



Alfa Pool a distancia

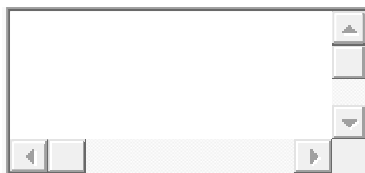
Límite de tiempo: 2000 ms Límite de memoria: 462 MB

En Alfa Pool, los jugadores juegan uno contra el otro en parejas, en el orden dictado por el sistema. Para cada juego, el ganador gana un número determinado de puntos y el perdedor no gana puntos.

Para hacer el torneo más interesante, los organizadores decidieron el siguiente conjunto de reglas:

- Los puntos otorgados se duplican por cada victoria sucesiva. La primera victoria gana 1 punto, la segunda victoria sucesiva gana 2 puntos, la tercera victoria sucesiva gana 4 puntos, y así sucesivamente.
- En caso de pérdida, se restablece la racha de victorias sucesivas. Una victoria posterior gana nuevamente 1 punto.
- Si un jugador pierde dos veces seguidas, es eliminado del torneo.

Descubre de cuántas maneras diferentes puede ganar un jugador entre AA y BB puntos, antes de ser eliminado. Por ejemplo, dejemos $A = 2$ y $B = 4$. Un jugador puede ganar entre 2 y 4 puntos de 12 maneras diferentes, donde un número KK denota una ganancia ganadora KK puntos y X denota una pérdida.



1	1	X	1	X	X			
2	X	1	X	1	X	X		
3	1	X	1	X	1	X	X	
4	X	1	X	1	X	1	X	X
5	1	X	1	X	1	X	1	X
6	X	1	X	1	X	1	X	1
7	1	X	1	2	X	X		
8	X	1	X	1	2	X	X	
9	1	2	X	X				
10	X	1	2	X	X			
11	1	2	X	1	X	X		
12	X	1	2	X	1	X	X	
13								

En todos los 12 escenarios anteriores, el jugador salió del torneo con un total de 2, 3 o 4 puntos.

Entrada estándar

Su programa debe leer de la entrada estándar. La primera línea contiene el número de consultas n que tienes que responder. Cada uno de la siguientes n líneas contiene una consulta, que consta de dos enteros no negativos separados por espacios A_i y B_i .

Salida estándar

Su programa debe imprimir exactamente a la salida estándar n líneas, cada una con exactamente un número entero: la cantidad de formas diferentes en que un jugador puede ganar entre A_i y B_i puntos antes de salir del torneo. Para cada consulta, debe imprimir el módulo de resultados $10^9 + 7$.

Restricciones y notas

Restricciones y notas

- $1 \leq n \leq 10^4$
- $0 \leq A_i \leq B_i \leq 10^6$

Entrada	Salida	Explicación
<pre>1 2 4</pre>	<pre>12</pre>	Este es el ejemplo que se muestra arriba.

<https://csacademy.com/ieeextreme13/task/ranged-alfa-pool/>



Alfa Pool

Límite de tiempo: 1000 ms Límite de memoria: 362 MB

En un torneo de billar en línea, los jugadores juegan uno contra el otro en parejas, en el orden dictado por el sistema. Para cada juego, el ganador gana un número determinado de puntos y el perdedor no gana puntos.

Para hacer el torneo más interesante, los organizadores decidieron el siguiente conjunto de reglas:

- Los puntos otorgados se duplican por cada victoria sucesiva. La primera victoria gana 1 punto, la segunda victoria sucesiva gana 2 puntos, la tercera victoria sucesiva gana 4 puntos, y así sucesivamente.
- En caso de pérdida, se restablece la racha de victorias sucesivas. Una victoria posterior gana nuevamente 1 punto.
- Si un jugador pierde dos veces seguidas, es eliminado del torneo.

Descubre de cuántas maneras diferentes puede ganar un jugador BB puntos, antes de ser eliminado. Por ejemplo, dejemos $B = 5$ $B = 5$. Un jugador puede ganar 5 puntos de 8 maneras diferentes, donde un número KK denota una ganancia ganadora KK puntos y X denota una pérdida.

1	1	X	1	X	1	X	1	X	1	X	X	
2	X	1	X	1	X	1	X	1	X	1	X	X
3	1	2	X	1	X	1	X	X				
4	X	1	2	X	1	X	1	X	X			
5	1	X	1	2	X	1	X	X				
6	X	1	X	1	2	X	1	X	X			
7	1	X	1	X	1	2	X	X				
8	X	1	X	1	X	1	2	X	X			
9												

1

En los 8 escenarios anteriores, el jugador salió del torneo con un total de 5 puntos.

Entrada estándar

Su programa debe leer de la entrada estándar. La primera línea contiene el número de consultas n que tienes que responder. Cada una de las siguientes n líneas contienen una consulta, que consta de un entero no negativo B_i .

Salida estándar

Su programa debe imprimir exactamente a la salida estándar n líneas, cada una con exactamente un número entero: la cantidad de formas diferentes en que un jugador puede ganar B_i puntos antes de salir del torneo. Para cada consulta, debe imprimir el módulo de resultados $10^9 + 7$.

Restricciones y notas

Restricciones y notas

- $1 \leq n \leq 10^4$
- $0 \leq B_i \leq 10^5$

Entrada	Salida	Explicación
1 5	8	Este es el ejemplo que se muestra arriba.

<https://csacademy.com/ieeextreme13/task/alfa-pool/>



el pastor solitario

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Bogdan el pastor solitario está de pie debajo de un **árbol** especial con n nodos, de los cuales k son azules y el resto son rojos. Bogdan realmente quiere un unicornio, está cansado de tantas ovejas y piensa que si pudiera hacer algo de magia con este árbol, aparecerá un unicornio. ¿Qué tipo de magia, preguntas? Tiene que alternar un conjunto de nodos de azul a rojo o de rojo a azul (hacer esto dos veces cancela cualquier cambio) y luego cortar un borde del árbol, dando como resultado dos árboles. Bogdan piensa que un par de árboles es un unicornio (mágico) si los árboles tienen el mismo número de nodos azules. Debe generar un conjunto de operaciones de alternancia y el borde a cortar, de modo que al final los dos árboles resultantes tengan el mismo número de nodos azules.



