



# III CodeBoost

## ETSI Sistemas Informáticos - 2026

<http://blogs.upm.es/codeboost>

### Cuadernillo de problemas

17 de marzo de 2026



Proyecto de innovación educativa Challenger (IE26.6110) financiado por la Universidad Politécnica de Madrid



## Índice

A Red cuántica orbital	3
B La guarida de los drones	5
C Influencer Conf 2026	7
D <i>Panopticon</i>	9
E El archipiélago de Tropicocanía	11

Autores de los problemas:

- Alberto Díaz Álvarez (Universidad Politécnica de Madrid)
- Jorge Dueñas Lerín (Universidad Politécnica de Madrid)
- Guillermo Iglesias Hernández (Universidad Politécnica de Madrid)
- Fernando Ortega Requena (Universidad Politécnica de Madrid)



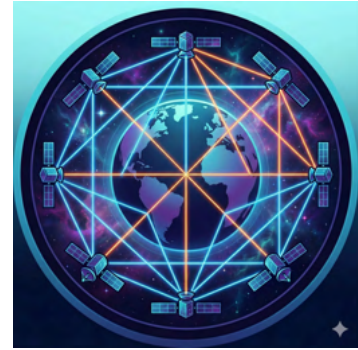
Tiempo: 1 segundo

# ● A

## Red cuántica orbital

La agencia espacial **AstroX** ha desplegado una constelación de  $N$  satélites en la órbita baja terrestre. Para que la red funcione, todos los satélites deben estar conectados entre sí, ya sea de forma directa o a través de otros satélites (comunicación indirecta).

Existen  $M$  posibles enlaces láser que se pueden activar entre pares de satélites. Cada enlace tiene un **coste de energía** específico basado en la distancia y la interferencia atmosférica. Tu misión es seleccionar qué enlaces activar para que todos los satélites queden integrados en una única red, minimizando el coste total de energía.



### Entrada

- La primera línea contiene un entero  $N$  con el número de satélites.
- Las siguientes líneas contienen tres enteros cada una:  $u$ ,  $v$  y  $w$ .
  - $u$  y  $v$ : los identificadores de los satélites (de 0 a  $N - 1$ ).
  - $w$ : el coste de energía para activar el enlace entre  $u$  y  $v$ . Ten en cuenta que el coste del enlace de  $u$  a  $v$  es **el mismo** que de  $v$  a  $u$ .

### Salida

- Un único entero que represente el coste mínimo total para conectar todos los satélites.
- Si es imposible conectar todos los satélites con los enlaces disponibles, imprime  $-1$ .

### Entrada de ejemplo

La siguiente entrada muestra una entrada en la que es posible conectar todos los satélites:

```
4
0 1 10
1 2 15
0 2 5
2 3 30
1 3 8
```

### Salida de ejemplo

```
23
```

### Entrada de ejemplo

La siguiente entrada muestra una entrada en la que **no** es posible conectar todos los satélites:

```
4
0 3 20
1 2 50
```

## Salida de ejemplo

-1

## Límites

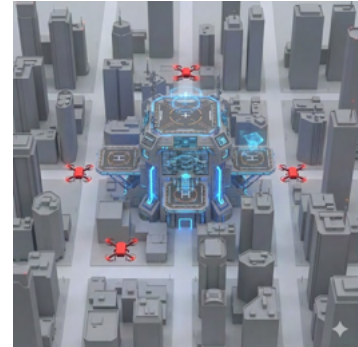
- $2 \leq N \leq 500$
- $1 \leq w \leq 100$

Tiempo: 0.3 segundos



## La guarida de los drones

En una ciudad futurista la gente vive organizada una cuadrícula de infinitos edificios. La Agencia de reparto **RepartX** quiere construir su centro logístico en unas coordenadas  $(X, Y)$ , desde el que ciertos drones deberán abastecer a la sociedad futura, ansiosa por productos absurdos e inútiles. Como en el futuro *ciber-hiper-capitalista* la optimización de recursos es el objetivo final de la vida, RepartX quiere posicionar su centro de reparto en el mejor punto posible.



