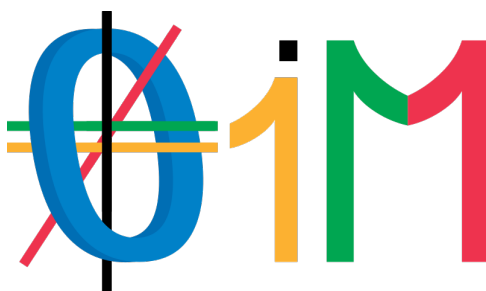


# IV Olimpiada Informática de Madrid

16 y 17 de febrero de 2024



Cuadernillo de problemas



Facultad  
de  
Informática



UNIVERSIDAD COMPLUTENSE  
MADRID

17 de febrero de 2024

*Every great developer you know got there by solving problems  
they were unqualified to solve until they actually did it*  
**Patrick McKenzie**

17 de febrero de 2024

## Listado de problemas

<b>A. Sopas de letras</b>	<b>3</b>
<b>B. El sensor de las naves Klingon</b>	<b>5</b>
<b>C. Foto de grupo</b>	<b>7</b>
<b>D. Mejorando el servicio de atención al cliente</b>	<b>9</b>
<b>E. Las películas de Bella y Bestia</b>	<b>11</b>
<b>F. Triángulos y más triángulos</b>	<b>13</b>

Autores de los problemas:

- Paco Criado (CUNEF Universidad)
- Marco Antonio Gómez Martín (Universidad Complutense de Madrid)
- Pedro Pablo Gómez Martín (Universidad Complutense de Madrid)
- Manuel Montenegro Montes (Universidad Complutense de Madrid)
- Isabel Pita Andreu (Universidad Complutense de Madrid)
- Javier del Río López (Universidad Complutense de Madrid)
- Rubén Rubio (Universidad Complutense de Madrid)
- Alberto Verdejo (Universidad Complutense de Madrid)



# A

## Sopas de letras

Las sopas de letras han cautivado a entusiastas de las palabras y amantes de los pasatiempos durante generaciones. Originarias de los crucigramas, las sopas de letras ofrecen un desafío único al ocultar palabras entre una cuadrícula de letras. Con el tiempo, las sopas de letras se han convertido en un componente esencial de revistas, libros de pasatiempos y aplicaciones móviles dedicadas al entretenimiento cerebral. Este desafío no solo pone a prueba la agudeza mental de quienes se aventuran a buscar palabras, sino que también ha dejado una huella perdurable en la cultura de los pasatiempos y juegos de palabras.

Las sopas de letras consisten en encontrar palabras dentro de un rompecabezas de letras dispuestas en filas y columnas. Las palabras pueden estar dispuestas en cualquier dirección: ya sea de forma horizontal, vertical o diagonal, incluso en dirección inversa (de abajo arriba o de derecha a izquierda), desafiando a los buscadores de palabras a descubrirlas en un fascinante juego de observación y concentración.

En este problema se te presenta una sopa de letras y una lista de palabras para encontrar. Tu tarea es determinar cuántas de esas palabras aparecen realmente en la sopa de letras. ¡Sumérgete en el fascinante mundo de las sopas de letras y demuestra tu destreza en este desafío de palabras!

P	R	O	G	R	A	M	A	R	M
X	R	U	I	J	K	A	D	B	C
O	L	I	S	T	O	E	A	W	B
A	C	R	N	G	B	Z	I	S	C
I	S	O	I	M	E	R	P	Q	U
R	L	D	G	V	I	V	M	S	O
S	O	A	E	T	M	D	I	U	L
C	B	B	O	B	U	C	L	E	S
J	O	I	N	S	U	A	O	R	T
V	E	T	D	E	S	T	X	U	R

### Entrada

La entrada comienza con el número  $T$  de casos de prueba que aparecerán a continuación.

La primera línea de cada caso contiene dos números, el número  $N$  de filas de la sopa de letras y el número  $M$  de columnas, seguidos por  $N$  líneas de  $M$  caracteres cada una, representando la sopa de letras. A continuación aparece el número  $Q$  de palabras que hay que buscar en la sopa, seguido por una línea con esas  $Q$  palabras, todas distintas.

Tanto la sopa de letras como las palabras a buscar solamente tienen letras mayúsculas del alfabeto inglés.

