



## Sky Walking

Kenan desenează un plan al clădirilor și pasarelelor de-a lungul bulevardului principal din Baku. Există  $n$  clădiri numerotate de la 0 la  $n - 1$  și  $m$  pasarele numerotate de la 0 la  $m - 1$ . Planul este desenat pe o suprafață bidimensională, unde clădirile și pasarelele sunt segmente verticale respectiv orizontale.

Partea de jos a clădirii  $i$  ( $0 \leq i \leq n - 1$ ) se află la punctul  $(x[i], 0)$ , iar clădirea are înălțimea  $h[i]$ . Prin urmare, este un segment care conectează punctele  $(x[i], 0)$  și  $(x[i], h[i])$ .

Pasarela  $j$  ( $0 \leq j \leq m - 1$ ) are puncte terminale la clădirile numerotate cu  $l[j]$  și  $r[j]$ , aflată la o coordonată  $y$  pozitivă cu valoarea  $y[j]$ . Prin urmare, este un segment care unește punctele  $(x[l[j]], y[j])$  și  $(x[r[j]], y[j])$ .

O pasarelă și o clădire **se intersectează** dacă au un punct comun. Prin urmare, o pasarelă intersectează două clădiri în punctele ei terminale, și deasemenea poate intersecta alte clădiri între acestea.

Kenan ar dori să găsească lungimea celui mai scurt drum de la baza clădirii  $s$  la baza clădirii  $g$ , presupunând că se poate merge doar prin clădiri și pasarele, sau să se determine dacă o astfel de cale nu există. Rețineți că nu este permis să mergeți pe jos, adică de-a lungul liniei orizontale cu coordonata  $y$  egală cu 0.

În orice intersecție se poate merge de pe pasarelă în clădire sau invers. Dacă punctele terminale pentru două pasarele sunt identice, atunci se poate merge de pe o pasarelă pe alta.

Sarcina voastră este să îl ajutați pe Kenan să răspundă la întrebarea lui.

## Detalii de implementare

Trebuie să implementați următoarea funcție. Aceasta va fi apelată de către grader câte o dată pentru fiecare test.

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

- $x$  și  $h$ : vectori cu numere întregi de lungime  $n$
- $l$ ,  $r$  și  $y$ : vectori cu numere întregi de lungime  $m$

- $s$  și  $g$ : două numere întregi
- Această funcție trebuie să returneze lungimea celui mai scurt drum dintre baza clădirii  $s$  și baza clădirii  $g$ , dacă o astfel de cale există. În caz contrar, trebuie să se returneze  $-1$ .

## Exemple

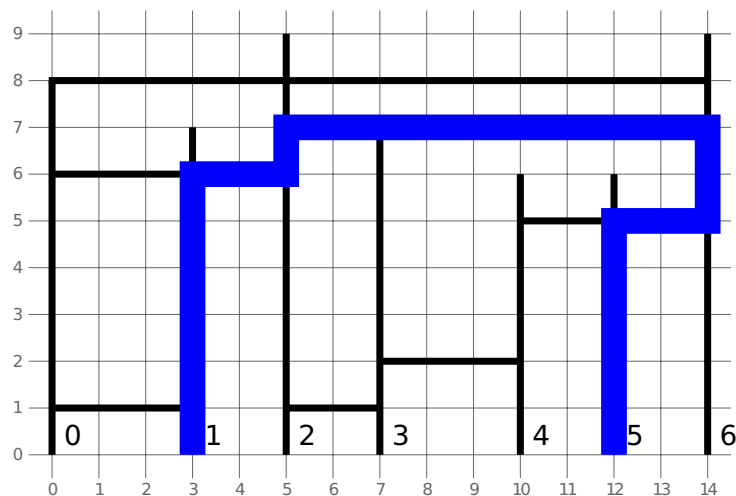
### Exemplul 1

Considerați următorul apel:

```
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)
```

Răspunsul corect este 27.

Imaginea de mai jos corespunde *Exemplului 1*:



### Exemplul 2

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

Răspunsul corect este 21.

## Restricții

- $1 \leq n, m \leq 100\,000$
- $0 \leq x[0] < x[1] < \dots < x[n-1] \leq 10^9$
- $1 \leq h[i] \leq 10^9$  (pentru  $0 \leq i \leq n-1$ )
- $0 \leq l[i] < r[i] \leq n-1$  (pentru  $0 \leq i \leq m-1$ )
- $1 \leq y[i] \leq \min(h[l[i]], h[r[i]])$  (pentru  $0 \leq i \leq m-1$ )
- $0 \leq s, g \leq n-1$
- $s \neq g$
- Două pasarele nu au puncte comune, eventuale excepții pot fi punctele lor terminale.

## Subtask-uri

1. (10 puncte)  $n, m \leq 50$
2. (14 puncte) Fiecare pasarelă intersectează cel mult 10 clădiri.
3. (15 puncte)  $s = 0, g = n-1$ , și toate clădirile au aceeași înălțime
4. (18 puncte)  $s = 0, g = n-1$
5. (43 de puncte) Fără restricții suplimentare.

## Exemplu de grader

Grader-ul citește intrarea în următorul format:

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

Grader-ul va tipări o singură linie conținând valoarea returnată de `min_distance`.