AHA! ¡Pensaste que esto sería una tarea fácil, pero hay una trampa! ¡Bogdan es daltónico! Entonces él no sabe la coloración inicial del árbol, excepto por el hecho de que hay K nodos azules. ¡Entonces le pide que encuentre un conjunto de operaciones de alternancia y un corte tal que para cualquier posible coloración inicial del árbol, el par resultante de dos árboles sea unicornio! En caso de que esto no sea posible, debe decírselo, porque la esperanza sin sentido es algo peligroso.

Entrada estándar

La primera línea contendrá dos enteros, N y K .

El seguimiento $N-1$ líneas contendrán cada una dos números x y y significa que hay un borde entre los nodos x y y en nuestro árbol

Salida estándar

En caso de que no haya forma de obtener el par unicornio, salida -1 , de lo contrario, salida como máximo 10^6 líneas, cada línea que denota una operación de conmutación o el corte final.

Una operación de alternancia estará representada por una línea que comienza con la cadena `flip` seguida de un número x , que denota el nodo a alternar, ambos separados por un espacio.

La operación de corte estará representada por una línea que comienza con la cadena `cut` seguida de dos números. x y y denota los puntos finales del borde a cortar, todos separados por espacios. Puede haber como máximo una operación de corte en la salida.

Restricciones y notas

- $2 \leq N \leq 10^5$
- $1 \leq K \leq N$
- Muchas gracias al verdadero Bogdan Ciobanu por ser un verdadero superhombre en la realización de este concurso.

Entrada	Salida	Explicación
<pre>3 1 1 2 2 3</pre>	<pre>voltear 1 cortar 2 1</pre>	Para cada coloración válida del árbol (con un solo nodo azul), cambie el color del primero y luego corte el $(1, 2)$ edge nos dejará con dos árboles con el mismo número de nodos azules
<pre>5 2 1 2 1 3 1 4 1 5</pre>	<pre>-1</pre>	No hay forma de aplicar las operaciones de modo que para cualquier coloración inicial válida (con solo dos nodos azules), al final terminaremos con dos árboles con el mismo número de nodos azules.

Restricciones y notas

- $2 \leq N \leq 10^5$
- $1 \leq K \leq N$

- Muchas gracias al verdadero Bogdan Ciobanu por ser un verdadero superhombre en la realización de este concurso.

Entrada	Salida	Explicación
<div>3 1</div> <div>1 2</div> <div>2 3</div>	<div>voltear 1</div> <div>cortar 2 1</div>	Para cada coloración válida del árbol (con un solo nodo azul), cambie el color del primer nodo azul. El edge $(1, 2)$ nos dejará con dos árboles con el mismo número de nodos azules.
<div>5 2</div> <div>1 2</div> <div>1 3</div> <div>1 4</div> <div>1 5</div>	<div>-1</div>	No hay forma de aplicar las operaciones de modo que para cualquier coloración inicial (con un solo nodo azul), al final terminaremos con dos árboles con el mismo número de nodos azules.

<https://csacademy.com/ieeextreme13/task/lonely-shepherd/>



Suerte-flip

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

En la isla Sierra hay un famoso juego llamado **Luck-i-flip**. Para realizar una apuesta, cada participante llena una fila de longitud n con 0 s y 1 s, que representan las dos caras de una moneda. Las apuestas son públicas para evitar que varios participantes hagan la misma apuesta.

El día del evento, el presidente de la Asociación Cultural Sierra lanza una moneda. n veces y el ganador es el participante con las suposiciones de monedas más correctas. En caso de empate, la asociación se queda con el premio gordo.

A Sara no le gusta apostar pero le interesa el hecho de que este juego no depende únicamente de la suerte. No todas las apuestas ofrecen las mismas posibilidades de ganar. En particular, ella quiere saber cuáles son las probabilidades de ganar para la peor apuesta en el libro público.

Para eso, Sara debe contar cuántos sorteos diferentes otorgarán el premio a la peor apuesta. ¿Puedes ayudarla?

Tome, por ejemplo, las tres apuestas 001 , 110 y 101 :

- para la apuesta 001 los sorteos ganadores son 000 , 001 y 011
- para la apuesta 110 son los sorteos ganadores 010 y 110
- para la apuesta, 101 el único empate ganador es 101

En este caso, la peor apuesta es 101 y el número de diferentes sorteos ganadores para la peor apuesta es 1 .

Entrada estándar

La primera línea es el número de apuestas, b en el libro mayor público, seguido de una apuesta en cada línea.

Salida estándar

La salida es el número mínimo, m , de sorteos ganadores para la peor apuesta.

Restricciones y notas

- $n \leq 22$
- $b \leq 100$
- $m \leq 20$

Entrada	Salida	Explicación
<div>2 00 11</div>	<div>1</div>	Hay dos apuestas, 00y 11. Hay cuatro posibles empates de longitud 2: 00, 01, 10, 11. Si el empate es 00, entonces la apuesta ganadora sería 00. Si el empate es 11la apuesta ganadora sería 11. Pero si el sorteo es 01o 10ambas apuestas fallaron en un lado de la moneda y pierden el premio gordo. Para cada apuesta, solo hay11 sorteo que lo lleva al premio. Entonces la respuesta es el mínimo que es11
<div>2 00 01</div>	<div>2</div>	Hay dos apuestas, 00y 01. Hay cuatro posibles empates de longitud 2: 00, 01, 10, 11. Ambos empates 00y 10serían ganados por 00. Por el contrario, ambos empatan 11y 01serían ganados por 01. Entonces ambas apuestas tienen22 sorteos ganadores. La respuesta es el mínimo que es22

Restricciones y notas

- $n \leq 22$
- $b \leq 100$
- $m \leq 20$

Entrada	Salida	Explicación
2 00 11	1	Hay dos apuestas, 00 y 11. Hay cuatro posibles empates de longitud 2: 00, 01, 10, 11. Si el empate es 00, entonces la apuesta ganadora sería 00. Si el empate es 11 la apuesta ganadora sería 11. Pero si el sorteo es 01 o 10 ambas apuestas fallaron en un lado de la moneda y pierden el premio gordo. Para cada apuesta, solo hay 1 sorteo que lo lleva al premio. Entonces la respuesta es el mínimo que es 1
2 00 01	2	Hay dos apuestas, 00 y 01. Hay cuatro posibles empates de longitud 2: 00, 01, 10, 11. Ambos empates 00 y 10 serían ganados por 00. Por el contrario, ambos empatan 11 y 01 serían ganados por 01. Entonces ambas apuestas tienen 2 sorteos ganadores. La respuesta es el mínimo que es 2

<https://csacademy.com/ieeextreme13/task/luck-i-flip/>



Googolplex

Límite de tiempo: 300 ms Límite de memoria: 256 MB

El universo Alpha es mucho más grande que nuestro universo actual y la distancia entre los diferentes planetas es del orden de unos pocos metros de googolplex. Un googolplex es un gran número igual a $10^{10^{100}}$. En notación decimal, se escribe como el dígito 1 seguido de 10^{100} ceros. La luz viaja a una velocidad de 3×10^8 metros por segundo. ¡Por lo tanto, incluso la luz tardaría mucho tiempo en recorrer un metro de googolplex! Los humanos en el universo Alpha tienen la tecnología para viajar a la velocidad de la luz, pero dado que la distancia entre los planetas todavía es muy grande. El tiempo para viajar entre diferentes planetas es mayor que la vida de los humanos normales, lo que hace que sea imposible viajar entre planetas directamente.

