

# 题解：导弹轰炸

## 问题描述

A 国有  $n$  个前哨站，每个前哨站都有一个重要程度  $w_i$ 。作为 B 国的总指挥，您需要决定如何使用导弹轰炸 A 国的前哨站。由于导弹在轰炸相邻的两个前哨站时会产生干扰，导致导弹失效，因此需要计算在导弹互不干扰的情况下，能够轰炸的前哨站的重要程度之和的最大值。

## 输入

- 第一行包含一个整数  $t$ ，表示测试数据的数量。
- 接下来  $t$  组测试数据，每组数据包含两行：
  - 第一行是一个正整数  $n$  ( $2 \leq n \leq 10^5$ )，表示前哨站的数量。
  - 第二行包含  $n$  个正整数  $w_1, w_2, \dots, w_n$  ( $1 \leq w_i \leq 10^5$ )，表示每个前哨站的重要程度。

数据保证  $\sum n \leq 4 \cdot 10^5$ 。

## 输出

对于每组测试数据，输出能够轰炸的前哨站的重要程度之和的最大值。

## 解题思路

### 动态规划

为了解决这个问题，可以使用动态规划的方法来构建解决方案。定义状态  $dp[i]$  为在前  $i$  个前哨站中可以获得的重要程度之和的最大值。根据题意，状态转移方程如下：

- 如果选择炸  $i$  号前哨站，则必须跳过  $i-1$  号前哨站，状态转移为  $dp[i] = w[i] + dp[i-2]$ 。
- 如果不选择炸  $i$  号前哨站，则状态转移为  $dp[i] = dp[i-1]$ 。
- 综合上述，我们有：

$$[ dp[i] = \max(dp[i-1], w[i] + dp[i-2]) ]$$

### 边界条件

- 初始化  $dp[1] = w[1]$ ，因为只有一个前哨站时，最大值为它的重要程度。

### 复杂度分析

- 时间复杂度：** $O(n)$ ，每组数据只需要一次遍历。
- 空间复杂度：** $O(n)$ ，用于存储  $dp$  数组。

## 代码实现

以下是完整的代码实现：

```
#include <stdio.h>
#define MAX(a,b) (((a)>(b))?(a):(b))

long long w[100100];
long long dp[100100];

int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int n;
        scanf("%d", &n);
        for (int i = 1; i <= n; i++)
            scanf("%lld", &w[i]);
        dp[1] = w[1];
        for (int i = 2; i <= n; i++) {
            dp[i] = MAX(dp[i-1], w[i] + dp[i-2]);
        }
        printf("%lld\n", dp[n]);
    }
    return 0;
}
```