Distanciamiento físico

Contribución de Facundo Martín Gutiérrez [adaptación]

Descripción del problema

Pedro está haciendo las compras en su supermercado favorito, y acaba de poner en el carrito el último ítem de su lista de compras. Ahora tiene que dirigirse a la caja, pero debido a las medidas sanitarias dispuestas en la zona donde vive, decide hacerlo de forma de mantener la mayor distancia posible a las demás personas que se encuentran en el supermercado, es decir, **busca maximizar la mínima distancia a alguna de las personas en el supermercado**.

El supermercado puede modelarse mediante puntos en el plano. La distancia entre dos puntos es la longitud del segmento que los une. Pedro comienza en la posición $(0, 0)$ y la caja a la que debe dirigirse se encuentra en la posición (W, H). Las paredes del supermercado corresponden a las rectas $x = 0$, $x = W$, $y = 0$ e $y = H$. **Pedro no puede salirse de los límites del supermercado en su recorrido a la caja**.

Actualmente en el supermercado se encuentran N personas, la i-ésima de ellas se encuentra en la posición (x_i, y_i). Para este modelo simplificado suponemos a las personas como puntos del plano que se mantienen en la misma posición en todo momento, y no hay dos personas en la misma posición (x_i, y_i).

Si la máxima distancia posible es d, se debe retornar $4 \cdot d^2$ (se puede demostrar que $4 \cdot d^2$ siempre será un entero).

Detalles de implementación

Debes implementar la función `distanciamiento(W,H,x,y)`, que recibe:

- `W,H`: Enteros, que denotan las coordenadas en los ejes x e y, respectivamente, correspondientes a la ubicación de la caja donde Pedro debe dirigirse.
- `x,y`: Arreglos de N posiciones, que contienen el valor correspondiente a la posición en el plano de la i-ésima persona. El arreglo x corresponde al valor de abscisas (eje x) y el arreglo y a las ordenadas (eje y).

Debe retornar un entero con el cuádruple del cuadrado de la máxima distancia posible a la que puede pasar Pedro de todas las personas y aún así lograr su objetivo.

Evaluador local

El evaluador local lee de la entrada estándar:

- Una primera línea con 3 enteros: N, W y H, separados por un espacio.
- N líneas, cada una de ellas con 2 enteros: x_i e y_i separados por un espacio.

Escribe a la salida estándar el resultado retornado por la función.

Cotas

- $1 \leq N \leq 4.000$
- $2 \leq W, H, \leq 1.000.000$
- $(x_i, y_i) \neq (x_j, y_j)$ si $i \neq j$
- $1 \leq x_i \leq W - 1$
- $1 \leq y_i \leq H - 1$

Las últimas dos restricciones dicen que **no habrá personas en las paredes del supermercado**, a excepción de Pedro.

Ejemplos

Si el evaluador local recibe la siguiente entrada:

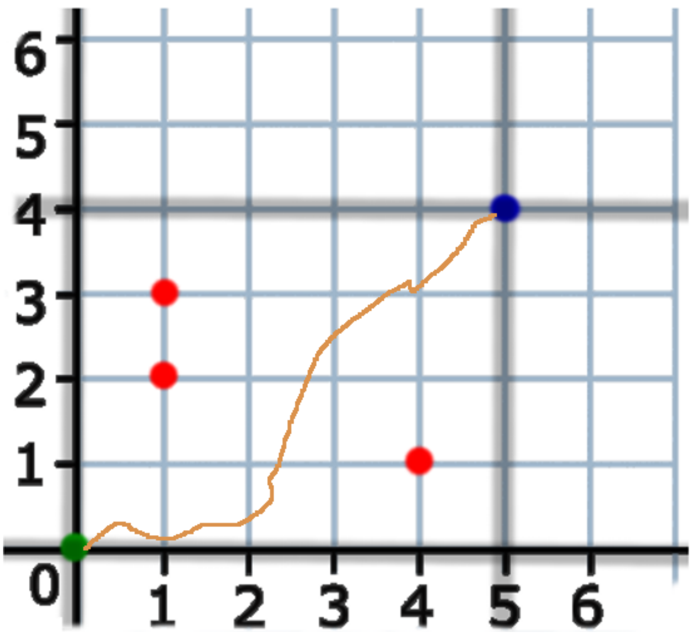
```
3 5 4
1 3
4 1
1 2
```

Para una implementación correcta escribirá:

```
10
```

Ya que la distancia máxima posible en este caso es $\frac{\sqrt{10}}{2}$.

La representación en el plano de la entrada del ejemplo puede verse en la siguiente figura, junto a un ejemplo de un camino óptimo posible.



En cambio para:

```
1 2 2
1 1
```

La respuesta es:

```
4
```

Subtareas

1. $N = 1$ (7 puntos)
2. $N, W, H \leq 100$ (9 puntos)
3. $W, H \leq 1.000$ (11 puntos)
4. $N \leq 100$ (39 puntos)
5. $N \leq 1000$ (21 puntos)
6. Sin más restricción (13 puntos)