Afortunadamente, en el universo Alfa hay algunos agujeros de gusano que acortan las distancias al actuar como túneles que conectan la Tierra con diferentes planetas. Para viajar desde la Tierra a un planeta distante que es XX metros de googolplex de distancia, tendrías que tomar el agujero de gusano YY para acortar la distancia. En este universo, el tiempo t , en segundos, gastado en el agujero de gusano viene dado por la fórmula $t = (X^{\text{googolplex}} + T) \bmod 10^{21}$ where T es la hora del día (la hora en segundos después de las 12:00 UTC).

Por ejemplo, si un planeta es $5A5$ metros de googolplex y está usando un agujero de gusano 22 , entonces el tiempo pasado en el agujero de gusano viene dado por $t = (5^{\text{googolplex}} + T) \bmod 10^{21}$ segundos a la hora del

día TT . Genere el tiempo mínimo de permanencia en el agujero de gusano determinando la hora óptima del día TT .

Por ejemplo, a la hora del día $T = 10T = 10$, el tiempo pasado en el agujero de gusano es $(5^{\text{googolplex} + 10} \cdot 5^{\text{googolplex} + 10}) \bmod 10^{2102}$. TT puede variar de 00 a 8639986399.

Entrada estándar

La primera línea contiene un número entero. N , el número de casos de prueba.

Cada uno de los siguientes N líneas contienen dos enteros XX y YY .

Salida estándar

Salida de un número entero para cada caso de prueba que denota el tiempo mínimo de permanencia en el agujero de gusano.

Restricciones y notas

- $1 \leq N \leq 20$
- $1 \leq X \leq 10^8$
- $1 \leq Y \leq 9$
- $0 \leq T \leq 86399$

Entrada	Salida	Explicación
<div>1 2 4</div>	<div>16</div>	Hay 1 caso de prueba. El planeta distante está a 2 metros de distancia de googolplex y e. Por lo tanto, la distancia es $(2^{\text{googolplex} + T} \cdot 2^{\text{googolplex} + T}) \bmod 10^{4104}$ cuando la hora del día $T = 4T = 4$.
<div>3 5 1 100 3 3 5</div>	<div>5 0 1</div>	Hay 3 casos de prueba. Los resultados previstos son 5, 0 y 1.

Salida estándar

Salida de un número entero para cada caso de prueba que denota el tiempo mínimo de permanencia en el agujero de gusano.

Restricciones y notas

- $1 \leq N \leq 20$
- $1 \leq X \leq 10^8$
- $1 \leq Y \leq 9$
- $0 \leq T \leq 86399$

Entrada	Salida	Explicación
1 2 4	16	Hay 1 caso de prueba. El planeta distante está a 2 metros de distancia de googolplex y el agujero de gusano que solía viajar es 4. Por lo tanto, la distancia es $(2^{\text{googolplex}+T}) \bmod 10^4$. El tiempo mínimo es de 16 segundos cuando la hora del día $T = 4$.
3 5 1 100 3 3 5	5 0 1	Hay 3 casos de prueba. Los resultados previstos son 5, 0 y 1.

<https://csacademy.com/ieeextreme13/task/googolplex/>



Siempre Tyrannosauridae!

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Después de descubrir que un senador que no puede masticar y el senador W han conspirado contra Herr. ¡Tyrannosaurus, César tuvo que abandonar sus esfuerzos para impresionar a Tiranca y venir a darles un castigo prometeico! Les dio una secuencia v de n números, un número inicial K y Q actualizaciones y consultas de la siguiente forma:

- **Actualización** : cambiar el valor de un elemento
- **Consulta** : se le dan dos números l y r . Una *pandereta* es la operación de tomar una submatriz v_l, v_{l+1}, \dots, v_r y la transformación de cada uno de sus elementos v_i en $\sum_{k=l}^r v_k$ (en algunos círculos más elevadas, que llaman a esta aplicación de sumas parciales sólo a los elementos de la submatriz dado). Tienes que dar salida al valor de v_r después de tambourining la submatriz K veces. Como estos valores pueden ser bastante grandes, debe generar el resto después de dividirse por $10^9 + 7$. La consulta no es persistente (después de aplicar los resultados del tambor, la secuencia vuelve a su estado anterior a la consulta).

Entrada estándar

La primera línea contendrá los números. n, K y Q en este orden. La segunda línea contendrán n números que representan los valores de la secuencia v . El i -th de la próxima Q líneas describirán actualizaciones y consultas en el siguiente formato: la línea comenzará con el carácter u . En caso de que el carácter sea u , la línea describirá una consulta y el carácter será seguido por dos números l y r mencionado en la declaración del problema. En

caso de que el carácter sea u, la línea describirá una actualización y el carácter será seguido por dos números. $p \ y \ x$ implicando que el valor $dev_p \ p$ será cambiado con x .

Salida estándar

La salida contendrá las respuestas de las consultas, cada una escrita en orden

Restricciones y notas

- $1 \leq N, Q \leq 10^5$
- $2 \leq K \leq 8$
- Los elementos de la secuencia nunca excederán 10^9
- Para todas las consultas, $1 \leq l \leq r \leq N$
- Para todas las actualizaciones, $1 \leq p \leq N$
- Nota: el autor sabe que el título del problema no es totalmente adecuado para el planteamiento del problema, pero no pudo resistirse a hacer que el juego de palabras, también *tambourinishment* es una palabra inventada que el autor tuvo que usar después de haber sido obligado por su novia.

