

Respondiendo pedidos de Radiotaxi

Contribución de Guillermo García

Descripción del problema

Una empresa de radiotaxi despacha sus taxis libres a los clientes que los solicitan mediante una llamada telefónica a la central. Los taxis se encuentran en distintos lugares de la ciudad. Periódicamente la central recolecta pedidos de clientes, y luego asigna un taxi a cada cliente. La central nunca toma más pedidos que los taxis disponibles.

Para identificar qué taxi debe asignarse a qué cliente, la central conoce la ubicación actual de los **T** taxis y la ubicación de los **C** clientes que llamaron. Las ubicaciones se describen indicando la esquina más próxima. La ciudad es una cuadrícula de **N** calles norte-sur y **N** calles este-oeste, todas de doble sentido, y la velocidad de los taxis es constante: 1 minuto por cuadra.

Los taxis y los clientes están numerados desde 0, al igual que ocurre con las calles norte-sur y las este-oeste. La esquinas de la ciudad se indican con dos números: primero el de su calle norte-sur, luego el de su calle este-oeste.

Para ayudar a la central de taxis, se te solicita que escribas una función que devuelva el mínimo tiempo en el que es posible que **todos** los clientes estén sentados en su taxi.

Una vez asignado un taxi a un cliente, este se dirige a buscarlo siguiendo la ruta más corta posible. Todos los taxis asignados reciben sus indicaciones al mismo tiempo, y por lo tanto **se dirigen en simultáneo** cada uno hacia su cliente.

Se deben considerar únicamente los tiempos de estos viajes. Los tiempos invertidos en realizar el pedido, subir al taxi, y asignar los taxis (gracias a tu eficiente función) son despreciables.

Descripción de la función

Debes implementar la función

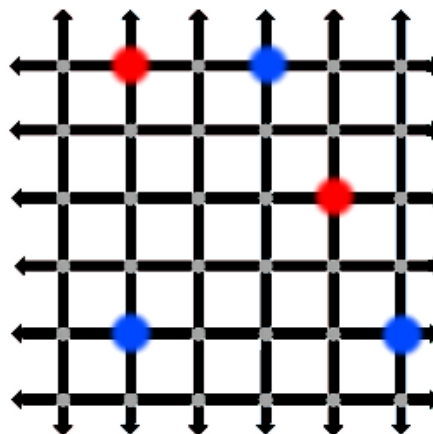
```
radiotaxi(
  taxiX, taxiY : ARREGLO[T] de ENTEROS;
  clienteX, clienteY,
  asignacion : ARREGLO[C] de ENTEROS
) : ENTERO
```

Sus parámetros son:

- taxiX, taxiY: Arreglos con las ubicaciones de los taxis. El taxi i se encuentra en la esquina $(\text{taxiX}[i], \text{taxiY}[i])$ ($0 \leq i < T$).
- clienteX, clienteY: Arreglos con las ubicaciones de los clientes. El cliente i se encuentra en la esquina $(\text{clienteX}[i], \text{clienteY}[i])$ ($0 \leq i < C$).
- asignacion: Arreglo en el cual se debe escribir la asignación de taxis a clientes.

La función deberá devolver en un entero, el menor tiempo posible hasta que todos los clientes ya hayan subido a un taxi.

Además, devuelve a través del arreglo `asignacion` el listado de taxis para cada cliente en una asignación óptima: `asignacion[0]` indica el número de taxi para el primer cliente, `asignacion[1]` el número para el segundo, y así siguiendo.



Ejemplo de ciudad con **N=6**

Evaluador

El evaluador local lee de la entrada estándar con el siguiente formato:

- Una línea con un entero: la cantidad taxis **T**.
- **T** líneas, cada una con dos enteros correspondientes a `taxiX[i]` y `taxiY[i]`.
- Una línea con un entero: la cantidad de clientes **C**.
- **C** líneas, cada una con dos enteros correspondientes a `clienteX[i]` y `clienteY[i]`.

El evaluador devuelve una primera línea con el valor devuelto por la función `radiotaxi`, y luego una segunda línea con los valores del arreglo `asignacion`, separados por espacios.

Puntaje

- 50 % por el tiempo mínimo posible
- 50 % por indicar una asignación óptima de taxis a clientes

Cotas

$$N \leq 10^9$$

$$1 \leq C \leq T \leq 100.000$$

$$C \leq 100$$

Ejemplo

Si se invoca al evaluador con la siguiente entrada:

```

3
1 1
3 5
5 1
2
1 5
4 3
    
```

Este llamaría a la función `radiotaxi` suministrando los datos correspondientes, y con una solución correcta podría devolver:

```

3
1 2
    
```

Ya que el cliente 0 será recogido por el taxi 1, y el cliente 1 por el taxi 2, dando un tiempo total hasta que todos han ingresado a su taxi de 3 minutos.

Este ejemplo corresponde a la ciudad de la figura anterior.

Subtareas

Habrà casos de prueba por 9 puntos en los que $T \leq 10$ y $N \leq 50$.

Habrà subtareas con C pequeño, y subtareas en las que todos los clientes y taxis compartirán la segunda coordenada (es decir, todos los `taxiY[i]` y los `clienteY[j]` serán iguales a un cierto valor v fijo).

El puntaje de estas subtareas se reparte de acuerdo a la siguiente tabla:

	Única calle	Sin rest.
$C = 1$	5	5
$C = 2$	7	8
$C \leq 10$	9	14
$C = T \leq 100$	12	8
$C \leq 100$	13	10

Notar además en la tabla que ciertas subtareas tienen $C = T$.