

# Nilen

Du vill transportera artefakter  $N$  genom Nilen. Artefakterna är numrerade från 0 till  $N - 1$ . Vikten av artefakt  $i$  ( $0 \leq i < N$ ) är  $W[i]$ .

För att transportera artefakterna använder du specialiserade båtar. Varje båt kan bära **upp till två** artefakter.

- Om du lägger en enda artefakt i en båt kan artefaktens vikt vara vad som helst.
- Om du vill lägga två artefakter i samma båt måste du se till att båten är jämnt balanserad. Mer exakt kan du skicka två artefakter  $p$  och  $q$  ( $0 \leq p < q < N$ ) i samma båt om den absoluta skillnaden mellan deras vikter är högst  $D$ , det vill säga  $|W[p] - W[q]| \leq D$ .

För att transportera en artefakt måste du betala en kostnad som beror på antalet artefakter som transporteras i samma båt. Kostnaden för att transportera artefakt  $i$  ( $0 \leq i < N$ ) är:

- $A[i]$ , om du lägger artefakten i sin egen båt, eller
- $B[i]$ , om du lägger den i en båt tillsammans med någon annan artefakt.

Observera att i det senare fallet måste du betala för båda artefakterna i båten. Det vill säga, om du bestämmer dig för att skicka artefakter  $p$  och  $q$  ( $0 \leq p < q < N$ ) i samma båt, så måste du måste betala  $B[p] + B[q]$ .

Att skicka en artefakt ensam i en båt är alltid dyrare än att skicka den med någon annan artefakt som då får dela båten med den, så  $B[i] < A[i]$  gäller för alla  $0 \leq i < N$ .

Tyvärr är floden väldigt oförutsägbar och värdet på  $D$  ändras ofta. Din uppgift är att svara på  $Q$  stycken frågor numrerade från 0 till  $Q - 1$ . Frågorna beskrivs av en array  $E$  med längd  $Q$ . Svaret på frågan  $j$  ( $0 \leq j < Q$ ) är den lägsta totala kostnaden för att transportera alla artefakter  $N$ , när värdet på  $D$  är lika med  $E[j]$ .

## Implementeringsdetaljer

Snälla implementera följande funktion 🙄

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

- $W, A, B$ : arrayer av heltal med längd  $N$ , som beskriver vikten av artefakterna och kostnaderna för att transportera dem.
- $E$ : en array med heltal med längden  $Q$  som beskriver värdet av  $D$  för varje fråga.
- Denna funktion bör returnera en array  $R$  med  $Q$  heltal som innehåller den lägsta totala kostnaden för att transportera artefakterna, där  $R[j]$  anger kostnaden när värdet på  $D$  är  $E[j]$  (för varje  $j$  så att  $0 \leq j < Q$ ).
- Denna funktion anropas exakt en gång för varje testfall.

## Begränsningar

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq W[i] \leq 10^9$  för alla  $i$  sådan att  $0 \leq i < N$
- $1 \leq B[i] < A[i] \leq 10^9$  för alla  $i$  sådan att  $0 \leq i < N$
- $1 \leq E[j] \leq 10^9$  för alla  $j$  sådan att  $0 \leq j < Q$

## Delpoäng

Grupp	Poäng	Ytterligare begränsningar
1	6	$Q \leq 5; N \leq 2000; W[i] = 1$ för alla $i$ sådan att $0 \leq i < N$
2	13	$Q \leq 5; W[i] = i + 1$ för alla $i$ sådan att $0 \leq i < N$
3	17	$Q \leq 5; A[i] = 2$ and $B[i] = 1$ för alla $i$ sådan att $0 \leq i < N$
4	11	$Q \leq 5; N \leq 2000$
5	20	$Q \leq 5$
6	15	$A[i] = 2$ and $B[i] = 1$ för alla $i$ sådan att $0 \leq i < N$
7	18	Inga ytterligare begränsningar.

## Exempel

Pondera på följande funktionsanrop.

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

I det här exemplet har vi  $N = 5$  artefakter och  $Q = 3$  frågor.

I den första frågan gäller  $D = 5$ . Du kan skicka artefakter 0 och 3 i en båt (eftersom  $|15 - 10| \leq 5$ ) och de återstående artefakterna i separata båtar. Detta ger den lägsta kostnaden för att

transportera alla artefakter, vilket är  $1 + 4 + 5 + 3 + 3 = 16$ .

I den andra frågan är  $D = 9$ . Du kan skicka artefakter 0 och 1 i en båt (eftersom  $|15 - 12| \leq 9$ ) och skicka artefakter 2 och 3 i en båt (eftersom  $|2 - 10| \leq 9$ ). Återstående artefakt kan skickas i en separat båt. Detta ger den lägsta kostnaden för att transportera alla artefakter, vilket är  $1 + 2 + 2 + 3 + 3 = 11$ .

I den sista frågan är  $D = 1$ . Du måste då skicka varje artefakt i sin egen båt. Detta ger den lägsta kostnaden för att transportera alla artefakter, vilket är  $5 + 4 + 5 + 6 + 3 = 23$ .

Därför bör denna procedur returnera  $[16, 11, 23]$ .

## Exempelgrader

Inputformat:

```
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]
```

Outputformat:

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

Här är  $S$  längden på arrayen  $R$  som returneras av funktionen `calculate_costs`.