La ubicación  $(X, Y)$  del centro logístico puede situarse en cualquier coordenada entera en la cuadrícula, ya que los decimales los ocupan edificios. El consumo de energía de un dron para moverse desde su puesto de carga  $(x_i, y_i)$  hasta el punto de encuentro  $(X, Y)$  se calcula utilizando la “*Distancia de Manhattan*” (ya que los drones solo pueden moverse horizontal y verticalmente a través de los edificios de la ciudad), donde cada unidad de movimiento corresponde con un coste de uno:

$$Distancia_{Manhattan} = |x_i - X| + |y_i - Y|$$

Tu tarea es encontrar una coordenada  $(X, Y)$  tal que la suma total de la energía consumida por todos los drones sea mínima y calcular el coste de energía de los drones para llegar allí.

### Entrada

- La primera línea contiene un entero:  $N$  (número de drones)
- Las siguientes  $N$  líneas contienen dos enteros cada una:  $x_i$  y  $y_i$ , coordenadas de cada dron.

### Salida

- Un único entero que represente el coste total de desplazamiento de los drones para la mejor ubicación del centro logístico.

### Entrada de ejemplo

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

### Salida de ejemplo

```
12
```

### Límites

- $1 \leq N \leq 10^4$
- $-10^4 \leq x, y \leq 10^4$



Tiempo: 0.1 segundos

# ● C Influencer Conf 2026

Te encuentras en la entrega de premios de la Influencer Conf 2026. Tu objetivo es cruzar el salón principal desde la entrada hasta el escenario para recoger tu premio al ‘Mejor Uso de Filtros’. Pero hay un problema: el salón está lleno de Haters que quieren pedirte explicaciones por tus últimos vídeos.

Para mantener tu aura de influencer y no perder el estilo, tienes que moverte siguiendo un protocolo estricto de comunicación. Si rompes el ritmo, es *cringe* total y pierdes seguidores.



## Las Reglas de la ‘Vibe’

- **Movimiento Alterno:** En cada paso que des, debes decir tu frase característica. El primer paso debe ser ‘Bro’, el segundo ‘En plan’, el tercero ‘Bro’, el cuarto ‘En plan’, y así sucesivamente. No puedes repetir la misma frase dos veces seguidas.
- **Esquivar Haters:** El salón es una matriz de  $N \times M$ . Los haters están situados en todas las casillas  $(r, c)$  donde la suma de las coordenadas  $r + c$  sea múltiplo de un número  $K$  (excepto en la entrada y la salida, que son zonas seguras).
- **Movilidad:** Puedes moverte en las 4 direcciones (arriba, abajo, izquierda, derecha), pero nunca puedes pisar una casilla con un *hater* ni salirte del mapa.
- **Llegada:** Siempre comienzas a moverte diciendo ‘Bro’ y al llegar debes terminar diciendo ‘En plan’.

## Entrada

- La primera línea contiene un entero  $T$ : el número de casos de prueba.
- Cada caso ocupa una línea con cinco enteros:  $N, M, K, R$  y  $C$ .
  - $N$  y  $M$  son las dimensiones de la cuadrícula (filas y columnas).
  - $K$  es el divisor de los haters.
  - $(R, C)$  son las coordenadas del destino (fila y columna).
- La casilla de inicio es siempre  $(0, 0)$ . El destino puede coincidir con el inicio.

## Salida

Por cada caso imprime:

- El número mínimo de pasos para llegar al escenario manteniendo el ritmo de ‘Bro’ / ‘En plan’.
- ‘*six seven, six seven*’ si es imposible llegar.

