Caminado por puentes

Kevan dibujó un plano de edificios y puentes a lo largo de una acera de la avenida principal de Bakú. Hay n edificios numerados de 0 a $n-1$ y m puentes numerados de 0 a $m-1$. El plano está dibujado en un plano cartesiano bidimensional, en el cual los edificios y puentes son segmentos verticales y horizontales respectivamente.

La base del edificio i $(0 \leq i \leq n-1)$ está ubicada en el punto $(x[i], 0)$ y el edificio tiene altura $h[i]$. Por lo tanto, es un segmento conectando los puntos $(x[i], 0)$ y $(x[i], h[i])$.

El puente j $(0 \leq j \leq m-1)$ tiene sus extremos en los edificios numerados $l[j]$ y $r[j]$ y tiene la coordenada y positiva de valor $y[j]$. Por lo tanto, es un segmento que conecta los puntos $(x[l[j]], y[j])$ y $(x[r[j]], y[j])$.

Un puente y un edificio se **intersecan** si comparten un punto en común. Por lo tanto, un puente interseca dos edificios en sus dos extremos, y también puede intersecar otros edificios que estén entre ellos.

Kenan quisiera encontrar la longitud del camino más corto desde la base del edificio s a la base del edificio g, asumiendo que solamente se puede desplazar por los edificios y puentes, o determine que no existe tal camino. Note que no se permite caminar por la calle; es decir, no se permite caminar a lo largo de la recta horizontal $y = 0$.

Se puede ir de un puente a un edificio o viceversa en cualquier intersección. Si en un mismo punto hay extremos de dos puentes, se puede ir de un puente al otro.

Su tarea es ayudar a Kenan a responder esta pregunta.

Detalles de implementación

Debes implementar el procedimiento siguiente. Este será llamado por el grader una vez por cada caso de prueba.

```
int64 min_distance(int[] x, int[] h, int[] l, int[] r, int[] y,
                   int s, int g)
```

- x y h: arrays de enteros de longitud n
- l, r, y y: arrays de enteros de longitud m
- s y g: dos enteros
- Este procedimiento debe devolver la longitud del camino más corto entre la base

del edificio s y la base del edificio g, si tal camino existe. En otro caso, debe devolver -1.

Ejemplos

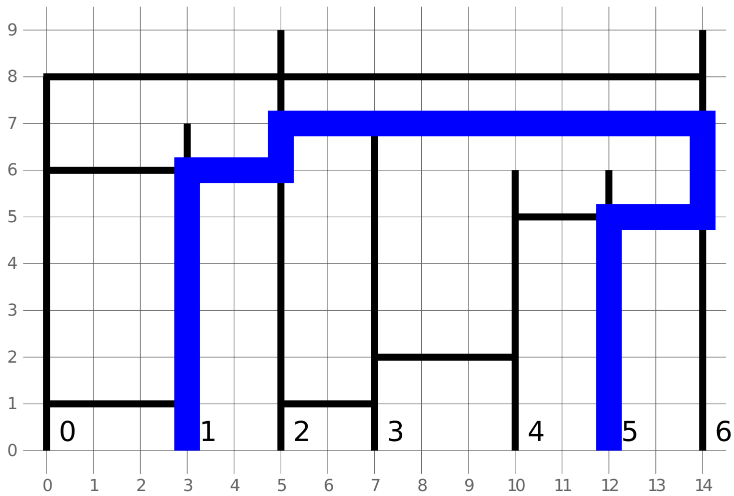
Ejemplo 1

Considere la siguiente llamada:

```
min_distance([0, 3, 5, 7, 10, 12, 14],
             [8, 7, 9, 7, 6, 6, 9],
             [0, 0, 0, 2, 2, 3, 4],
             [1, 2, 6, 3, 6, 4, 6],
             [1, 6, 8, 1, 7, 2, 5],
             1, 5)
```

La respuesta correcta es 27.

La figura mostrada a continuación corresponde al *Ejemplo 1*:



Ejemplo 2

```
min_distance([0, 4, 5, 6, 9],
             [6, 6, 6, 6, 6],
             [3, 1, 0],
             [4, 3, 2],
             [1, 3, 6],
             0, 4)
```

La respuesta correcta es 21.

Restricciones

- $1 \leq n, m \leq 100\,000$
- $0 \leq x[0] < x[1] < \ldots < x[n-1] \leq 10^9$
- $1 \leq h[i] \leq 10^9$ (para todo $0 \leq i \leq n-1$)
- $0 \leq l[j] < r[j] \leq n-1$ (para todo $0 \leq j \leq m-1$)
- $1 \leq y[j] \leq \min(h[l[j]], h[r[j]])$ (para todo $0 \leq j \leq m-1$)
- $0 \leq s, g \leq n-1$
- $s \neq g$
- No hay dos caminos que tengan un punto en común, excepto tal vez en sus extremos.

Subtareas

1. (10 puntos) $n, m \leq 50$
2. (14 puntos) Cada puente interseca como máximo con 10 edificios.
3. (15 puntos) $s = 0$, $g = n-1$, y todos los edificios son de la misma altura.
4. (18 puntos) $s = 0$, $g = n-1$
5. (43 puntos) No hay restricciones adicionales.

Sample grader

El sample grader lee la entrada en el formato siguiente:

- línea 1: $n \ \ m$
- línea $2+i$ ($0 \leq i \leq n-1$): $x[i] \ \ h[i]$
- línea $n+2+j$ ($0 \leq j \leq m-1$): $l[j] \ \ r[j] \ \ y[j]$
- línea $n+m+2$: $s \ \ g$

El sample grader imprime una sola línea conteniendo el valor devuelto por `min_distance`.