

Nilas

Nilo upe norite nuplukdyti N artefaktų. Artefaktai sunumeruoti nuo 0 iki $N - 1$. i -ojo ($0 \leq i < N$) artefakto svoris yra $W[i]$.

Artefaktai plukdomi specialiais laivais. Kiekviename laive telpa **daugiausiai du** artefaktai.

- Jei nusprendžiate laivu plukdyti vieną artefaktą, jo svoris gali būti bet koks.
- Jei norite laivu plukdyti du artefaktus, tai turite užtikrinti, kad laivas tinkamai subalansuotas. Konkrečiau, artefaktus p ir q ($0 \leq p < q < N$) galite plukdyti tuo pačiu laivu tik tada, kai artefaktų svorių skirtumų absoliuti reikšmė neviršija D , t. y. $|W[p] - W[q]| \leq D$.

Artefakto plukdymas kainuoja, o kaina priklauso nuo to, keli artefaktai plukdomi tuo pačiu laivu. Artefakto i ($0 \leq i < N$) plukdymo kaina lygi:

- $A[i]$, jei artefaktas plukdomas atskiru laivu, arba
- $B[i]$, jei artefaktas plukdomas laive kartu su kitu artefaktu.

Atkreipiame dėmesį, kad antruoju atveju reikia mokėti už abu artefaktus tame pačiame laive. Konkrečiau, jei nusprendžiate tame pačiame laive plukdyti artefaktus p ir q ($0 \leq p < q < N$), reikės sumokėti $B[p] + B[q]$.

Visuomet brangiau kainuos plukdyti artefaktą vieną laive, nei kartu su kitu artefaktu. Taigi, $B[i] < A[i]$ visiems i , kur $0 \leq i < N$.

Deja, upė neprognozuojama ir D vertė dažnai keičiasi. Jums reikia atsakyti į Q užklausų, sunumeruotų nuo 0 iki $Q - 1$. Užklausa nusako Q ilgio masyvas E . Atsakymas į j -ąją užklausą ($0 \leq j < Q$) lygus mažiausiai visų N artefaktų transportavimo kainai, kai D vertė lygi $E[j]$.

Realizacija

Parašykite šią procedūrą:

```
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 : sveikųjų skaičių N ilgio masyvai, nusakantys artefaktų svorius ir jų plukdymo kainas.

- E : sveikųjų skaičių Q ilgio masyvas, nusakantis D vertes kiekvienai užklausiai.
- Ši procedūra turi grąžinti masyvą R , kuriame yra Q sveikųjų skaičių, reiškiančių mažiausią bendrą artefaktų plukdymo kainą. Čia $R[j]$ reiškia kainą, kai D vertė lygi $E[j]$ (visiems j , kur $0 \leq j < Q$).
- Ši procedūra iškviečiama lygiai vieną kartą kiekvienam testui.

Ribojimai

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

Dalinės užduotys

| Dalinė užduotis | Taškai | Ribojimai |
|-----------------|--------|---|
| 1 | 6 | $Q \leq 5$; $N \leq 2000$; $W[i] = 1$ kiekvienam i , kur $0 \leq i < N$ |
| 2 | 13 | $Q \leq 5$; $W[i] = i + 1$ kiekvienam i , kur $0 \leq i < N$ |
| 3 | 17 | $Q \leq 5$; $A[i] = 2$ ir $B[i] = 1$ kiekvienam i , kur $0 \leq i < N$ |
| 4 | 11 | $Q \leq 5$; $N \leq 2000$ |
| 5 | 20 | $Q \leq 5$ |
| 6 | 15 | $A[i] = 2$ ir $B[i] = 1$ kiekvienam i , kur $0 \leq i < N$ |
| 7 | 18 | Papildomų ribojimų nėra. |

Pavyzdys

Panagrinėkime tokį iškvietimą.

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

Šiame pavyzdyje turime $N = 5$ artefaktus ir $Q = 3$ užklausas.

Pirmojoje užklausoje $D = 5$.

Viename laive galite plukdyti artefaktus 0 ir 3 (nes $|15 - 10| \leq 5$), o likusius artefaktus -- atskiruose laivuose. Minimali visų artefaktų perplukdymo kaina lygi $1 + 4 + 5 + 3 + 3 = 16$.

Antrojoje užklausoje $D = 9$. Viename laive galite plukdyti artefaktus 0 ir 1 (nes $|15 - 12| \leq 9$), kitame laive artefaktus 2 ir 3 (nes $|2 - 10| \leq 9$). Likęs artefaktas plukdomas atskiru laivu. Gauname minimalią visų artefaktų plukdymo kainą, kuri lygi $1 + 2 + 2 + 3 + 3 = 11$.

Paskutiniojoje užklausoje $D = 1$. Kiekvieną artefaktą reikės plukdyti atskirame laive. Tokiu atveju minimali visų artefaktų plukdymo kaina yra $5 + 4 + 5 + 6 + 3 = 23$.

Taigi, ši procedūra turi grąžinti $[16, 11, 23]$.

Pavyzdinė vertinimo programa

Pradinių duomenų formatas:

```
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]
```

Rezultatų formatas:

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

Čia S yra masyvo R , kurį grąžino `calculate_costs`, ilgis.