### Restricciones

- $1 \leq N, M \leq 50$
- $1 \leq Q \leq 100$
- Las longitudes de las palabras a buscar están entre 1 y 15.

### Subtareas

1. (5 puntos)  $N = 1$  y las palabras a buscar solamente tienen una letra.
2. (10 puntos)  $N = 1$  y las palabras solamente se encuentran colocadas de izquierda a derecha.
3. (35 puntos) Las palabras solamente se encuentran en horizontal (de izquierda a derecha) o en vertical (de arriba abajo).
4. (50 puntos) Sin restricciones adicionales.

### Salida

Para cada caso de prueba se escribirá una línea con el número de palabras a buscar que realmente se encuentran en la sopa de letras en cualquier dirección.

## Entrada de ejemplo

```
2
10 10
PROGRAMARM
XRUIJKADBC
OLISTOEAWB
ACRNGBZISC
ISOIMERPQU
RLDGVIVMSO
SOAETMDIUL
CBBOBUCLES
JOINSUAORT
VETDESTXUR
8
PROGRAMA BUG OLIMPIADA PREMIOS BIT BUCLE CODIGO CUBICO
2 7
SOPALOS
ABCDEFGG
2
SOPA PALOS
```

## Salida de ejemplo

```
6
2
```

# B

## El sensor de las naves Klingon

El capitán James T. Kirk y Mr. Spock se han infiltrado en una nave Klingon. Cada uno de ellos se encuentra en un extremo de la nave junto con un grupo de soldados fuertemente armados. Por desgracia, desconocían el poderoso sensor de equilibrio que sus enemigos han instalado recientemente en sus naves, capaz de detectar los más pequeños cambios en la distribución del peso.



La doble incursión, por dos lugares diferentes, ha hecho que el peso de la nave se desequilibre y ha disparado las alarmas del sensible sensor. Sin esperar un instante, los Klingon se han puesto a explorar la nave buscando una explicación. ¡Hay que restaurar el equilibrio para tranquilizar al sensor y que no descubran el asalto!

La única solución posible es conseguir que las dos tropas pesen exactamente lo mismo y la nave vuelva así al estado de equilibrio. Mr. Spock ha determinado que, por culpa de la redada en curso, es posible hacer llegar, como mucho, un soldado de cada tropa a la contraria. Ahora tiene que decidir a qué soldados pide que se muevan para estabilizar la nave.

### Entrada

La entrada consta de una serie de casos de prueba. Cada caso comienza con una línea en la que se indica el número de soldados que tiene cada tropa,  $N_1$  y  $N_2$ . En la segunda línea se muestran  $N_1$  números con los pesos de los integrantes de la primera tropa y en la tercera línea los  $N_2$  pesos de los integrantes de la segunda.

### Restricciones

- $1 \leq N_1, N_2 \leq 400.000$
- Los pesos son valores entre 1 y  $10^8$

### Subtareas

1. (5 puntos) Cada grupo está formado por una única persona.
2. (10 puntos) Cada grupo está formado como mucho por 2 personas.
3. (15 puntos) Los grupos tienen cómo máximo 100 integrantes.
4. (15 puntos) El peso máximo de los soldados es 100.
5. (55 puntos) Sin restricciones adicionales.

### Salida

Por cada caso de prueba se escribirá el peso de una de las tropas tras equilibrarlas. Si resulta imposible igualar el peso de las dos tropas moviendo como mucho un soldado de cada una se escribirá el mensaje NO SE PUEDE.

### Entrada de ejemplo

```
2 2
5 5
4 4
6 3
3 5 2 2 4 1
5 5 5
4 3
4 1 3 2
10 12 4
3 2
20 5 11
1 3
```

### Salida de ejemplo

```
9
16
18
NO SE PUEDE
```



# C

## Foto de grupo

Bienvenidos a la escuela “Pequeños Exploradores”, donde la aventura y el aprendizaje van de la mano. En esta ocasión, los intrépidos estudiantes se han aventurado en una excursión a una estupenda montaña cercana, con un río donde podrán tirar piñas, que desemboca en un lago donde podrán darse un chapuzón; un lugar lleno de sorpresas y maravillas naturales.

Con mochilas llenas de meriendas y cámaras repletas de expectativas listas para capturar cada momento, nuestros pequeños aventureros se han agrupado en parejas formadas por una niña y un niño, donde cada uno llama a su pareja su “gemelo”. El objetivo es asegurarse de que nadie se pierda en el camino. Además, las manos entrelazadas crean un lazo inquebrantable de amistad y seguridad.