Entrada	Salida	Explicación
<pre> 5 3 4 3 1 4 1 5 Q 1 5 Q 2 4 En 4 6 Q 3 5 </pre>	<pre> 87 19 47 </pre>	<p>Tenemos 33 consultas y 11 actualización.</p> <p>La última consulta se realiza en la siguiente submatriz: $[4, 6, 5][4, 6, 5]$.</p> <p>Después de cada una de las 33 pasos, el subconjunto tiene el siguiente aspecto:</p> <p>$k = 1$ $[4, 10, 15] a = 1 : [4, 10, 15]$</p> <p>$k = 2$: $[4, 14, 29] a = 2 : [4, 14, 29]$</p> <p>$k = 3$: $[4, 18, 47] a = 3 : [4, 18, 47]$</p> <p>El resultado es 4747.</p>

Restricciones y notas

- $1 \leq N, Q \leq 10^5$
- $2 \leq K \leq 8$
- Los elementos de la secuencia nunca excederán 10^9
- Para todas las consultas, $1 \leq l \leq r \leq N$
- Para todas las actualizaciones, $1 \leq p \leq N$
- Nota: el autor sabe que el título del problema no es totalmente adecuado para el planteamiento del problema, pero no pudo resistirse a hacer que el juego de palabras, también *tambourinishment* es una palabra inventada que el autor tuvo que usar después de haber sido obligado por su novia.

Entrada	Salida	Explicación
5 3 4 3 1 4 1 5 Q 1 5 Q 2 4 En 4 6 Q 3 5	87 19 47	Tenemos 3 consultas y 1 actualización. La última consulta se realiza en la siguiente submatriz: $[4, 6, 5]$. Después de cada una de las 3 pasos, el subconjunto tiene el siguiente aspecto: $a = 1 : [4, 10, 15]$ $a = 2 : [4, 14, 29]$ $a = 3 : [4, 18, 47]$ El resultado es 47.

<https://csacademy.com/ieeextreme13/task/sic-semper-tyrannosaurus/>



Ave Caesar!

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Cansado de hacer cifras y brutalizar triángulos, César decidió impresionar a Tiranca, el unicornosaurus, con algo más, ¡su obsesión con los cierres algebraicos! Escéptico de su habilidad, Tiranca le da el siguiente problema:

Una *cadena válida* es:

1. Una cadena que consta de un solo carácter
2. La concatenación de 22 *cadenas válidas* AA y BB , si $BA \leq B$ lexicográficamente.

Dado $norteN$ cadenas, decida cuáles son *cadenas válidas*.

Entrada estándar

- La primera línea contiene un número entero. $norteN$, el número de cadenas dadas.
- El seguimiento $norteN$ líneas contienen una cadena en cada línea.

Salida estándar

La salida debe contener una sola línea con $norteN$ caracteres, $elyoi$ -ésimo $1si$ eli $^$
 $\{th\}$ i th cadena a partir de la entrada es válida, 0 en caso contrario.

Restricciones y notas

- La longitud total de las cadenas será menor que 10^6
- Todas las cadenas contienen solo caracteres en mayúscula

Entrada

Salida

5
BBBABB
Acerca de
El padre
BBABAB
ABBBBA

01000

Restricciones y notas

- La longitud total de las cadenas será menor que 10^6
- Todas las cadenas contienen solo caracteres en mayúscula

Entrada

Salida

5
BBBABB
Acerca de
El padre
BBABAB
ABBBBA

01000

<https://csacademy.com/ieeextreme13/task/ave-caesar/>



Protegiendo la colmena

Límite de tiempo: 500 ms Límite de memoria: 256 MB

El conejo Alfa tiene un amigo abeja que le proporciona miel, llamada Maya. Maya es la reina de las abejas y pertenece a la colmena. La colmena es el lugar donde las abejas obreras producen y conservan miel.

La colmena está hecha de celdas, cada celda tiene la forma de un hexágono donde cada lado de este hexágono mide 11 cm. La colmena tiene N filas, donde cada fila impar tiene M células y cada fila par tiene $M - 1$ células. Cada célula puede estar activa o inactiva. Una célula activa es donde las abejas pueden producir miel, mientras que una célula inactiva tiene algún problema y las abejas no pueden producir miel allí.

La colmena se compone de muchas secciones, donde una sección está hecha de un conjunto de celdas conectadas (dos celdas están conectadas si están adyacentes entre sí a través de uno de sus 6 lados). Maya es capaz de:

1. Cree un perímetro con una sustancia especial para cada sección para protegerlo de otros insectos.
2. Activa una célula inactiva.

De vez en cuando, Maya quiere saber, dada una celda específica, cuál es el perímetro de la sección a la que pertenece. ¿Puedes ayudar a la reina?

Entrada estándar

Entrada	Salida	Explicación
3 3 0 1 0 0 1 0 0 0 5 a 1 3	12 0 6	

Entrada	Salida	Explicación
---------	--------	-------------

k 2 2

k 2 1

a 3 1

k 3 1

5 5

0 1 0 0 0

0 1 0 0

0 0 0 1 1

1 0 1 0

0 0 0 1 1

8

a 2 1

k 2 1

k 1 2

k 3 4

12

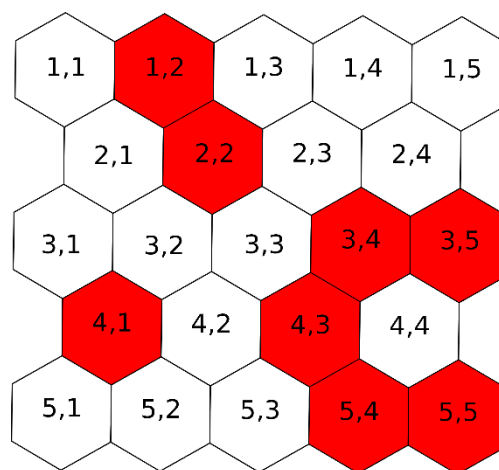
12

22

6

22

34



Entrada	Salida	Explicación
<pre> k 4 1 k 5 5 a 3 3 k 3 3 </pre>		

Salida estándar

El resultado consiste en un número de líneas igual al número de `k` consultas, donde cada línea representa la respuesta a cada consulta donde la reina quiere saber el perímetro de la sección.

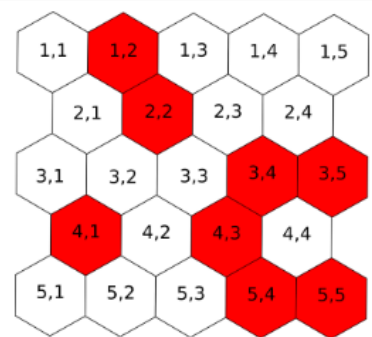
Restricciones y notas

- $2 \leq N \leq 100$
- $2 \leq M \leq 100$
- $1 \leq Q \leq 10^5$
- Las consultas consistirán solo en celdas válidas.

