

Nile

Dorești să transporti N artefacte prin Nile. Artefactele sunt numerotate de la 0 la $N - 1$. Greutatea artefactului i ($0 \leq i < N$) este $W[i]$.

Pentru a transporta artefactele, utilizezi bărci specializate. Fiecare barcă poate transporta **cel mult două** artefacte.

- În cazul în care decizi să transporti un singur artefact, greutatea artefactului poate fi arbitrară.
- În cazul în care decizi să transporti două artefacte, trebuie să te asiguri că barca este balansată în mod uniform. În particular, poți transporta artefactele p și q ($0 \leq p < q < N$) în aceeași barcă dacă diferența absolută dintre greutatea lor este cel mult D , anume $|W[p] - W[q]| \leq D$.

Pentru a transporta un artefact, trebuie să plătești un cost care depinde de numărul de artefacte transportate în aceeași barcă. Costul de transportare a artefactului i ($0 \leq i < N$) este:

- $A[i]$, dacă transporti acest artefact singur în barcă, sau
- $B[i]$, dacă transporti artefactul împreună cu un alt artefact în aceeași barcă.

De notat că în ultimul caz, vei plăti pentru ambele artefacte din barcă. În particular, dacă decizi să transporti artefactele p și q ($0 \leq p < q < N$) în aceeași barcă, vei plăti $B[p] + B[q]$.

Costul transportării unui singur artefact în barcă este întotdeauna mai mare decât transportarea artefactului împreună cu alt artefact, adică $B[i] < A[i]$ pentru toate i , $0 \leq i < N$.

Din păcate, râul este foarte impredictibil și valoarea lui D se schimbă des. Sarcina ta este să răspunzi la Q întrebări numerotate de la 0 la $Q - 1$. Întrebările sunt descrise de un tablou E de lungime Q . Răspunsul la întrebarea j ($0 \leq j < Q$) este costul total minimal de transportare a tuturor celor N artefacte, unde valoarea lui D este egală cu $E[j]$.

Detalii de implementare

Trebuie să implementezi următoarea funcție:

```
std::vector<long long> calculate_costs(  
    std::vector<int> W, std::vector<int> A,  
    std::vector<int> B, std::vector<int> E)
```

- W, A, B : tablouri de întregi de lungime N , ce conțin greutatea artefactelor și costurile lor de transportare.
- E : un tablou de întregi de lungime Q , ce conține valorile lui D pentru fiecare întrebare.
- Această funcție trebuie să returneze un tablou R de Q întregi ce conține costul total minimal de transportare a artefactelor, unde $R[j]$ conține costul când valoarea lui D este $E[j]$ (pentru fiecare j , $0 \leq j < Q$).
- Această funcție se va apela o singură dată pentru fiecare test.

Restricții

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq W[i] \leq 10^9$ pentru fiecare i , $0 \leq i < N$
- $1 \leq B[i] < A[i] \leq 10^9$ pentru fiecare i , $0 \leq i < N$
- $1 \leq E[j] \leq 10^9$ pentru fiecare j , $0 \leq j < Q$

Subtask-uri

Subtask	Punctaj	Restricții adiționale
1	6	$Q \leq 5$; $N \leq 2000$; $W[i] = 1$ pentru fiecare i , $0 \leq i < N$
2	13	$Q \leq 5$; $W[i] = i + 1$ pentru fiecare i , $0 \leq i < N$
3	17	$Q \leq 5$; $A[i] = 2$ and $B[i] = 1$ pentru fiecare i , $0 \leq i < N$
4	11	$Q \leq 5$; $N \leq 2000$
5	20	$Q \leq 5$
6	15	$A[i] = 2$ și $B[i] = 1$ pentru fiecare i , $0 \leq i < N$
7	18	fără restricții adiționale.

Exemplu

Considerăm următorul apel.

```
calculate_costs([15, 12, 2, 10, 21],
               [5, 4, 5, 6, 3],
               [1, 2, 2, 3, 2],
               [5, 9, 1])
```

În acest exemplu avem $N = 5$ artefacte și $Q = 3$ întrebări.

La prima întrebare, $D = 5$. Poți transporta artefactele 0 și 3 într-o barcă (deoarece $|15 - 10| \leq 5$) și artefactele rămase în bărci separate. Acest lucru duce la costul minimal pentru transportul tuturor

artefactelor, care este $1 + 4 + 5 + 3 + 3 = 16$.

La a doua întrebare, $D = 9$. Poți transporta artefactele 0 și 1 într-o barcă (deoarece $|15 - 12| \leq 9$) și transporta artefactele 2 și 3 înaltă barcă (deoarece $|2 - 10| \leq 9$). Artefactul rămas poate fi transportat în barcă separată. Acest lucru duce la costul minimal pentru transportul tuturor artefactelor, care este $1 + 2 + 2 + 3 + 3 = 11$.

La ultima întrebare, $D = 1$. Ești nevoit să transporti fiecare artefact separat. Acest lucru duce la costul minimal pentru transportul tuturor artefactelor, care este $5 + 4 + 5 + 6 + 3 = 23$.

Prin urmare, funcția trebuie să returneze $[16, 11, 23]$.

Grader-ul local

Format intrare:

```
N
W[0] A[0] B[0]
W[1] A[1] B[1]
...
W[N-1] A[N-1] B[N-1]
Q
E[0]
E[1]
...
E[Q-1]
```

Format ieșire:

```
R[0]
R[1]
...
R[S-1]
```

Aici, S este lungimea tabloului R returnat de `calculate_costs`.