La maestra, una sabia guía en este viaje lúdico-educativo, es conocida por sus entretenidas historias y su habilidad para convertir cada momento en una valiosa lección. Pero consciente de que la curiosidad infantil puede transformarse en caos en cualquier momento, ha decidido immortalizar en una foto grupal ese breve instante de calma antes de que la algarabía comience. Los niños se han alineado en una fila bastante larga, cada uno agarrado de la mano de su inseparable gemelo. Y ella se ha colocado entre dos parejas, temiendo que el buen rato que están pasando termine en cualquier momento, mientras el conductor del autobús hace la foto. . .

Y eso es lo último que se ha sabido de la maestra, los niños y el conductor. Nadie sabe su paradero actual. La policía lleva buscándolos desde entonces, y el inspector Lestra D. encargado del caso, lo mejorcito de un grupo de torpes, no sabe ni por dónde empezar.

El detective Sherloquillo Bromes, mucho más sagaz, está convencido de que en la última foto que se hicieron está la clave para encontrarlos. En particular, cree que la posición desde la izquierda que ocupó la maestra cuando se colocaron para la instantánea es la pista que le ayudará a saber su paradero. Por desgracia, Lestra D. guarda a buen recaudo la foto, para proteger la intimidad de los pequeños, y no está dispuesto a enseñársela. Sherloquillo quiere saber la posición que ocupa la maestra, pero no quiere preguntársela directamente a Lestra D. para no desvelar sus cartas tan pronto y que termine él llevándose el mérito de la resolución del caso. Prefiere averiguarlo a través de preguntas indirectas.

Pero la perspicacia de Bromes no es suficiente y te ha pedido ayuda con las preguntas. Tendrás que diseñar un programa que primero lea (de la entrada estándar) el número  $N$  de personas que aparecen en la foto (niños, niñas y la maestra). Después, el programa podrá hacer preguntas sobre la foto, del estilo “¿Quién se encontraba en la posición  $p$ ?”, escribiendo “?  $p$ ”. Las posiciones se numeran de izquierda a derecha comenzado por el número 1. A continuación el programa podrá leer la respuesta (transcrita de la proporcionada por Lestra D.), que será una ‘A’, si en esa posición había una niña, una ‘O’, si en esa posición había un niño, y una ‘M’, si en esa posición estaba la maestra. El programa deberá seguir haciendo consultas hasta que encuentre a la maestra en la foto. En ese momento escribirá “Respuesta:”, seguido de un espacio en blanco y la posición donde se ha encontrado a la maestra. Resulta que hay diferentes versiones de la foto, por lo que este proceso completo se puede repetir varias veces. El proceso terminará cuando al leer el número de personas de una foto se obtenga un 0.

A continuación aparece un ejemplo de ejecución. En **negrita** aparece lo escrito por el programa y en  *cursiva* las respuestas de Lestra D. que se le han proporcionado.

*11*

**? 1**

*A*

**? 2**

*O*

**? 3**

*M*

**Respuesta: 3**

1001

? 999

M

Respuesta: 999

0

### Restricciones

- $3 \leq N \leq 10^9$ . Si hay  $P$  parejas de alumnos,  $N = 2 \times P + 1$ .
- En todas las fotos los alumnos están por parejas, y la niña aparece a la izquierda del niño.
- La maestra se puede encontrar entre dos parejas, o la izquierda o a la derecha de todas las parejas. Nunca estará entre los dos componentes de una misma pareja.

### Subtareas

1. (9 puntos)  $N \leq 5$ .
2. (6 puntos) La maestra se encuentra en un extremo de la foto.
3. (20 puntos)  $N \leq 1.000$ .
4. (65 puntos) Sin restricciones adicionales.

### Notas

Al tratarse de un problema *interactivo* es importante que cada vez que tu solución escriba algo que el juez deba leer se asegure de volcar la salida (usando terminología inglesa, haga “*flush*”). La forma de hacerlo varía entre lenguajes. En los admitidos en la competición puede hacerse con:

- C++: `cout << endl;` o `cout << flush;`
- C: `fflush(stdout);`
- Java: `System.out.flush();`
- Python: `print(..., flush=True)` o `sys.stdout.flush()`

# D

## Mejorando el servicio de atención al cliente

Como miembro del comité de empresa, estás recibiendo muchas quejas por parte de los trabajadores del servicio de atención al cliente que tratan cara a cara con estos.