Entrada	Salida	Explicación
<pre> 3 3 0 1 0 0 1 0 0 0 5 a 1 3 k 2 2 k 2 1 a 3 1 k 3 1 </pre>	<pre> 12 0 6 </pre>	

```
5 5
0 1 0 0 0
0 1 0 0
0 0 0 1 1
1 0 1 0
0 0 0 1 1
8
a 2 1
k 2 1
k 1 2
k 3 4
k 4 1
k 5 5
a 3 3
k 3 3
```

```
12
12
22
6
22
34
```





Alquiler Bearcity

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Vangelis the Bear y su amigo Charlie quieren abrir su propia tienda en Bearcity. Necesitan decidir qué lugar alquilarán para comenzar su tienda. Debido a que constantemente necesitarán comprar suministros para la tienda, quieren encontrar un lugar cerca de un almacén.

Bearcity solía ser una ciudad pequeña, por lo que todas las calles son de una sola dirección y siempre hay un atasco de tráfico pesado. El nuevo alcalde, Lisa el oso, al ver que la ciudad se está expandiendo, quiere convertir algunas calles de sentido único en calles de doble sentido. Algunos costos de conversión están asociados con cada calle. Lisa quiere hacer algunas conversiones de manera que el costo total de conversión se minimice, teniendo en cuenta que todos los puntos de almacén deben estar conectados por caminos de dos vías.

Vangelis y Charlie, después de conocer esta idea, decidieron alquilar un lugar en una calle de doble sentido, para que los clientes puedan entrar y salir fácilmente de la tienda. Como puede haber más de un plan de conversión, ayude a Vangelis y Charlie a decidir cuántas calles son parte de todos los posibles planes de conversión.

Entrada estándar

La primera línea de la entrada contiene dos enteros positivos. El primero indica el número n de almacenes considerados en el plan de conversión y el segundo el número m de calles. El siguiente m es un trillizo que indica, para cada calle, sus puntos finales u, v , seguido de su costo de conversión.

Salida estándar

Genere un valor único que represente el número de calles que forman parte de todos los posibles planes de conversión.

Restricciones y notas

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 2 * 10^5$
- El costo de la conversión c_s para cada calle s : $1 \leq c_s \leq 10^4$
- Hay como máximo una carretera que conecta dos almacenes.
- Se garantiza que existe al menos un plan de conversión.

Entrada	Salida	Explicación
---------	--------	-------------

3 3
1 2 2
3 1 2
2 3 1

1

221123

Hay 2 posibles planes de conversión de costo mínimo: $(1, 3)$, $(2, 3)(1, 3)$, $(2, 3)$ o $(1, 2)$, $(2, 3)(1, 2)$, $(2, 3)$. Solo la calle $(2, 3)(2, 3)$ ocurre en ambos.

6 9
1 2 2
1 3 1
1 6 2
2 4 2
2 5 2
4 3 2
3 5 2
4 5 4
5 6 4

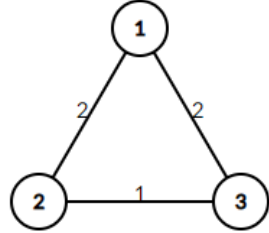
2

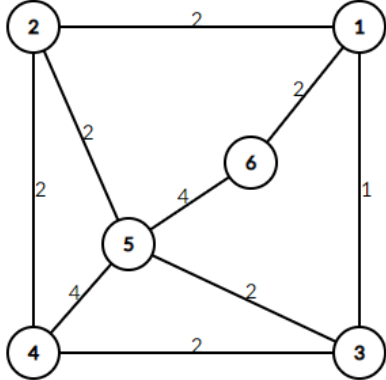
2122222442123456

Solo los caminos $(1, 3)(1, 3)$ y $(1, 6)(1, 6)$ son parte de todos los posibles planes de conversión.

Restricciones y notas

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 2 * 10^5$
- El costo de la conversión c_s para cada calles : $1 \leq c_s \leq 10^4$
- Hay como máximo una carretera que conecta dos almacenes.
- Se garantiza que existe al menos un plan de conversión.

Entrada	Salida	Explicación
<pre> 3 3 1 2 2 3 1 2 2 3 1 </pre>	1	 <p>Hay 2 posibles planes de conversión de costo mínimo: $(1, 3)$, $(2, 3)$ o $(1, 2)$, $(2, 3)$. Solo la calle $(2, 3)$ ocurre en ambos.</p>

<pre> 6 9 1 2 2 1 3 1 1 6 2 2 4 2 2 5 2 4 3 2 3 5 2 4 5 4 5 6 4 </pre>	2	 <p>Solo los caminos $(1, 3)$ y $(1, 6)$ son parte de todos los posibles planes de conversión.</p>
--	---	---



Yin Yang

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Encontrar el equilibrio perfecto es algo buscado por muchas personas y de muchas maneras ... a veces incluso con cuerdas. Llamaremos a una cadena desequilibrada si tiene una longitud uniforme y sus dos mitades son iguales. Encuentra una cadena de longitud n , que consta solo de caracteres y y Y , de modo que tiene la menor cantidad posible de subcadenas desequilibradas distintas.

Su puntaje por prueba se computará como $\{(1 + \frac{1}{10})^K\}^{1+101-K}$, en donde K es $\lfloor F - 0 \rfloor$ y $F - 0,00$ es el número óptimo de subcadenas desequilibradas distintas y F es el número de subcadenas desequilibradas distintas que ha obtenido.

Perfectamente equilibrado como deberían ser todas las cosas.

Entrada estándar

La primera línea contiene un número entero. n .

Salida estándar

Imprime la respuesta en la primera línea.

Restricciones y notas

- $1 \leq n \leq 300$

- Por *ye* nos referimos al [número de Euler](#), que es ≈ 2.718282

Entrada

Salida

4

aaaa

Salida estándar

Imprime la respuesta en la primera línea.

Restricciones y notas

- $1 \leq N \leq 300$
- Por *e* nos referimos al [número de Euler](#), que es ≈ 2.718282

Entrada

Salida

4

aaaa

<https://csacademy.com/ieeextreme13/task/yin-yang/>



Cadena infinita

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

A Ivan le gusta jugar con cuerdas. Hoy decidió hacer la cuerda más grande de la historia.

