

蒙德的卡牌

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

七圣召唤是风靡提瓦特的卡牌游戏，这个游戏中每个角色牌有三种技能，在每回合的开始，玩家通过投掷获取一定数量的骰子，每个骰子可能是七种元素之一，也可能是无色骰子，玩家可以消耗骰子来发动一些技能。

蒙德的猫尾酒馆经常举行热斗模式的七圣召唤对决，这种模式下一般会有许多离谱的规则，比如减少技能所需骰子，所有骰子均为无色等等。

现在我们考虑一种简单的热斗模式对决，所有骰子均为无色，元素战技和普通攻击仅需1骰子，元素爆发仅需2骰子，且不再需要充能，场上只有一位角色。

为了避免浪费，我们总是希望用完所有骰子，你要求出用完所有骰子的方案数。

具体地说，你一开始有 k 个骰子，每次你可以进行以下三个操作，直到剩余0个骰子为止：

- 普通攻击，消耗1骰子。
- 元素战技，消耗1骰子。
- 元素爆发，消耗2骰子。

两种操作方案被认为是不同的当且仅当它们总的操作次数不同，或者存在一个 i 使得两种操作的第 i 步操作不同。

由于这个数可能特别大，你只需要输出对998244353取模的结果。



猫尾酒馆

Input

一个整数 k ，一开始的骰子数量。（ $1 \leq k \leq 10^{15}$ ）

Output

一个整数，用完所有骰子的方案数，对998244353取模。

Examples

standard input	standard output
1	2
3	12
87842506579379	893188283

Note

一些选手可能不知道如何计算 x^n (n 是自然数)，这里给出一种算法：

注意到：

$$x^0 = 1, x^n = x^{n \bmod 2} \cdot (x^2)^{\lfloor n/2 \rfloor}$$

在编程中实际上将递归展开成循环，有算法：

```
POWER( $x, n$ ) :  
     $r = 1$   
    while  $n > 0$  :  
        if  $n \bmod 2 == 1$  :  
             $r = r \times x$   
         $x = x \times x$   
         $n = \lfloor n/2 \rfloor$   
    return  $r$ 
```

时间复杂度: $O(\lg n)$