

# Problema A

## Enamorados a distancia

Ayer estuviste en un popular evento, y de entre todas las personas te fijaste en un chico. Era el chico de tus sueños. Guapo, alto, y muy majo cuando hablaste con él.

Sin embargo, por culpa del alcohol que llevabas encima, te olvidaste de pedirle el número de teléfono. Aun así, te acuerdas de que estuvisteis hablando de una red social que utilizais en común, así que decides buscarle por ahí.

Con todas tus esperanzas, vas a ir a buscarle, así que abres la aplicación con ilusión, hasta que te das cuenta del problema: La barra de búsquedas de la aplicación no devuelve ningún resultado, pongas lo que pongas.

Pensando que es un error de la aplicación, te dispones a actualizarla, pero en tu móvil ya no hay espacio para actualizar más. Pensando en otra forma de buscarle, se te ocurre mirar en tu lista de amigos, por si tuvierais algún amigo en común. Por suerte, las listas de amigos sí que cargan, así que aun tienes una oportunidad de encontrarle navegando a través de las listas de amigos de cada usuario en la aplicación.

Mirando entre las listas de amigos de tus amigos, y así sucesivamente, quizás tengas suerte y te topes con él. Te va a costar, pero lo intentas igual, es el chico de tus sueños y no puedes dejarle escapar.

Sabiendo que, en la aplicación, si eres amigo de un usuario, ese otro usuario es siempre también amigo tuyo, y que puedes empezar a buscar por tus propios amigos, que salen en tu perfil, ¿encontrarás al chico de tus sueños, o tendrás que conformarte con helado y galletas?

### Input

La entrada consistirá en un único caso, comenzando por un número  $U$ , la cantidad de usuarios de la red social (Incluyendote a ti mismo). Posteriormente, aparecerán  $U - 1$  líneas con todos los nombres de usuario de la aplicación (Excepto el tuyo).

A continuación, aparecerá un número  $N$  con la cantidad de amigos que tienes en la aplicación, seguido de  $N$  líneas, cada una con un nombre de un amigo tuyo.

Tras ello, aparecerá un número  $A$  de amistades realizadas en la aplicación, cada una descrita por una línea con dos nombres, los nombres de usuario de dos usuarios que son amigos entre ellos.

Finalmente acompañará una última línea con un nombre, el nombre de usuario del chico del que te has enamorado. Se garantiza que cada nombre de usuario es único en la aplicación, y que los nombres son cadenas de caracteres que solo contendrán letras mayúsculas y minúsculas del alfabeto inglés.

La entrada debe ser leída de forma estándar.

## Output

La salida contendrá una sola línea, indicando "ENCONTRE A MI PRINCIPE AZUL" si es posible encontrar al chico del que te has enamorado en la red social, o "HELADO Y GALLETAS" si no es posible encontrarlo.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
4 Jose Pedro Manuel 2 Jose Manuel 2 Jose Manuel Manuel Pedro Pedro	ENCONTRE A MI PRINCIPE AZUL

Entrada ejemplo	Salida ejemplo
5 Ivan Isaac Alejandra Marcos 2 Ivan Isaac 2 Ivan Isaac Marcos Alejandra Marcos	HELADO Y GALLETAS

## Constraints

- $1 \leq U \leq 10000$
- $1 \leq N \leq 10000$
- $1 \leq A \leq 60000$

# Problema B

## Poniendo Baldosas

Poner baldosas puede parecer fácil, pero en realidad hay que tener en cuenta varias cosas. Esta vez Carlos necesita ayuda para poner baldosas en su casa y, debido a su afición al ajedrez, quiere poner intercaladas baldosas blancas y negras para que el suelo siga el patrón de un tablero de ajedrez.

Tiene el plano de su casa, con las habitaciones a reformar marcadas, pero no conoce el número de baldosas de cada tipo que necesita comprar. Una cosa que tiene clara es que, a poder ser, le gustaría que hubiera más baldosas blancas que negras, ya que hacen que la casa sea más luminosa.