En los trabajos de cara al público, es fácil caer en la desesperación cuando se trata con la gente, sobre todo si es de forma continuada en el tiempo. Es por eso que le has propuesto al jefe la idea de que los trabajadores tengan garantizados unos minutos de descanso entre cliente y cliente.



El jefe no está en contra de este tipo de descansos, de hecho, parece no importarle lo que duren, siempre y cuando en el minuto exacto en el que termine la jornada laboral se haya terminado de atender a todos los clientes de ese día.

Gracias a un avanzado sistema de citas, se pueden conocer con anterioridad los detalles de todos los clientes a los que se tendrá que atender cada día: el momento de llegada y el tiempo exacto que va a necesitar para ser atendido.

Los clientes siempre se comportan de la misma forma: hacen cola por orden de llegada para ser atendidos, y en el momento en el que la primera persona de la cola encuentra un mostrador con un trabajador disponible, se dirige a este para ser atendido. No se considera como disponible a un trabajador mientras se encuentre descansando. El tiempo que un cliente tarda en avanzar puestos en la cola o en desplazarse desde el primer puesto hasta cualquier mostrador es despreciable, por lo que se considerará 0.

### Entrada

La entrada comienza con el número de casos de prueba que vendrán a continuación. Cada caso describe a lo largo de varias líneas una posible situación en la empresa.

La primera línea de cada caso de prueba contiene tres números: el número  $N$  de trabajadores que ese día estarán trabajando de cara al público, el número  $C$  de clientes que llegarán a lo largo del día para hacer algún trámite, y el número  $J$  de minutos que tiene la jornada laboral de todos los trabajadores. Se considera que la jornada laboral empieza en el minuto 0, por lo que no se podrá atender a los clientes a partir del minuto  $J$ .

La siguiente línea del caso de prueba contiene los valores  $m_1, m_2, \dots, m_C$ , indicando el minuto de llegada de cada cliente. Un cliente puede empezar a hacer un trámite en el mismo minuto de llegada, siempre que encuentre a algún trabajador disponible. Los clientes harán cola en el mismo orden en el que aparecen en la entrada.

La siguiente línea del caso de prueba contiene los valores  $t_1, t_2, \dots, t_C$ , con el número de minutos necesarios para atender a cada cliente. Cada cliente llegará con tiempo suficiente para que se le termine de atender antes del fin de la jornada si es atendido nada más llegar.

### Restricciones

- $1 \leq N \leq 10^4$
- $1 \leq C \leq 10^5$
- $1 \leq J \leq 10^9$
- $0 \leq m_i \leq m_{i+1} < J$
- $1 \leq t_i \leq 10^9$
- $m_i + t_i \leq J$

### Subtareas

1. (5 puntos) Nunca habrá más clientes que trabajadores.

2. (15 puntos)  $N = 1$ , todos los clientes llegan en el minuto 0 y ninguno necesita más de 1000 minutos para atender su solicitud.
3. (20 puntos)  $N \leq 10$ ,  $C \leq 1000$ ,  $J \leq 1000$ .
4. (25 puntos)  $J \leq 1000$ .
5. (35 puntos) Sin restricciones adicionales.

### Salida

Por cada caso de prueba, el programa escribirá un número no negativo  $Z$  con el máximo número de minutos que los empleados pueden tomarse de descanso entre cada par de clientes que atienden sin que por ello se deje de atender a ninguno dentro de la jornada laboral. Ten en cuenta que todos los trabajadores descansarán siempre al menos  $Z$  minutos entre cliente y cliente y se debe garantizar que da tiempo a atender a todo el mundo.

Si se puede conseguir que cada empleado atienda como mucho a un cliente durante toda la jornada, se escribirá DIA TRANQUILO.

Si es imposible atender a todos los clientes dentro de la jornada laboral, incluso sin tomarse ningún descanso obligatorio, se escribirá FALTAN RECURSOS.

