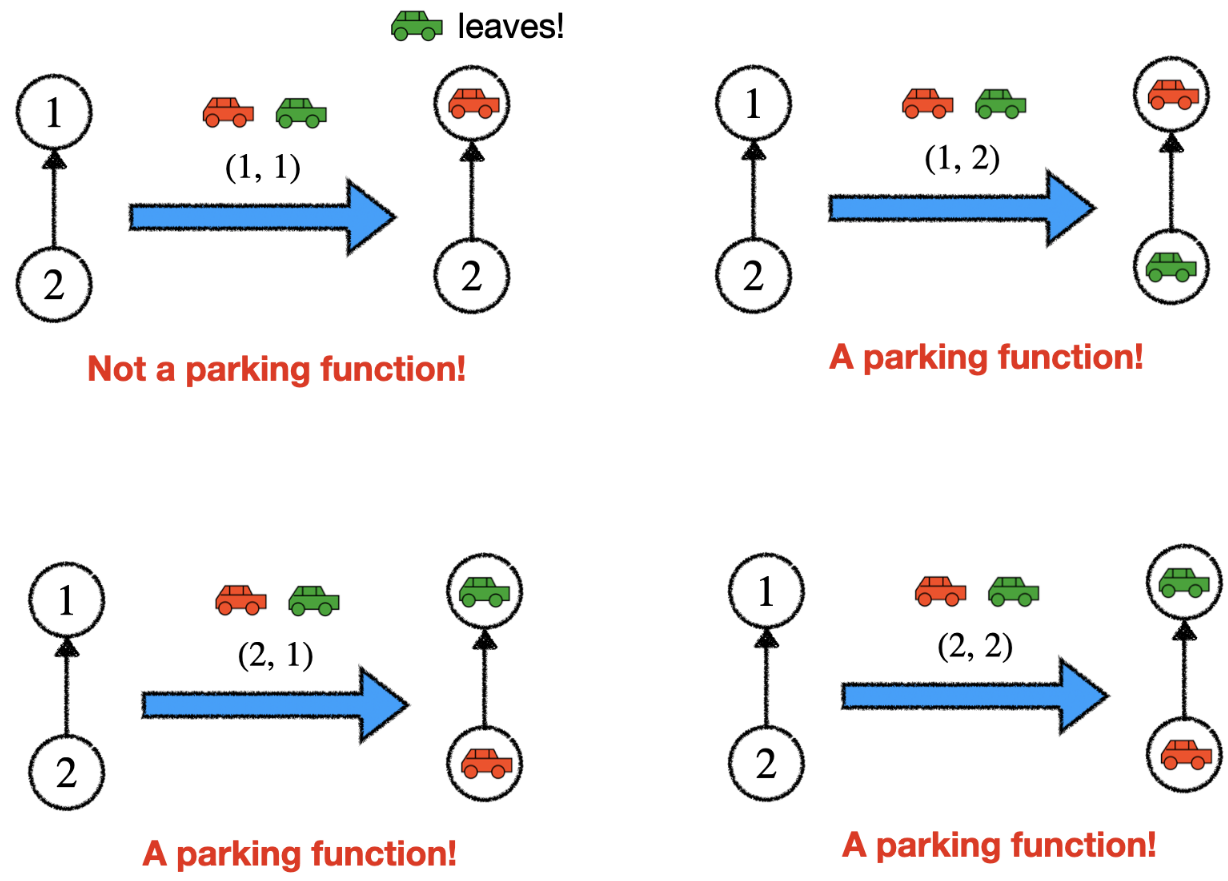
Random Tree Parking

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	1024 megabytes

Parking functions are combinatorial objects originally introduced by Konheim and Weiss during their studies of the so-called linear probing collision resolution scheme for hash tables. As a crazy fan of mathematics, Bobo now wants to investigate parking functions on trees.

Consider a parking lot in the form of a tree T, with labels in $[n] = \{1, 2, \dots, n\}$, rooted at vertex 1. The edges of the tree are oriented towards the root vertex. We have a sequence of n drivers, each with their preferred parking space, which is a vertex in the tree. The drivers arrive one by one, and each driver i ($1 \le i \le n$) tries to park at their preferred space $s_i \in [n]$. If the space is free, they park there. Otherwise, they follow the edges towards the root vertex and park at the first empty vertex they encounter. If there are no empty vertices, they leave the parking lot, i.e., the tree, without parking.

A sequence $\mathbf{s} \in [n]^n$ of vertices is then called a parking function of the tree T if all drivers are successful, i.e., if all drivers are able to find a parking space. For example, for the following tree T with two vertices, there are three parking functions, namely $(1, 2)$, $(2, 1)$ and $(2, 2)$.



Now, given a tree T with n vertices, Bobo wonders what is the number of parking functions of T. As Bobo is also obsessed with randomness, he decided that the tree T should be generated **randomly** in the following way:

- For $i = 2, \dots, n$, the vertex i points to is chosen independently uniformly at random from $\{1, 2, \dots, i - 1\}$.

Since the answer might be too large, you should output the answer modulo 998 244 353 (a prime number).

Input

The first line of input contains an integer n ($2 \leq n \leq 10^5$), denoting the size of the tree.

The next line contains $n-1$ integers $p_2, p_3, \ldots, p_n$ ($1 \leq p_i \leq i-1$), where p_i denotes the unique vertex i points to in T. **You can assume that each p_i is chosen independently uniformly at random from** $\{1, 2, \ldots, i-1\}$.

It is guaranteed that there are at most 20 test sets for this problem.

Output

Output an integer in a line, denoting the number of parking functions of T, taken modulo 998 244 353.

Examples

standard input	standard output
3 1 1	12
3 1 2	16
4 1 2 3	125
8 1 2 3 1 3 4 3	1198736
15 1 2 2 2 2 3 3 2 7 7 3 10 3 13	938578089