### Input

La entrada comenzará con una línea indicando el ancho ( $X$ ) y el alto ( $Y$ ) de la hoja en la que ha dibujado el plano de su casa.

A continuación vendrá una línea con  $H$ , el número de habitaciones marcadas en el plano. Cada una de las  $H$  líneas que vienen a continuación indicará la esquina inferior izquierda y la superior derecha de una habitación, siguiendo el siguiente formato:  $II_x II_y SD_x SD_y$ .

Las habitaciones no están superpuestas, pero si comparten alguno de sus lados están conectadas y tienen que seguir las reglas de poner baldosas como si se tratara de un único cuarto.

La entrada debe ser leída de forma estándar.

### Output

Se debe imprimir el número total de baldosas, seguido del número de baldosas blancas y de baldosas negras.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
<pre>6 4 3 4 2 5 3 0 0 1 3 2 0 3 0</pre>	<pre>14 7 7</pre>

Entrada ejemplo	Salida ejemplo
<pre>4 3 2 0 0 0 2 3 0 3 2</pre>	<pre>6 4 2</pre>

### Constraints

- $1 \leq X, Y \leq 2000$
- $1 \leq H \leq 10000$
- $0 \leq II_x \leq SD_x < X$
- $0 \leq II_y \leq SD_y < Y$

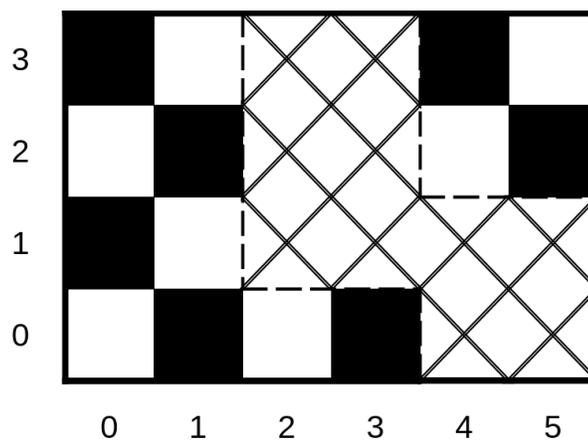


Figura que representa el primer caso de prueba.

# Problema C

## Pagando la hipoteca

Munchi es uno de nuestros vecinos del pueblo del nuevo juego de Animal Crossing, y como es nuestro vecino favorito queremos ayudarlo a pagar la hipoteca de su casa. Lo que todo el mundo hace en Animal Crossing para ganar dinero es sencillo, vender cosas a cambio de bayas, la moneda en circulación.

Tendo y Nendo, los dueños de la tienda de la isla, comprarán cualquier artículo que les ofrezcamos, eso sí, cada uno a un precio distinto. Lo más fácil en el juego es capturar mariposas (Que, la más cara, se vende a 4000 bayas) o peces (Que pueden llegar hasta las 15000 bayas de valor), para venderlos e ir consiguiendo dinero.

Hoy la captura ha sido fructífera, y llevamos los bolsillos llenos de artículos caros. Además, tenemos ahorrado algo de dinero en el Bayero Automático, que podemos también utilizar.

Sabiendo lo que vale cada artículo que compran en la tienda, los artículos que llevamos en el bolsillo, lo que tenemos ahorrado ya de dinero, y lo que le falta a Munchi por pagar de hipoteca, ¿Tenemos suficiente dinero para terminar de pagarla?

### Input

La entrada se compone de un único caso de prueba, comenzando por un primer número  $A$  de artículos que compran en la tienda, seguido de  $A$  líneas con el nombre del producto seguido de su valor  $V_i$ .

A continuación, aparecerá un número  $B$  indicando el número de artículos en nuestros bolsillos, seguido de  $B$  líneas con el nombre de los artículos. Se garantiza que conocemos el valor de todos los artículos que llevamos en los bolsillos.

