

Tiempo límite: 1 segundo.

Memoria límite: 256MB.

Igualando Arreglos

Dados dos arreglos A y B , de n elementos cada uno, puede realizar cualquiera de las siguientes operaciones cualquier cantidad de veces:

- Seleccione un índice i , tal que $2 \leq i \leq n - 1$, súmele 1 a A_{i-1} , réstele 1 a A_i y súmele 1 a A_{i+1} .
- Seleccione un índice i , tal que $2 \leq i \leq n - 1$, réstele 1 a A_{i-1} , súmele 1 a A_i y réstele 1 a A_{i+1} .

Además tendrá que procesar q updates de la forma (l, r, x) , que consisten en sumarle x a A_l, A_{l+1}, \dots, A_r .

Después de cada update usted tendrá que decir si se puede convertir A en B usando las operaciones dadas, note que las operaciones que usted haga no se guardarán para próximos updates, pero los elementos a los que se les sumó por el update si se guardan para próximos updates.

Restricciones:

$$3 \leq n \leq 2 \cdot 10^5$$

$$0 \leq q \leq 2 \cdot 10^5$$

$$1 \leq A_i, B_i \leq 10^6 \text{ para cada } i \text{ tal que } 1 \leq i \leq n$$

$$1 \leq l_i \leq r_i \leq n \text{ y } -10^6 \leq x_i \leq 10^6 \text{ para cada } i \text{ tal que } 1 \leq i \leq q$$

Subtareas:

- Subtarea 1: $1 \leq n \leq 100$, $1 \leq A_i, B_i \leq 100$ para cada i tal que $1 \leq i \leq n$, $q = 0$ (no hay updates). (10 puntos)
- Subtarea 2: $q = 0$ (no hay updates). (20 puntos)
- Subtarea 3: para todos los updates, se cumple que $l_i = r_i$. (40 puntos)
- Subtarea 4: Sin restricciones adicionales. (30 puntos)

Formato de entrada:

Primero una línea con n y q , dos enteros separados por un espacio, el tamaño de los arreglos A y B , y la cantidad de updates respectivamente.

Luego una línea con n enteros A_1, A_2, \dots, A_n separados por un espacio.

Luego una línea con n enteros B_1, B_2, \dots, B_n separados por un espacio.

Luego q líneas, la i -ésima de ellas describiendo el i -ésimo update, l_i, r_i, x_i , separados por un espacio.

Formato de salida:

Debe imprimir $q + 1$ líneas, cada una debe contener la respuesta después de cada update, la primera línea contiene la respuesta antes de que se haga el primer update. Debe imprimir *YES* si se puede alcanzar el arreglo B a partir de A usando las operaciones requeridas o *NO* de lo contrario.

Ejemplo de entrada 1:

```
5 0
1 2 3 2 2
2 3 5 1 5
```

Ejemplo de salida 1:

YES

Ejemplo de entrada 2:

```
12 12
17 23 2 35 32 8 24 4 28 18 4 28
20 19 8 31 34 5 26 3 23 20 2 28
2 6 8
4 4 -10
9 9 -1
11 12 7
10 10 -7
3 9 10
9 9 4
3 9 8
3 3 -8
3 5 4
10 10 -8
5 11 6
```

Ejemplo de salida 2:

NO
NO
NO
YES
NO
NO
NO
YES
NO
YES
NO
YES
NO