### Entrada de ejemplo

```
4
2 3 10
0 1 4
5 4 3
3 4 10
0 1 5 6
8 9 3 3
5 5 60
3 6 9 12 15
45 45 45 45 45
1 4 100
0 2 13 25
20 20 20 20
```

### Salida de ejemplo

```
2
FALTAN RECURSOS
DIA TRANQUILO
6
```

# E

## Las películas de Bella y Bestia

En el cuento clásico de “*La Bella y la Bestia*”, Bella se ve obligada a vivir en un castillo con Bestia, un ser deforme e intimidante víctima del hechizo de una bruja. Con el paso del tiempo, Bella termina enamorándose de Bestia, lo que hace que se rompa el hechizo, Bestia se convierta en un apuesto joven y sean felices y coman perdices. En la adaptación de la historia que hizo Disney en su película de 1991, el punto de inflexión en el que Bella comienza a darse cuenta de que podría llegar a querer a Bestia a pesar de su aspecto y modales rudos ocurre cuando descubre la biblioteca del castillo. Apasionada a la lectura, saber que a Bestia le gustan también los libros hasta el punto de acumular miles de volúmenes planta en ella la semilla que conducirá al final feliz.



Lo que en la película no se cuenta es que, tras ese final feliz, durante su vida juntos llegaron a tener una colección similar de películas, primero en vídeo, después en DVD y luego en Blu-Ray (era 1991, las plataformas digitales aún no habían aterrizado...). Y en el mantenimiento de la colección apareció una de las diferencias fundamentales en el carácter de ambos: el empeño por mantener el orden.

Al principio, cuando Bella colocaba las películas, todas ellas quedan en la estantería por orden alfabético. La primera película (en lugar de su título, diremos que es la película 1) a la izquierda del todo y la última (la película  $n$  siendo  $n$  el número total de películas en la colección) a la derecha. En ese momento todo es fácil: la película  $k$  está en la posición  $k$  si contamos de izquierda a derecha.

El desorden llega cuando Bestia es el encargado de guardar la película que acaban de ver. En lugar de colocarla en su sitio original, la pone al final del todo, empuja todo el bloque hacia la izquierda y se queda tan a gusto, cambiando de un plumazo las posiciones de todas las películas que estaban a la derecha de la que acaban de ver. Y así no hay luego quien encuentre nada.

### Entrada

La entrada está compuesta por distintos casos de prueba. Cada uno comienza con una línea con el número  $N$  de películas de la colección seguido de otra línea con el número de operaciones que se hacen sobre la estantería.

Cada operación aparece en una línea independiente que comienza con un carácter que indica el tipo de operación. Una  $V$  indica que se realiza el *visionado* de una película y su posterior devolución a la estantería por parte de Bestia. Una  $Q$  indica la consulta de la posición que ocupa una película en la estantería. Ambas operaciones van seguidas del identificador de la película vista o consultada (o lo que es lo mismo, la posición que ocupa la película en la estantería en la colocación inicial de Bella).

### Restricciones

- $1 \leq N \leq 500.000$
- El número de operaciones de cada tipo es menor o igual que 200.000

### Subtareas

1. (5 puntos) No hay operaciones de visionado.
2. (5 puntos) Las operaciones de consulta preguntan siempre por la última película vista.
3. (12 puntos) Hay una única operación de visionado.
4. (15 puntos) Nunca se ve la misma película más de una vez y las consultas son siempre sobre alguna de las cinco últimas películas vistas.
5. (18 puntos) Hay como mucho 100 películas y 100 operaciones de cada tipo.
6. (45 puntos) Sin restricciones adicionales.

## Salida

Por cada operación de consulta (Q) se escribirá una línea independiente indicando qué posición de la estantería ocupa la película indicada en ese momento teniendo en cuenta que la película colocada más a la izquierda ocupa la posición 1.

Tras cada caso de prueba aparecerá una línea con tres guiones, ---.

