

Aggiornamento della macchina virtuale (vm)

Georgio Audritto deve aggiornare la macchina virtuale delle OII. La macchina virtuale contiene N programmi, numerati da 0 a $N - 1$, e l' i -esimo è attualmente alla versione A_i . Gli A_i sono ordinati in ordine crescente (vale $A_i \leq A_{i+1}$ per ogni $0 \leq i \leq N - 2$).

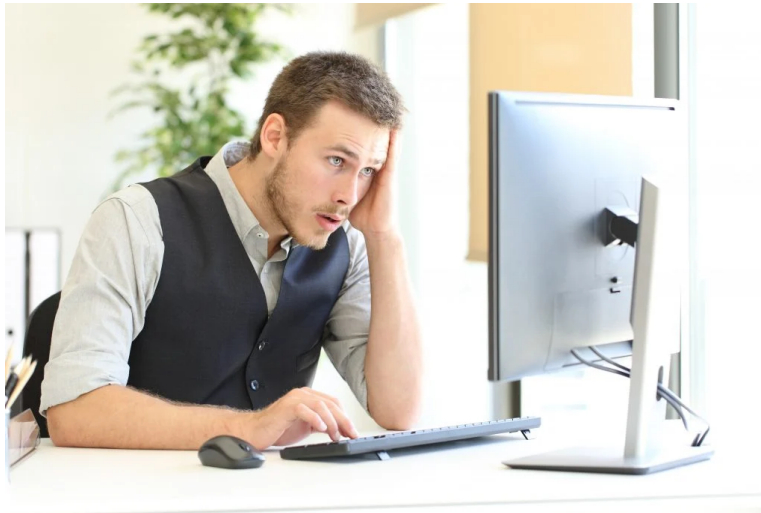


Figura 1: Georgio mentre aggiorna la macchina virtuale

In un aggiornamento, Georgio può scegliere un programma i e aggiornarlo di B_i versioni (quindi A_i aumenta di B_i). Lo stesso programma può essere aggiornato più volte. Inoltre:

- Georgio non può aggiornare il programma $N - 1$.
- L' i -esimo programma dipende da tutti i precedenti, quindi per retrocompatibilità non deve avere versione minore dei programmi con indice minore. In altre parole, dopo ogni aggiornamento, gli A_i devono essere ordinati in ordine crescente (deve valere $A_i \leq A_{i+1}$ per ogni $0 \leq i \leq N - 2$). Nota che l'ordine dei programmi non cambia dopo un aggiornamento.

Qual è il numero massimo di aggiornamenti che Georgio può effettuare?

Implementazione

Dovrai sottoporre un unico file, con estensione `.cpp`.

 Tra gli allegati a questo task troverai un template `vm.cpp` con un esempio di implementazione.

Dovrai implementare la seguente funzione:

```
C++ long long aggiorna(int N, vector<int> A, vector<int> B);
```

La funzione viene chiamata durante l'esecuzione del programma con i seguenti parametri:

- L'intero N rappresenta il numero di programmi nella macchina virtuale.

- L'array A contiene le versioni iniziali dei programmi. In particolare, per ogni $0 \leq i \leq N - 1$, A_i indica la versione dell' i -esimo programma.
- L'array B indica di quante versioni alla volta si possono aggiornare i programmi. In particolare, per ogni $0 \leq i \leq N - 2$, B_i indica quanto aumenta A_i dopo un aggiornamento dell' i -esimo programma.

La funzione `aggiorna` deve restituire il massimo numero di aggiornamenti che Giorgio può effettuare.

Grader di prova

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la correzione, che puoi usare per testare le tue soluzioni in locale. Il grader di esempio legge i dati da `stdin`, chiama la funzione che devi implementare e scrive su `stdout`, secondo il seguente formato.

L'input è composto da 3 righe, contenenti:

- Riga 1: l'intero N .
- Riga 2: N interi A_0, A_1, \dots, A_{N-1} .
- Riga 3: $N - 1$ interi B_0, B_1, \dots, B_{N-2} .

L'output è composto da un'unica riga:

- Riga 1: il valore restituito dalla funzione `aggiorna`.

Assunzioni

- $2 \leq N \leq 2 \cdot 10^5$.
- $1 \leq A_i, B_i \leq 10^9$.
- $A_i \leq A_{i+1}$ per ogni $0 \leq i \leq N - 2$.

Assegnazione del punteggio

Il tuo programma verrà testato su diversi test case raggruppati in subtask. Per ottenere il punteggio relativo ad un subtask, è necessario risolvere correttamente tutti i test relativi ad esso.

- **Subtask 1 [0 punti]:** Casi d'esempio.
- **Subtask 2 [12 punti]:** $B_i = 1$
- **Subtask 3 [20 punti]:** $N \leq 500, A_i \leq 200$
- **Subtask 4 [22 punti]:** $A_i \leq 200$
- **Subtask 5 [19 punti]:** $N = 2 \cdot 10^5$, l'input è scelto uniformemente a caso tra tutti gli input validi.
- **Subtask 6 [27 punti]:** Nessuna limitazione specifica.

Esempi di input/output

stdin	stdout
6 2 7 11 14 18 25 8 4 6 3 4	7
8 3 27 86 95 419 852 1473 6461 100 100 100 1000 1000 1000 10000	0

Spiegazioni

Nel **primo caso d'esempio**, Georgio può effettuare i seguenti aggiornamenti:

- Aggiornamento del programma 4: le versioni diventano [2, 7, 11, 17, 18, 25];
- Aggiornamento del programma 3: le versioni diventano [2, 7, 17, 17, 18, 25];
- Aggiornamento del programma 2: le versioni diventano [2, 11, 17, 17, 18, 25];
- Aggiornamento del programma 1: le versioni diventano [10, 11, 17, 17, 18, 25];
- Aggiornamento del programma 2: le versioni diventano [10, 15, 17, 17, 18, 25];
- Aggiornamento del programma 5: le versioni diventano [10, 15, 17, 17, 22, 25];
- Aggiornamento del programma 4: le versioni diventano [10, 15, 17, 20, 22, 25].

Sono stati effettuati 7 aggiornamenti: è possibile dimostrare che si tratta del numero massimo possibile per questo input.

Nel **secondo caso d'esempio** Georgio non può effettuare aggiornamenti.