

Permutation

Input file:permutation.in

Output file:permutation.out

Time limit:1 second

Memory limit:1024 mebibytes

Compiling:-O2 -std=c++14

给定正整数 n 和一个数组 a_1, a_2, \dots, a_n 。请你计数有多少 $1 \sim n$ 的排列 P ，满足：

- 令 M 是一个 $n \times n$ 的矩阵，满足 $M_{i,j} = \begin{cases} 1 & j = P_i; \\ 0 & j \neq P_i \end{cases}$;
- 对每个 $M_{i,j} = 1$ 的格子 (i,j) ，**不存在** M 的 $(a_j + 1) \times (a_j + 1)$ 子矩阵（行和列均连续）**覆盖** (i,j) ，满足 M 是恒等矩阵（主对角线为 1，其余为 0）或恒等矩阵的镜像（反对角线为 1，其余为 0）。

答案对 998244353 取模。

Input

第一行一个正整数 n 。
第二行 n 个正整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$)。提示： $a_i = n$ 相当于没有限制。

Output

一行一个整数表示答案，对 998244353 取模。

Constraints

$1 \leq n \leq 5000.$

Subtask 1 (points: 10)

$n \leq 10.$

Subtask 2 (points: 30)

$n \leq 50.$

Subtask 3 (points: 30)

$n \leq 400.$

Subtask 4 (points: 30)

没有特殊性质。

Examples

permutation.in	permutation.out
3 1 2 3	2
4 4 2 3 2	18
5 1 4 4 4 3	70