Finalmente, aparecerán dos líneas, la primera con un número  $D$  de dinero ahorrado previamente, y la segunda con un número  $H$ , lo que le queda por pagar a Munchi de hipoteca.

Todos los nombres de artículos serán cadenas de caracteres consistentes únicamente de mayúsculas y minúsculas del alfabeto inglés.

La entrada debe ser leída de forma estándar.

### Output

La salida será un único número, el dinero que quedará por pagar de la hipoteca de Munchi, habiendo gastado todos los ahorros y habiendo vendido todos nuestros productos de los bolsillos.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
5 Tarantula 8000 Armario 500 Hierbajo 10 Mariposa 4000 Atun 12000 3 Mariposa Atun Tarantula 120000 150000	6000

Entrada ejemplo	Salida ejemplo
4 Crown 500000 LongBook 2000 Fossil 100000 Cherry 500 5 Crown Crown Crown Crown Fossil 97900000 100000000	0

### Constraints

- $1 \leq A \leq 100000$
- $1 \leq B \leq 100000$
- $0 \leq D < 2^{63}$
- $1 \leq H < 2^{63}$
- $1 \leq V_i < 2^{63}$

# Problema D

## Esquivando al MamáRadar

La hora se acerca. Esta tarde sale el nuevo Call Of Duty y la ciudad está preparada para ello. En esta guerra, tenemos tres bandos: Los chavales, potenciales compradores del juego, las tiendas, deseando vender el que parece que va a ser el mayor éxito del año, y las madres, que son conscientes, y no quieren que sus hijos se compren un juego tan violento.

Sabiendo la situación, cada bando ha hecho sus acuerdos y alianzas para esta guerra. Cada chico interesado en conseguir el juego va a aprovechar para comer fuera de casa, en casa de su abuelo o abuela, o algún otro amigo, y salir corriendo a por el juego en cuanto sea la hora.

Por otro lado, las madres han hecho piña, han creado un grupo de Whatsapp y se han instalado la nueva aplicación MamáRadar, capaz de detectar la localización del móvil de todos los hijos, sin que ellos lo sepan. Además, como son muy listas, se han compartido los números de móvil de sus hijos entre ellas, así que cualquier madre puede pillar a cualquier chico.

Sabiendo la descripción de la ciudad a la hora de salida del juego y la localización de todas las madres, dado el punto de inicio de cada chico, y la localización de la tienda donde tiene reservado el juego, será capaz de conseguirlo, ¿o le pillarán en el intento y le castigarán?

Podemos asegurar que un chico será castigado si desde su punto de inicio podemos llegar a cualquiera de los puntos de las madres. En caso contrario, si somos capaces de llegar a la tienda tendremos el juego, y si no, nos quedaremos sin él.

### Input

La entrada se compondrá de un único caso. En primer lugar, aparecerá un número  $N$  con el número de intersecciones de ancho y alto de la ciudad (El ancho y el alto siempre serán iguales). A continuación, aparecerán  $2N+1$  líneas, con la descripción gráfica de la ciudad. “o” indica un edificio, las líneas horizontales y verticales indican una calle cortada, y un espacio (“ ”) indica una calle transitable. Por simplicidad, la ciudad siempre será cuadrada. Se garantiza que todas las intersecciones entre calles son transitables, que en todas las esquinas entre calles habrá un edificio, y que todas las calles serán de longitud 1. También se asegura que el perímetro de la ciudad estará siempre cerrado.

Tras la descripción gráfica de la ciudad, aparecerá una línea con un número  $M$ , el número de Madres con el MamáRadar, seguido de  $M$  líneas con la localización  $F_{mi}, C_{mi}$  de cada madre.

Finalmente, aparecerá un número  $C$  de chicos queriendo comprar el juego, seguido de  $C$  líneas conteniendo cuatro números, primero la localización  $F_{ci}, C_{ci}$  del chico, y luego la localización  $F_{ti}, C_{ti}$  de la tienda.

Todas las localizaciones serán siempre coordenadas de intersecciones entre calles.