Usando el primero BB caracteres del alfabeto latino, quiere escribir una cadena que primero consista en todas las palabras de longitud 11 , en orden lexicográfico, luego todas las palabras de longitud 22 , en orden lexicográfico, etc.

Por ejemplo, usando el primer 44 caracteres (a, b, c, d), la cadena será la siguiente (sin espacios):

abcd aa ab ac ad ba bb bc bd ca cb cc cd da db dc dd aaa aab aac aad aba ...

Como la cadena es infinita y Ivan tiene que ir a la universidad por la mañana, le interesa saber qué personaje está en el índice. XX .

Entrada estándar

La primera línea contiene un entero, TT , el número de consultas.

El siguiente T Las líneas T contienen dos enteros, BB y XX , que corresponde al número de caracteres utilizados en el alfabeto y el 0 índice basado en 0 del personaje que Ivan quiere saber.

Salida estándar

Salida T Líneas T . Los $y0$ La línea tiene el carácter para el $y0i$ th consulta.

Restricciones y notas

- $1 \leq T \leq 100$
- $1 \leq B \leq 26$
- $0 \leq X \leq 10^{18}$

Entrada	Salida	Explicación
6	a	Para casos donde $B=2$, el 3 ^{er} 2 ^º caracter es el siguiente:
4 5	re	abaaabbbabaaaaababaabbbbaababbbab b baaaaaabaabaaabbabaaabab ...
4 3	b	Para casos donde $B=4$, el 3 ^{er} 5 th y 2 ⁷ 7 caracteres son los siguientes:
2 32	y	abc d un un abacadbabbbcbdcacbcc d dadbdcdaaaaaabaacaadaba ...
26 24	a	Para casos donde $B=26$, el 2 ⁴ 4 th y 5 ⁰ 0 th caracteres son los siguientes:
26 50	re	abcdefghijklmnopqrstuvwxyz y zaaabacadaefagahaiajakal a manaoapaqar ...
4 27		Tenga en cuenta que todos los índices son Basado en 0.

Restricciones y notas

- $1 \leq T \leq 100$
- $1 \leq B \leq 26$
- $0 \leq X \leq 10^{18}$

Entrada	Salida	Explicación
6	a	Para casos donde $B = 2$, el 3 ^{er} 2 ^º caracter es el siguiente:
4 5	re	abaaabbbabaaaaababaabbbbaababbbab b
4 3	b	baaaaaaabaabababaaabab ...
2 32	y	Para casos donde $B = 4$, el 3 ^{er} 5 th y 2 ⁷ 7 caracteres son los siguientes:
26 24	a	abc d un un abacadbabbbcbdcacbcc d
26 50	re	dadbdcdaaaaaabaacaadaba ...
4 27		Para casos donde $B = 26$, el 2 ⁴ 4 th y 5 ⁰ 0 th caracteres son los siguientes:
		abcdefghijklmnopqrstuvwxyz y
		zaaabacadaefagahaiajakal a manaoapaqar ...
		Tenga en cuenta que todos los índices son Basado en 0.

<https://csacademy.com/ieeextreme13/task/infinite-string/>



Juego de concentración

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Interactivo

Concentration es un juego de cartas en el que se colocan un conjunto de cartas boca abajo sobre una superficie y se vuelcan dos cartas boca arriba en cada turno. El objetivo del juego es entregar pares de cartas iguales.

Habrán $2N$ cartas de N diferentes tipos etiquetados con enteros de 1 a N .

Interacción

Deberías leer un entero N .

Un movimiento consiste en elegir dos índices de cartas distintos i_1 y i_2 y se las has revelado. Para hacer un movimiento, imprime una línea con ambos índices separados por un espacio entre ellos (¡y no olvide enjuagar la salida!). Si estas tarjetas tienen el mismo valor escrito, el interactor responderá `MATCH` y eliminará las tarjetas de la superficie. De lo contrario, responderá con dos enteros, el primero representa el valor de i_1 y el segundo representa el valor de i_2 .

Si en algún momento el interactor decide que ha realizado un movimiento no válido, se imprimirá `-1` y saldrá. Debe leer este valor y salir de inmediato, de lo contrario podría obtener un veredicto que no refleje adecuadamente su movimiento no válido.

Debe imprimir `-1` después de haber decidido no realizar más movimientos.

Interacción	Explicación
5	El orden inicial de las tarjetas es [3, 4, 3, 4, 2, 5, 5, 1, 2][3,4,3,4,1,2,5,5,2]
1 2	
3 4	
2 4	
PARTIDO	
1 3	
PARTIDO	
9 10	
1 2	
5 9	
PARTIDO	
8 7	
PARTIDO	
10 6	
PARTIDO	
-1	

Restricciones y notas

- $1 \leq N \leq 10^3$
- Estás permitido a lo sumo 2 movimientos N
- No puede solicitar hacer un movimiento con una tarjeta que ya ha sido retirada de la superficie
- Los índices de las tarjetas **NO** cambian después de que otras tarjetas se retiran de la superficie

Interacción	Explicación
5	El orden inicial de las tarjetas es [3, 4, 3, 4, 1, 2, 5, 5, 1, 2].
	1 2
3 4	
	2 4
PARTIDO	
	1 3
PARTIDO	
	9 10
1 2	
	5 9
PARTIDO	
	8 7
PARTIDO	
	10 6
PARTIDO	
	-1

<https://csacademy.com/ieeextreme13/task/concentration-game/>

Restricciones y notas

- $1 \leq N \leq 10^3$
- Estás permitido a lo sumo 2N2 movimientos N

- No puede solicitar hacer un movimiento con una tarjeta que ya ha sido retirada de la superficie
- Los índices de las tarjetas **NO** cambian después de que otras tarjetas se retiran de la superficie



Monokeros

Límite de tiempo: 2500 ms Límite de memoria: 256 MB

El verdadero tirano, señor. W le ha dado a Tiranca un nuevo problema: se le da un árbol de búsqueda binario inicialmente vacío y una secuencia de números (x_1, x_2, \dots, x_N) (X_1, x_2, \dots, X_N) . Un árbol de búsqueda binario es un **árbol binario**, que almacena un valor en cada nodo y respeta las siguientes reglas:

- el valor almacenado en su hijo izquierdo es menor o igual al valor del nodo
- el valor almacenado en su hijo derecho es estrictamente mayor que el valor del nodo