## Entrada de ejemplo

```
5
3 3 100 2 2
3 3 100 2 1
3 3 3 2 2
3 3 1 0 0
3 3 1 2 2
```

## Salida de ejemplo

```
4
six seven, six seven
six seven, six seven
0
six seven, six seven
```

## Límites

- $1 \leq T \leq 100$
- $1 \leq N, M \leq 500$
- $1 \leq K \leq 1000$
- $0 \leq R < N, 0 \leq C < M$

Tiempo: 0,25 segundos

# ● D Panopticon

Año 2184. La Autoridad gobierna la última ciudad humana, manteniendo un férreo orden absoluto. El libre albedrío ha sido catalogado oficialmente como una patología social obsoleta que pone en riesgo la estabilidad del colectivo.

Se ha aislado un nuevo brote de disidencia: una “*idea corrupta*” que infecta la mente como un virus. Esta infección no es inmediata; está limitada físicamente por una velocidad máxima bautizada como **velocidad de sedición** ( $C$ ).

El sistema de vigilancia ha detectado anomalías biométricas —un parpadeo fuera de ritmo, un pulso acelerado— en sujetos ubicados en coordenadas  $(x, y, z)$  y tiempo  $(t)$ .



## La posibilidad es culpabilidad

Como Inquisidor del Ministerio tu mandato es la **ejecución preventiva**. Debes identificar a cualquier par de ciudadanos  $(A, B)$  cuya interacción haya sido físicamente posible. Se considera que ha habido contacto si la distancia euclídea  $d$  entre sus ubicaciones es menor o igual a la distancia que la infección pudo recorrer en el tiempo transcurrido.

$$d(A, B) \leq C \times |t_A - t_B|$$

La presunción de inocencia es un riesgo innecesario. Si la física permitió la transmisión, el silencio es obligatorio.

### Entrada

La primera línea contiene dos enteros  $N$  y  $C$ :

- $N \in [1, 2000]$ : El número de sujetos bajo vigilancia.
- $C \in [1, 1000]$ : La velocidad de sedición.

Las siguientes  $N$  líneas representan los registros biométricos, cada una con cuatro enteros  $t, x, y, z$  con  $0 \leq t, x, y, z \leq 10^9$  representando el momento de la anomalía y sus coordenadas espaciales.

### Salida

Un único entero con el número total de pares de ciudadanos que cumplen la condición de contacto físico posible. Todos serán ejecutados, este valor se usará para estadísticas.

### Entrada de ejemplo

```
3 100
0 0 0 0
10 500 0 0
20 2000 0 0
```

### Salida de ejemplo

```
2
```



Tiempo: 0.1 segundos



## El archipiélago de Tropicocéanía

En el remoto archipiélago paraíso fiscal de Tropicocéanía se quiere hacer un proyecto arquitectónico nunca antes visto. El país de Oceanía esta actualmente en guerra con Esteasia y quiere demostrar opulencia para ganar la batalla mental, por lo que su objetivo es hacer un proyecto sorprendente, aunque ello conlleve un despilfarro tremendo.

Para ello, se quiere construir un archipiélago de  $N$  islas, para el cual se debe organizar un sistema de puentes. El Rey ha dictado tres reglas extrañas para la construcción de puentes (caminos bidireccionales):

1. Cada puente debe conectar exactamente dos islas distintas.
2. No puede haber más de un puente entre las mismas dos islas.
3. **La Regla de Oro:** Para cualquier grupo de 3 islas, debe haber exactamente un puente o tres puentes conectándolas entre sí. (Es decir, no pueden tener ni 0 ni 2 puentes internos).



Dado  $N$ , debes determinar el número máximo de puentes que se pueden construir en total sin romper la Regla de Oro.

### Entrada

- El programa recibe una línea, correspondiente al número total de islas  $N$ .

### Salida

- Un único entero que sea el número máximo de puentes construibles en el archipiélago.

### Entrada de ejemplo

4

### Salida de ejemplo

6

### Límites

- $3 \leq N \leq 10^9$