

# 汉诺塔

---

时间限制：1.0s 内存限制：64M

---

## 题目描述

---

简述题意：

用两只手玩汉诺塔，最小化拿起次数。

形式化题意：

你有三个栈  $A, B, C$  和两个变量  $L, R$ 。

每个栈内元素遵循以下原则：越靠近栈底的元素越大。

初始状态  $L = R = 0$ ，栈  $A$  中有  $n$  个元素（按照从栈底到栈顶，从大到小的顺序，分别是  $n, n - 1, \dots, 3, 2, 1$ ），栈  $B, C$  为空栈。

你需要通过如下两种类型的操作，将所有元素移动到栈  $C$ ，并且保证在操作过程中的任意时刻，所有栈满足上述原则。

操作 1：如果栈  $S$  非空，变量  $V = 0$ ，设  $x$  为当前  $S$  栈顶元素， $S$  弹出栈顶，并且令  $V = x$ 。

操作 2：如果变量  $V = x, (x \neq 0)$ ，将  $x$  作为一个元素压入栈  $S$ ，并且令  $V = 0$ 。

上述的  $V$  是在  $L, R$  中选择一个变量视作  $V$ 。

每次操作可以选择任意类型，选择符合要求的任意栈和变量，你需要最小化使用操作 1 的次数。

给定  $n$ ，输出最小使用操作 1 的次数在模 998244353 意义下的结果。

## 输入格式

---

第一行输入一个整数  $T$  表示测试数据组数。

接下来  $T$  行每行输入一个整数  $n$ 。

$1 \leq T \leq 100$ ,  $1 \leq n \leq 10^6$ 。

## 输出格式

---

输出  $T$  行，每行一个整数，表示答案在模 998244353 意义下的结果。

## 样例输入

---

```
5
1
2
3
10
100
```

## 样例输出

---

```
1
2
4
36
268435446
```

## 提示

---

对于样例中  $n = 3$  的解释：

初始状态： $A(3, 2, 1), B(), C(), L = 0, R = 0$

操作 1 :  $A(3, 2), B(), C(), L = 1, R = 0$

操作 2 :  $A(3, 2), B(1), C(), L = 0, R = 0$

操作 1 :  $A(3), B(1), C(), L = 2, R = 0$

操作 1 :  $A(), B(1), C(), L = 2, R = 3$

操作 2 :  $A(), B(1), C(3), L = 2, R = 0$

操作 2 :  $A(), B(1), C(3, 2), L = 0, R = 0$

操作 1 :  $A(), B(), C(3, 2), L = 1, R = 0$

操作 2 :  $A(), B(), C(3, 2, 1), L = 0, R = 0$