Su tarea es insertar estos números en el árbol de búsqueda binario (en el orden dado) y generar la profundidad (distancia de borde desde el nodo a la raíz) del nodo recién agregado después de cada inserción. Una inserción es así:



1
2
3

4
5
6
7
8
9
10
11
12

```
insert_value ( current_node , new_value ) :  
    if new_value <= value ( current_node ) :  
        Si la izquierda niño de current_node existe :  
            insert_value ( left_child ( current_node ) , new_value )  
        más :  
            // crea un nuevo nodo con new_value y colócalo como hijo izquierdo  
            de current_node  
        más :  
            si el derecho del niño de current_node existe :  
                insert_value ( right_child ( current_node ) , new_value )  
            más :  
                // crea un nuevo nodo con new_value y colócalo como el hijo derecho  
                de current_node
```

```
1 insert_value ( current_node , new_value ) :  
2     if new_value <= value ( current_node ) :  
3         Si la izquierda niño de current_node existe :  
4             insert_value ( left_child ( current_node ) , new_value )  
5         más :  
6             // crea un nuevo nodo con new_value y colócalo como hijo izquierdo de current_node  
7     más :  
8         si el derecho del niño de current_node existe :  
9             insert_value ( right_child ( current_node ) , new_value )  
10        más :  
11            // crea un nuevo nodo con new_value y colócalo como el hijo derecho de current_node  
12
```

Entrada estándar

La primera línea contendrá n , el número de elementos en la secuencia.

La segunda línea contendrá n números: x_1, x_2, \dots, x_n .

Salida estándar

La primera línea contendrá n números, correspondientes a las profundidades de los nodos agregados.

Restricciones y notas

- $1 \leq n \leq 2 \cdot 10^5$
- $-10^9 \leq x_i \leq 10^9$

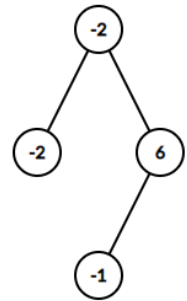
Entrada

4
-2 -2 6 -1

Salida

1 2 2 3

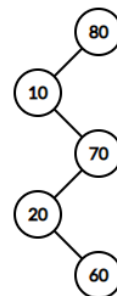
Explicación



5

80 10 70 20 60

1 2 3 4 5



<https://csacademy.com/ieeextreme13/task/monokeros/statement/>



Batchman y GCD

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

¡Las calles de Gotham han sido envenenadas por el crimen y los problemas de implementación nuevamente! Pero tenga la seguridad, nuestro héroe, Batchman ha venido a salvar el día. Hoy, tiene que enfrentarse al terrible monstruo de análisis XML al contrarrestarlo con un problema de teoría de números. El problema es que mientras lucha contra la terrible criatura, no puede pensar en una solución eficiente para el problema, ¡así que pide tu ayuda!

El problema es el siguiente: se le da una matriz $A = (a_1, a_2, \dots, a_N)$ y un número K . Considere todas las subsecuencias de A de tamaño como máximo K . Para cada subsecuencia, calcule el MCD de los números en el subconjunto. ¿Cuántos MCD distintos puedes obtener?

Entrada estándar

La primera línea contendrá los números N y K .

La segunda línea contendrá N enteros correspondientes a los elementos de A .

Salida estándar

La salida contendrá un solo número correspondiente a la respuesta del problema.

Restricciones y notas

- $1 \leq N \leq 10^5$
- $1 \leq K \leq N$
- $1 \leq a_i \leq 10^5$

Entrada	Salida	Explicación
<div>5 5</div> <div>4 8 12 16 18</div>	<div>7</div>	<p>Los únicos valores de GCD que se pueden obtener son:</p> <p>$2 = \gcd\{4, 18\}$,</p> <p>$4 = \gcd\{4\}$,</p> <p>$6 = \gcd\{12, 18\}$,</p> <p>$8 = \gcd\{8, 16\}$,</p> <p>$12 = \gcd\{12\}$,</p> <p>$16 = \gcd\{16\}$,</p> <p>$18 = \gcd\{18\}$</p>

Salida estándar

La salida contendrá un solo número correspondiente a la respuesta del problema.

Restricciones y notas

- $1 \leq N \leq 10^5$
- $1 \leq K \leq N$
- $1 \leq a_i \leq 10^5$

Entrada	Salida	Explicación
<div>5 5</div> <div>4 8 12 16 18</div>	<div>7</div>	<p>Los únicos valores de GCD que se pueden obtener son:</p> <p>$2 = \gcd\{4, 18\}$,</p> <p>$4 = \gcd\{4\}$,</p> <p>$6 = \gcd\{12, 18\}$,</p> <p>$8 = \gcd\{8, 16\}$,</p> <p>$12 = \gcd\{12\}$,</p> <p>$16 = \gcd\{16\}$,</p> <p>$18 = \gcd\{18\}$</p>

<https://csacademy.com/ieeextreme13/task/batchman-and-gcd/>



Factor de impacto

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

IEEE Xplore

¡Saludos cordiales a todos los participantes de IEEEExtreme del equipo API de Xplore!

Como parte de la Competencia IEEEExtreme, se le encargará un desafío de programación para determinar el Factor de Impacto de una revista académica. El archivo de contenido de entrada de este ejercicio es estático (no cambia) para permitir resultados consistentes durante esta competencia.

Para una búsqueda de base de datos dinámica completa, IEEE Xplore API está disponible para sus necesidades de investigación de IEEE. Xplore API proporciona metadatos en trabajos académicos de 4.9 mm y ahora ofrece contenido de texto completo en 50k artículos de 'acceso abierto'. Xplore API hará que sus necesidades de investigación sean rápidas y fáciles. El Portal API de Xplore es compatible con PHP, Python y Java, además de proporcionar resultados en formatos Json y XML. Muchos casos de uso de API se enumeran en el Portal de API.

El registro de la API Xplore es gratuito. Para obtener más información sobre IEEE Xplore API, visite developer.ieee.org/ y regístrese hoy mismo para obtener una clave API.

