



## Jembatan Layang

Kenan menggambar sebuah rencana akan gedung-gedung dan jembatan-jembatan layang pada satu sisi jalan raya Baku. Terdapat  $n$  gedung yang dinomori dari 0 hingga  $n - 1$  dan  $m$  jembatan layang yang dinomori dari 0 hingga  $m - 1$ . Rencana tersebut digambarkan di sebuah bidang dua dimensi, dimana gedung-gedung dan jembatan-jembatan layang adalah segmen vertikal dan horisontal secara berturutan.

Dasar dari gedung  $i$  ( $0 \leq i \leq n - 1$ ) terletak pada titik  $(x[i], 0)$  dan gedung tersebut memiliki tinggi  $h[i]$ . Sehingga, gedung adalah segmen yang menghubungkan titik  $(x[i], 0)$  dan  $(x[i], h[i])$ .

Jembatan layang  $j$  ( $0 \leq j \leq m - 1$ ) memiliki ujung-ujung pada gedung bernomor  $l[j]$  dan  $r[j]$  dan memiliki koordinat- $y$  yang bernilai positif  $y[j]$ . Sehingga, jembatan layang adalah segmen yang menghubungkan titik  $(x[l[j]], y[j])$  dan  $(x[r[j]], y[j])$ .

Suatu jembatan layang dan suatu gedung **berpotongan** jika mereka berbagi suatu titik yang sama. Sehingga, suatu jembatan layang berpotongan dengan dua gedung pada kedua ujungnya, dan mungkin berpotongan dengan gedung-gedung lain di antaranya.

Kenan ingin mengetahui panjang dari jalan terpendek (shortest path) dari dasar gedung  $s$  menuju dasar gedung  $g$ , dengan asumsi bahwa seseorang hanya dapat berjalan pada gedung dan jembatan layang, atau menentukan bahwa tidak ada jalan tersebut. Perhatikan bahwa tidak diperbolehkan untuk berjalan di dasar, maksudnya pada garis horisontal dengan koordinat- $y$  0.

Seseorang dapat berjalan dari jembatan layang ke gedung atau sebaliknya pada titik perpotongan. Jika ujung dari dua jembatan layang terletak pada titik yang sama, seseorang dapat berjalan dari satu jembatan layang ke jembatan layang yang lainnya.

Tugas Anda adalah untuk membantu Kenan menjawab pertanyaannya.

### Rincian implementasi

Anda harus melakukan implementasi dari prosedur berikut. Prosedur akan sekali dipanggil oleh grader untuk setiap kasus uji.

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

- $x$  dan  $h$ : array bilangan bulat dengan panjang  $n$
- $l$ ,  $r$ , dan  $y$ : array bilangan bulat dengan panjang  $m$
- $s$  dan  $g$ : dua buah bilangan bulat
- Prosedur ini harus mengembalikan panjang dari jalan terpendek (shortest path) antara dasar gedung  $s$  dan dasar gedung  $g$ , jika terdapat jalan tersebut. Jika tidak, prosedur ini harus mengembalikan  $-1$ .

## Contoh

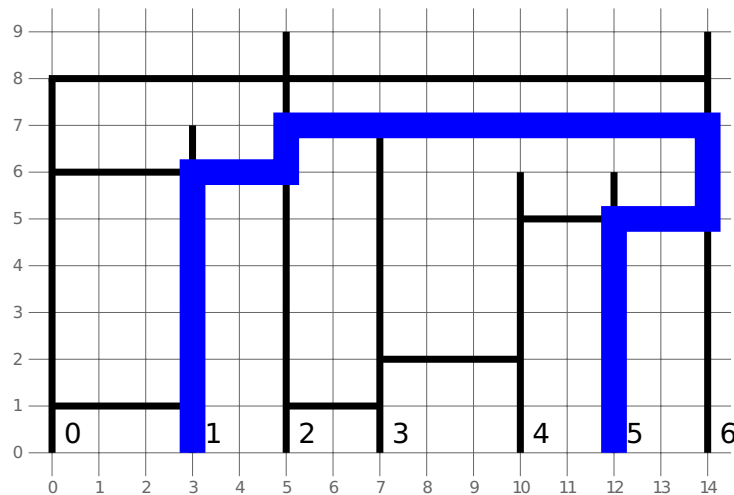
### Contoh 1

Perhatikan panggilan berikut:

```
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)
```

Jawaban yang benar adalah 27.

Gambar di bawah ini mengacu pada *Contoh 1*:



### Contoh 2

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

Jawaban yang benar adalah 21.

## Batasan

- $1 \leq n, m \leq 100\,000$
- $0 \leq x[0] < x[1] < \dots < x[n-1] \leq 10^9$
- $1 \leq h[i] \leq 10^9$  (untuk semua  $0 \leq i \leq n-1$ )
- $0 \leq l[j] < r[j] \leq n-1$  (untuk semua  $0 \leq j \leq m-1$ )
- $1 \leq y[j] \leq \min(h[l[j]], h[r[j]])$  (untuk semua  $0 \leq j \leq m-1$ )
- $0 \leq s, g \leq n-1$
- $s \neq g$
- Tidak ada dua jembatan layang yang memiliki titik yang sama, kecuali mungkin pada ujung-ujungnya.

## Subsoal

1. (10 poin)  $n, m \leq 50$
2. (14 poin) Setiap jembatan layang berpotongan dengan maksimal 10 gedung.
3. (15 poin)  $s = 0, g = n-1$ , dan semua gedung memiliki tinggi yang sama.
4. (18 poin)  $s = 0, g = n-1$
5. (43 poin) Tidak ada batasan tambahan.

## Grader contoh

Grader contoh membaca masukan dengan format sebagai berikut:

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

Grader contoh mencetak sebuah baris yang berisi nilai `min_distance`.