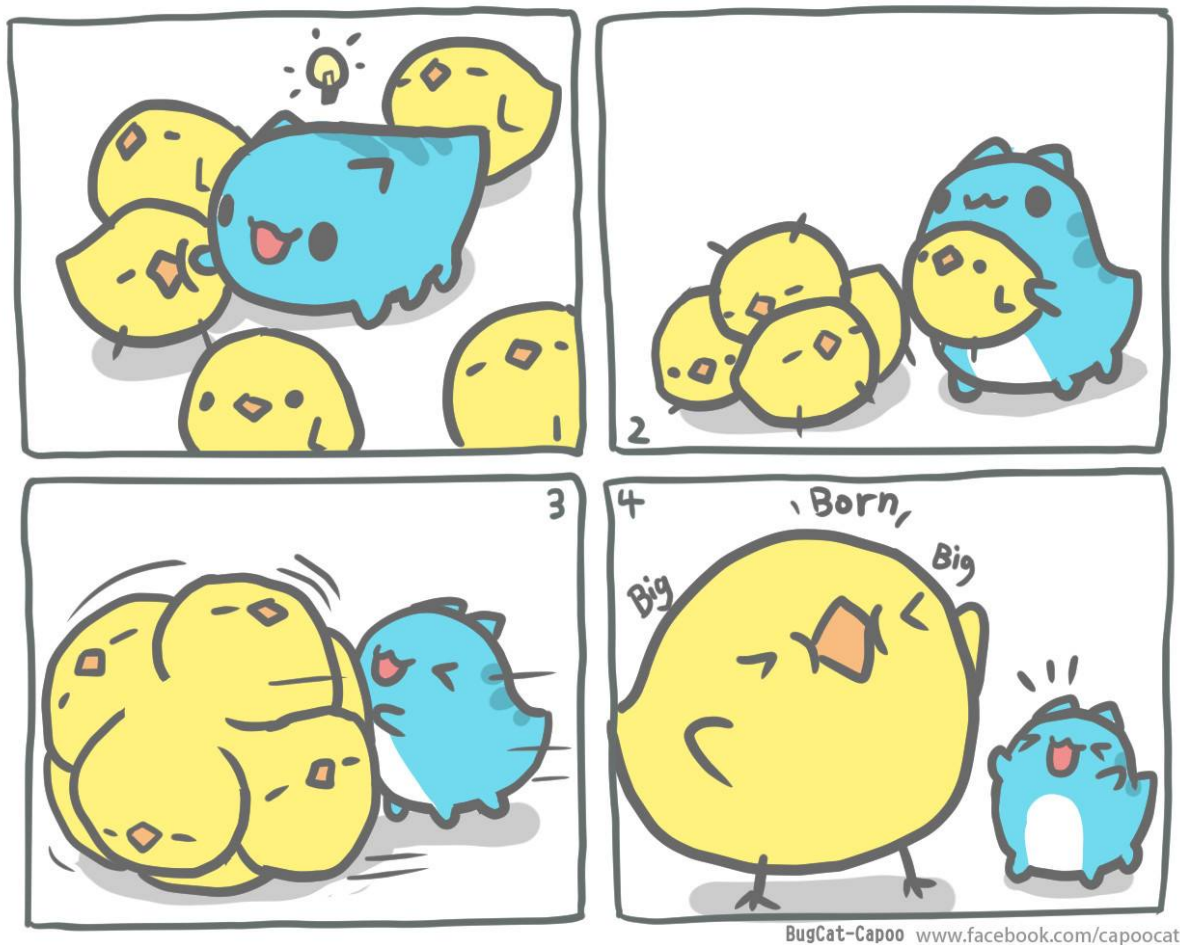


D. 小雞的潛力

potential



咖波覺得計算小雞的飽足感有點累。好險聰明的咖波很快就發現更好的方法。

咖波發現，小雞是可以合成和分割的。於是他這麼做：

咖波先把 N 隻大小不同的小雞排成一直線，飽足感由左至右為 H_1, H_2, \dots, H_n ，咖波可以任選連續的小雞 $H_i, H_{i+1}, H_{i+2}, \dots, H_r$ ，把這些小雞的大小同時增加或減少一個數值 V 。要是小雞變得太小，飽足感也可能會小於零(消化的耗能還比不上得到的能量)。

最後咖波想知道最大的小雞和最小的的小雞，相差多少飽足感。但是咖波只想快點繼續進食。你能幫咖波解決這個問題嗎？

實作說明

你必須上傳一個指定函數，其函數原型如下：

```
int Answer(int N,int P,int *H,int *L,int *R,int *V);
```

參數意義代表如下：

N：小雞的數量

P：修改的次數

H[0],H[1]...H[N-1]：小雞一開始的大小，注意：H[0]是題目敘述中的H₁喔，別搞錯了

L[0],L[1]...L[P-1]：每次修改的左界

R[0],R[1]...R[P-1]：每次修改的右界

V[0],V[1]...V[P-1]：每次修改增加的大小

回傳值意義代表如下：

一個整數，為完成操作後，最大和最小的的小雞大小差多少。

我們可以參考這一個範例：

```
N=5
P=3
H[]={0,0,1,0,1}
L[]={1,2,1}
R[]={3,5,1}
V[]={1,-1,2}
```

代表有5隻小雞，大小分別為{0,0,1,0,1}，接下來有3次修改。

第一次修改的區間為H₁~H₃，增加大小1，修改完後小雞大小為{1,1,2,0,1}

第二次修改的區間為H₂~H₅，增加大小-1，修改完後小雞大小為{1,0,1,-1,0}

第三次修改的區間為H₁~H₁，增加大小2，修改完後小雞大小為{3,0,1,-1,0}

於是答案為最大減最小，即3-(-1)=4。

限制

函式被使用的次數不超過一次

$$1 \leq N \leq 10^6$$

$$0 \leq P \leq 10^6$$

$$-100 \leq H[i] \leq 100$$

$$1 \leq L[i] \leq R[i] \leq N$$

$$-100 \leq V[i] \leq 100$$

子任務

子任務	分數	額外輸入限制
1	18	$N \leq 5, P = 1$
2	22	$N \leq 5$
3	30	$P = 0$
4	30	$P \leq 10^4$
5	40	修改的區間全部不相交
6	60	無

範例程式

為方便您撰寫此題，下提供範例程式檔案，在指定位置撰寫您的程式碼即可。下提供測試資料的輸入方法。

下載: 題目附件中 potential.cpp

範例程式輸入

第一行有兩個數字：N,P，接下來下一行有N個數字，分別代表H[i]的數值，在接下來有P行，每行有三個數字，分別為L[i],R[i],V[i]。

範例程式輸出

直接輸出Answer的回傳值。

範例輸入、輸出

INPUT	OUTPUT
5 3 0 0 1 0 1 1 3 1 2 5 -1 1 1 2	4