## Entrada de ejemplo

```
5
5
Q 2
Q 3
V 2
Q 2
Q 3
5
3
Q 1
V 5
Q 1
```

## Salida de ejemplo

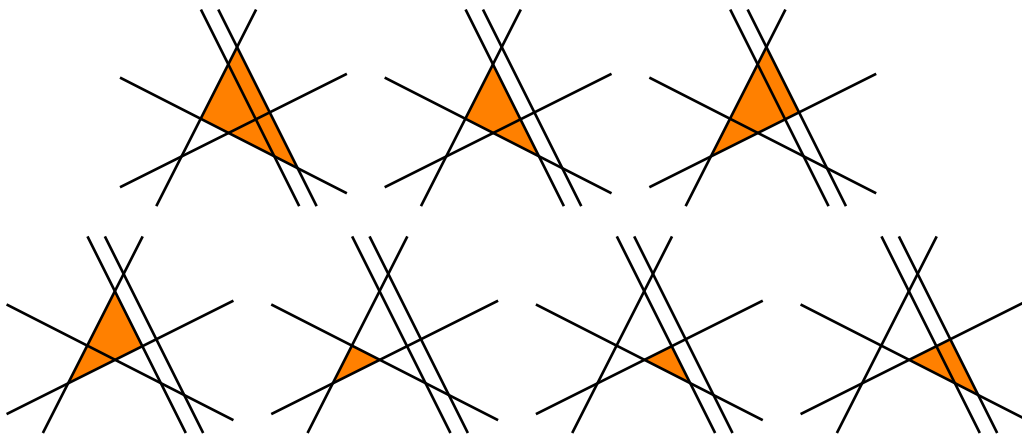
```
2
3
5
2
---
1
1
---
```

# F

## Triángulos y más triángulos

El profesor de dibujo técnico de Juan está chapado a la antigua. Para una persona acostumbrada a enseñar a dibujar mediante escuadra, cartabón y *rottings*, tener que explicar cómo funciona el programa de Diseño Asistido por Computador (CAD) es todo un reto.

Mientras el profesor intenta, a duras penas, explicar la diferencia entre dibujos vectoriales y mapas de bits, Juan se distrae pintando líneas rectas en el programa de CAD. Eso sí, aunque no atiende, curiosidad no le falta. Se ha dado cuenta de que al dibujar las tres primeras rectas, ha formado un triángulo. Al añadir una recta más, se han formado cuatro. Llevado por la emoción, ha seguido añadiendo más rectas y ahora está contando el número total de triángulos que ha formado. Por supuesto, la tarea no es tan sencilla como parece, ya que hay triángulos dentro de triángulos, y a su vez dentro de más triángulos que se unen para formar más triángulos, etc. Por ejemplo, al pintar cinco rectas de la forma que se muestra a continuación, ha podido formar siete triángulos distintos:



A veces, cuando finaliza un dibujo, Juan se da cuenta de que el número de triángulos que consigue con esas rectas podría ser aún mayor. Pero no pasa nada, porque ha aprendido a desplazar cualquier recta de su dibujo a lo largo del eje X (hacia la izquierda o derecha) o del eje Y (hacia arriba o hacia abajo) y en ocasiones eso le permite incrementar el número de triángulos finales. Eso sí, no puede cambiar la inclinación de las rectas porque aún no ha aprendido a hacer rotaciones... más le valdría atender en clase.

### Entrada

La entrada está compuesta por distintos casos de prueba. Cada uno comienza con una línea indicando el número  $n$  de rectas dibujadas. A continuación aparecen  $n$  líneas, cada una con la descripción de cada una de las rectas. La descripción de una recta viene dada por tres enteros  $a$ ,  $b$  y  $c$  representando a la recta  $ax + by = c$ .

La entrada finaliza con un caso sin rectas que no debe procesarse.

### Restricciones

1.  $n \leq 200.000$
2. Para cada recta  $ax + by = c$  se cumple que  $-10^9 \leq a, b, c \leq 10^9$  y  $(a, b) \neq (0, 0)$ .
3. En el dibujo no hay dos rectas superpuestas.

### Subtareas

1. (3 puntos) Todas las rectas son paralelas.

2. (10 puntos)  $n \leq 500$  No hay dos rectas paralelas y en sus posiciones actuales proporcionan, sin necesidad de desplazarlas, el máximo número de triángulos posible.
3. (15 puntos)  $n \leq 500$ .
4. (20 puntos)  $n \leq 5000$ .
5. (42 puntos)  $0 \leq a, b$ .
6. (10 puntos) Sin restricciones adicionales.

### Salida

Por cada caso de prueba se escribirá una línea indicando el máximo número de triángulos que se pueden formar desplazando las rectas sin rotarlas.

### Entrada de ejemplo

```
5
2 1 2
2 1 3
-2 1 2
1 -2 2
-1 -2 2
5
1 0 0
-2 0 -2
0 3 0
0 -1 1
1 1 2
4
1 1 1
1 1 -1
1 -1 1
1 -1 -1
0
```

### Salida de ejemplo

```
7
4
0
```

### Notas

La figura del enunciado se corresponde con el primer caso de prueba.