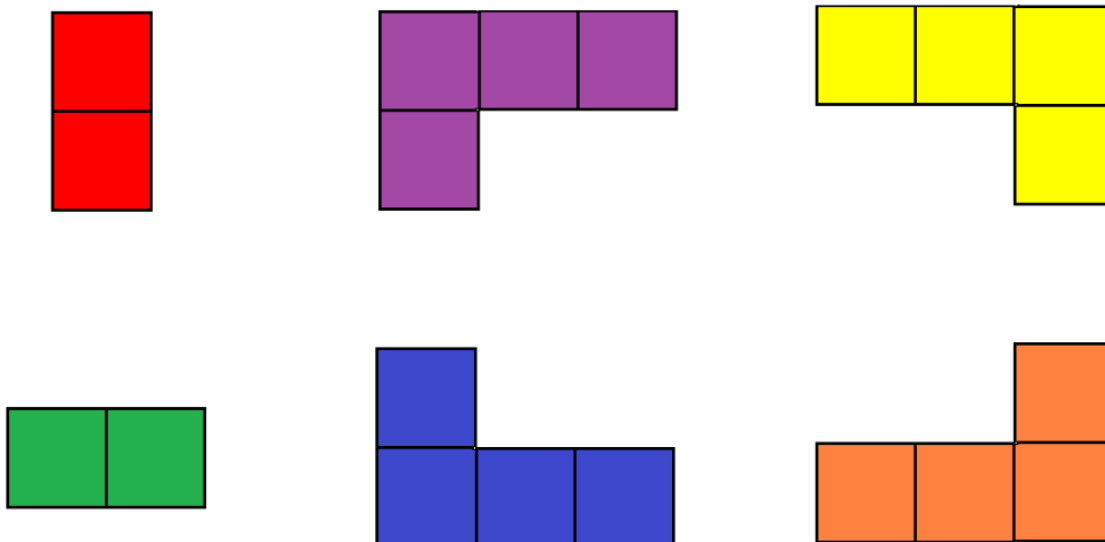


# C2-C problem

## 题目描述

有一个宽度为 2，长度为  $n$  的地板需要贴上瓷砖，有以下 6 种形状的瓷砖可供选择：



瓷砖不能**旋转、翻转、重叠、切割**，且必须恰好把地板**铺满**，可以结合输入输出样例理解。

the\_ignorant 想知道一共有多少种不同的铺瓷砖方案，两种方案不同当且仅当存在一个位置上的瓷砖颜色不同。由于答案可能很大，你只需要输出答案对 998244353 取模后的结果。

## 题解思路

- 根据提示选择递推思路。根据样例得知1,2,3,4对应的结果时1,2,7,15。
- 设  $f(n)$  为长度为  $n$  时的方案数量，当  $n \leq 4$  时，直接输出答案，当  $n > 4$  时利用递推公式计算。
- 递推公式： $f(n) = f(n-1) \times 1 + f(n-2) \times 1 + f(n-3) \times 4 + f(n-4) \times 2$  【 $n > 4$ 】
- 公式解释：铺满长度为  $n-1$  的地板后，再拼接一个红砖，即可铺满长度为  $n$  的地板，由此种方法铺满长度为  $n$  的地板共有  $f(n-1) \times 1$  种铺法；铺满长度为  $n-2$  的地板后，再拼接两个绿色砖块，即可铺满长度为  $n$  的地板，由此种方法铺满长度为  $n$  的地板共有  $f(n-2) \times 1$  种铺法；铺满长度为  $n-3$  的地板后，分别可以由蓝、紫、橙、黄砖块与绿砖块拼成  $2 \times 3$  砖块，拼接在最后，由此种方法铺满长度为  $n$  的地板共有  $f(n-3) \times 4$  种铺法；铺满长度为  $n-4$  的地板后，分别可以由紫橙、蓝黄砖块拼成  $2 \times 4$  砖块，拼接在最后，由此种方法铺满长度为  $n$  的地板共有  $f(n-4) \times 2$  种铺法。
- 在代码实现时不能直接设计成递归函数，递归层数过多会导致栈溢出。使用while循环，直接从5开始计算全部值存储，读入  $n$  后索引即可。

## 代码

```
#include <iostream>
#include<bits/stdc++.h>

using namespace std;

#define MAX 1000005
```

```
#define mod 998244353
long long box[MAX];

int main() {
    memset(box, 0, sizeof(box));
    box[1] = 1;
    box[2] = 2;
    box[3] = 7;
    box[4] = 15;
    for (int i = 5; i <= 1000000; i++) {
        box[i] = box[i - 1] + box[i - 2] + 4 * box[i - 3] + 2 * box[i - 4];
        if (box[i] > mod) {
            box[i] %= mod;
        }
    }
    int t;
    scanf("%d",&t);
    while (t--) {
        int n;
        scanf("%d", &n);
        printf("%d\n", box[n]);
    }
    return 0;
}
```