

Nile

Quieres transportar N artefactos a través del Nilo. Los artefactos están enumerados desde 0 hasta $N - 1$. El peso del artefacto i ($0 \leq i < N$) es $W[i]$.

Para transportar los artefactos, usas botes especiales. Cada bote puede llevar **a lo sumo dos** artefactos.

- Si decides colocar un sólo artefacto en un bote, el peso del artefacto puede ser arbitrario.
- Si quieres colocar dos artefactos en un mismo bote, debes asegurar que el bote esté balanceado uniformemente.

Específicamente puedes enviar dos artefactos p y q ($0 \leq p < q < N$) en el mismo bote si la diferencia absoluta entre sus pesos es a lo sumo D , esto es $|W[p] - W[q]| \leq D$.

Para transportar un artefacto tienes que pagar un costo que depende del número de artefactos llevados en un mismo bote. El costo de transportar el artefacto i ($0 \leq i < N$) es:

- $A[i]$, si lo colocas en su propio bote, o
- $B[i]$, si lo colocas en un bote junto con algún otro artefacto.

Nota que en el segundo escenario tienes que pagar por ambos artefactos en el bote. Específicamente si decides enviar los artefactos p y q ($0 \leq p < q < N$) en el mismo bote, deberás pagar $B[p] + B[q]$.

Enviar un solo artefacto en un bote siempre es más costoso que enviarlo junto con otro artefacto compartiendo el mismo bote, así que $B[i] < A[i]$ para todo i tal que $0 \leq i < N$.

Desafortunadamente el río es muy impredecible y el valor de D cambia a menudo. Tu tarea es responder a Q preguntas enumeradas de 0 a $Q - 1$. Las preguntas están descritas por un arreglo E de longitud Q . La respuesta a la pregunta j ($0 \leq j < Q$) es el mínimo costo total de transportar todos los N artefactos cuando el valor de D es igual a $E[j]$.

Detalles de implementación

Deberás implementar la siguiente función.

```
std::vector calculate_costs(
    std::vector W, std::vector A,
    std::vector B, std::vector E)
```

- W, A, B : arreglos de enteros de longitud N que describen los pesos de los artefactos y sus costos para transportarlos.
- E : un arreglo de enteros de longitud Q que describe los diferentes valores de D para cada pregunta.
- Esta función debe retornar un arreglo R de Q enteros conteniendo el mínimo costo total de transportar los artefactos, donde $R[j]$ da el costo cuando el valor de D es $E[j]$ (para todo j tal que $0 \leq j < Q$).
- Esta función es llamada exactamente una vez por cada caso.

Restricciones

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq W[i] \leq 10^9$ para todo i tal que $0 \leq i < N$
- $1 \leq B[i] < A[i] \leq 10^9$ para todo i tal que $0 \leq i < N$
- $1 \leq E[j] \leq 10^9$ para todo j tal que $0 \leq j < Q$

Subtareas

Subtarea	Puntaje	Restricciones adicionales
1	6	$Q \leq 5; N \leq 2000; W[i] = 1$ para todo i tal que $0 \leq i < N$
2	13	$Q \leq 5; W[i] = i + 1$ para todo i tal que $0 \leq i < N$
3	17	$Q \leq 5; A[i] = 2$ y $B[i] = 1$ para todo i tal que $0 \leq i < N$
4	11	$Q \leq 5; N \leq 2000$
5	20	$Q \leq 5$
6	15	$A[i] = 2$ y $B[i] = 1$ para todo i tal que $0 \leq i < N$
7	18	Sin restricciones adicionales.

Ejemplo

Considera la siguiente llamada

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

En este ejemplo tenemos $N = 5$ artefactos y $Q = 3$ preguntas.

En la primera pregunta, $D = 5$. Puedes enviar los artefactos 0 y 3 en un mismo bote (puesto que $|15 - 10| \leq 5$), y en botes separados al resto de los artefactos. Esto resulta en el mínimo costo para transportar todos los artefactos, el cual es $1 + 4 + 5 + 3 + 3 = 16$.

En la segunda pregunta, $D = 9$. Puedes enviar los artefactos 0 y 1 en un mismo bote (puesto que $|15 - 12| \leq 9$) y enviar los artefactos 2 y 3 en un mismo bote (puesto que $|2 - 10| \leq 9$). El artefacto restante puede ser enviado en un bote por separado. Esto resulta en el mínimo costo para transportar todos los artefactos, el cual es $1 + 2 + 2 + 3 + 3 = 11$.

En la última pregunta, $D = 1$. Tienes que enviar cada artefacto en su propio bote. Esto resulta en el mínimo costo para transportar todos los artefactos, el cual es $5 + 4 + 5 + 6 + 3 = 23$.

Por lo tanto, la función debe retornar $[16, 11, 23]$.

Evaluador de prueba

Formato de entrada:

```
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]
```

Formato de salida:

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

Aquí, S es la longitud del arreglo R retornado por `calculate_costs`.