Se garantiza que, para cada caso individual, no coincidirán la coordenada de la tienda con la del punto de inicio, y ninguna de estas dos coordenadas coincidirá con ninguna de las coordenadas de las madres.

Nota: F indica Fila y C indica columna. Todas las coordenadas se dan en [F, C]

La entrada debe ser leída de forma estándar.

## Output

Para cada chico queriendo comprar el juego, se imprimirá “CASTIGADO SIN JUEGO” si serán pillados por el MamáRadar, “MAL PLANEADO” si desde el punto de inicio no se puede llegar a la tienda, y “CALL OF DUTY” si el chico es capaz de llegar a la tienda sin ser interceptado por ninguna madre.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
<pre> 5 o-o-o-o-o-o               o o-o-o-o o             o o-o o o-o             o o o-o-o o                 o o-o-o-o o                 o-o-o-o-o-o 2 0 4 1 1 4 0 0 4 0 0 0 1 2 4 4 2 4 4 4 1 4                     </pre>	<pre> CASTIGADO SIN JUEGO CASTIGADO SIN JUEGO CALL OF DUTY MAL PLANEADO                     </pre>

## Constraints

- $1 \leq N \leq 1000$
- $0 \leq M \leq 1000$
- $0 < C \leq 60000$

En la siguiente página acompaña una figura descriptiva del ejemplo.

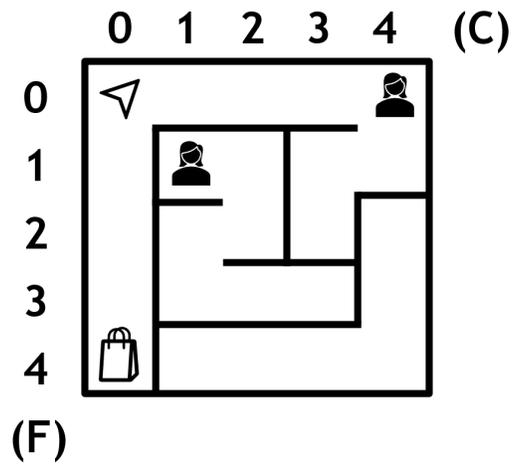


Figura que representa el primer caso de prueba del ejemplo.

El icono del triángulo indica la posición inicial de origen del primer chico.

El icono de la mujer representa la localización de cada una de las madres con MamáRadar.

El icono de la bolsa indica la localización de la tienda.

# Problema E

## El curioso caso de Benjamin Botones

La mayoría de los mortales envejecemos cada año, aumentando nuestra edad en una unidad por cada cumpleaños. Sin embargo, Benjamin Botones es un caso muy especial, aunque la mayoría de veces cumple años como el resto de los mortales, a veces pierde 5 años de golpe, ¡y a veces se multiplica por dos su edad!

Aunque Benjamin no puede controlar que va a pasar con su edad el día de su cumpleaños, te preguntas, ¿y si pudiera? ¿Cuánto años tardaría en tener la edad que el que quisiera?

### Input

La entrada comenzará con un número  $N$ , el número de casos de prueba. Para cada una de las siguientes  $N$  líneas, aparecerán dos números,  $A$  y  $B$  separados por espacios, siendo  $A$  la edad actual de Benjamín y  $B$  la edad objetivo. Nota 1: La edad debe mantenerse dentro del rango especificado por los límites del problema durante toda la ejecución del programa.

La entrada debe ser leída de forma estándar.

### Output

Debes imprimir el mínimo número de años que le costaría a Benjamin obtener la edad pedida, suponiendo que puede controlar si suma uno, resta cinco o multiplica por dos su edad en su cumpleaños. Si no es posible llegar al número pedido, debes imprimir -1.

La salida debe ser escrita de forma estándar

Entrada ejemplo	Salida ejemplo
4	0
15 15	2
20 35	2
8 11	5
20 19	

### Constraints

- $1 \leq N \leq 1000$
- $1 \leq A, B \leq 1000$