

TOI推廣計畫

解題 - 獎金關卡



Icon made by <https://www.flaticon.com/authors/nhor-phai> from www.flaticon.com

題目

Bob參加益智遊戲的節目錄影，並且闖到了獎金關卡，而獎金關卡的規則是這樣的，現在桌上有著一排按鈕，每個按鈕上面都有相對應的數字，當Bob按下其中一個按鈕，便可得到相當於按鈕數字的積分，但會使其左右兩邊的按鈕失效，當按下失效的按鈕時，無法獲得積分，但仍會使左右兩側的按鈕失效。遊戲結束後，便可獲得和積分相等的獎金，請你寫一個程式，計算Bob最多可以得到多少獎金。

輸入格式

每筆測試資料為二列，

第一列有一個正整數 N ($3 \leq N \leq 10^6$)，總共有幾個按鈕。

第二列有 N 個正整數，數字間以一個空格隔開，按鈕上的數字不超過 99。

輸出格式

對每筆資料請輸出一列共一個數字，請輸出 Bob 可以取得的最高獎金。

| 輸入範例 | 輸出範例 |
|------------------|------|
| 6 1 2 3 4 5 6 | 12 |



解題重點:

1. 動態規劃
2. 搜尋最佳解

Icon made by <https://www.flaticon.com/authors/freepik> from www.flaticon.com

◆ 重點一：動態規劃

狀態轉移方程式為：

$$DP[i] = \max(DP[i-2] + arr[i], DP[i-3] + arr[i])$$

要尋找按鈕 i 的最佳解，先從按鈕 $i-2$ 和 $i-3$ 中取出較大者，再加上按鈕 i 本身的權重即可。

重點提醒

$DP[0]$ 、 $DP[1]$ 、和 $DP[2]$ 無法用上述規則，否則會導致Runtime-error，需事先計算。

◆ 重點二：搜尋最佳解

求得DP的陣列之後，我們只需要從DP的**最後兩項**中取出**較大者**即可，不須遍歷整個陣列。

範例測資1

| arr[i] | 1 | 2 | 3 | 4 | 5 | 6 |
|--------|---|---|---|---|---|----|
| DP[i] | 1 | 2 | 4 | 6 | 9 | 12 |

範例測資2

| arr[i] | 2 | 1 | 2 | 5 | 2 |
|--------|---|---|---|---|---|
| DP[i] | 2 | 1 | 4 | 7 | 6 |



參考解答

```
#include<bits/stdc++.h>
int arr[1000005]={};
int DP[1000005]={};
using namespace std;
int main()
{
    int N;
    scanf("%d",&N);
    for(int i=0;i<N;i+=1)
    {
        scanf("%d",&arr[i]);
        if(i<2) DP[i]=arr[i];
        else if(i==2) DP[i]=arr[0]+arr[2];
        else DP[i]=max(DP[i-2]+arr[i],DP[i-3]+arr[i]);
    }
    printf("%d\n",max(DP[N-1],DP[N-2]));
}
```