Desafío

El factor de impacto (IF) de una revista académica es un índice cuantitativo que refleja el número promedio anual de citas a artículos recientes publicados en esa revista. En cualquier

año dado, el factor de impacto de una revista es el número de citas, recibidas en ese año, de artículos publicados en esa revista durante los dos años anteriores, dividido por el número total de **artículos citables** publicados en esa revista durante los dos anteriores. años:

$$\mathrm{IF}_y = \frac{\mathrm{Citations}_{y-1} + \mathrm{Citations}_{y-2}}{\mathrm{Publications}_{y-1} + \mathrm{Publications}_{y-2}}$$

Entrada estándar

La primera línea es el número de registros a procesar. La segunda línea son las publicaciones que se incluirán en el informe. El número de **publicación** es la clave única para identificar la publicación. El **publicationTitle** es el título de la publicación. La matriz **articleCounts** contiene el **año** y **articleCount** para ese año y publicación.

Las siguientes líneas son paquetes JSON que contienen citas para publicaciones aleatorias. Los **paperCitations** contiene un conjunto, llamado **IEEE**, con información acerca de los artículos citados. El número de **publicación** en la matriz es el número de publicación de la publicación en la que aparece el artículo citado. El **año** es el año en que se publicó el artículo.

Salida estándar

Liste el título de la publicación y su Factor de Impacto. Ordenar por factor de impacto descendente. Formatee la salida con un nombre de diario por línea y con el nombre y la puntuación (con dos decimales) separados por dos puntos. Ejemplo:Letters on IEEEExtreme: 1.78

En caso de que varias publicaciones tengan el mismo factor de impacto, imprímalas en orden alfanumérico.

Restricciones y notas

- El número máximo de registros es 1001.

Entrada	Salida	Explicación
<pre>6 {"publicaciones": [{"publishingTitle": "Letters on IEEEExtreme", "publishingNumber": "1", "articleCounts": [{"year": "2017", "articleCount": "3"}, {"año ":" 2018 ", " articleCount ":" 6 "}]}, {" publishingTitle ":" Diario de programación de 24 horas ", " publishingNumber ":" 2 ", " articleCounts ": [{"year ":" 2017 ", " articleCount ":" 1 "}, {" año ":" 2018 ", " articleCount ":" 4 "}]}} {"publicador": "IEEE", "título": "Título de publicación 1", "tipo de contenido": "publicaciones periódicas", "ieeeCitati</pre>	<div>Cartas en I EEExtreme: 1.78</div> <div>Diario de p rogramación de 24 horas : 1.20</div>	Tomando la segunda Impacto para publica programación de 24 mayor factor de impa

Entrada

Salida

Explicación

```
onCount": "4", "número de publicación": "15", "paperCitation
s": {"ieee ": [{" orden ":" 1 ", " número de artículo ":" 41
", " número de publicación ":" 4 ", " año ":" 2018 ", " título
":" Artículo 41 "}, {" orden " : "2", "número de artículo":
"109", "número de publicación": "3", "año": "2015", "título"
: "Artículo 109"}, {"orden": "3", "número de artículo" ":" 1
35 ", " publicaciónNumber ":" 1 ", " año ":" 2018 ", "título ":"
" Artículo 135 "}, {" orden ":" 4 ", " número de artículo ":"
97 ", " número de publicación ":" 1 ", " año ":" 2016 ", " títu
lo ":" Artículo 97 " }, {"orden": "5", "número de artículo":
"31", "número de publicación": "1", "año": "2015", "título":
"Artículo 31"}, {"orden": "6", "número de artículo": "89", "
número de publicación": "4", "año": "2018", "título": "Artíc
ulo 89"}, {"orden": "7", "número de artículo" : "9", "public
aciónNumber": "4", "año": "2018", "título": "Artículo 9"}, {
"orden ":" 8 ", " número de artículo ":" 26 ", " número de pub
licación ":" 1 ", " año ":" 2015 ", " título ":" Artículo 26 "
}, {" orden ":" 9 ", "articleNumber": "117", "publishingNumb
er": "1", "year": "2015", "title": "Article 117"}, {"order":
"10", "articleNumber": "35" , "publishingNumber": "2", "year
": "2019", "title": "Article 35"}, {"order": "11", "articleN
umber": "9", "publishingNumber": "2 ", " año ":" 2016 ", " tít
ulo ":" Artículo 9 "}, {" orden ":" 12 ", "articleNumber ":"
61 ", " publishingNumber ":" 1 ", " year ":" 2017 ", " title ":"
" Article 61 "}, {" order ":" 13 ", " articleNumber ":" 75 ",
"publishingNumber": "3", "year": "2019", "title": "Article 7
5"}, {"order": "14", "articleNumber": "25", "publishingNumbe
r": "2" , "año": "2019", "título": "Artículo 25"}, {"orden":
"15", "número de artículo": "56", "número de publicación": "
3", "año": "2016 ", " title ":" Artículo 56 "}}}}2017", "tít
ulo": "Artículo 61"}, {"orden": "13", "número de artículo":
"75", "número de publicación": "3", "año": "2019", "título"
: "Artículo 75"}, {"orden": "14", "número de artículo": "25"
, "número de publicación": "2", "año": "2019", "título": "Ar
tículo 25"}, {"orden": "15", "número de artículo": "56", "nú
mero de publicación": "3", "año": "2016", "título": "Artícu
lo 56"}}}}2017", "título": "Artículo 61"}, {"orden": "13", "
número de artículo": "75", "número de publicación": "3", "añ
o": "2019", "título" : "Artículo 75"}, {"orden": "14", "núme
ro de artículo": "25", "número de publicación": "2", "año":
"2019", "título": "Artículo 25"}, {"orden": "15", "número de
artículo": "56", "número de publicación": "3", "año": "2016"
, "título": "Artículo 56"}}}}año": "2019", "título": "Artíc
ulo 75"}, {"orden": "14", "número de artículo": "25", "númer
o de publicación": "2", "año": "2019" , "título": "Artículo
25"}, {"orden": "15", "número de artículo": "56", "número de
publicación": "3", "año": "2016", "título": "Artículo 56 "}}
}}año": "2019", "título": "Artículo 75"}, {"orden": "14", "
número de artículo": "25", "número de publicación": "2", "añ
o": "2019" , "título": "Artículo 25"}, {"orden": "15", "núme
```

Entrada

Salida

Explicación

ro de artículo": "56", "número de publicación": "3", "año": "2016", "título": "Artículo 56 "}}}}Número de publicación ": " 3 ", " año ":" 2016 ", " título ":" Artículo 56 "}}}}Número de publicación ":" 3 ", " año ":" 2016 ", " título ":" Artícul o 56 "}}}}

{"editor": "IEEE", "título": "Título de publicación 2", "tip o de contenido": "publicaciones periódicas", "ieeeCitationCo unt": "2", "número de publicación": "28", "paperCitations": {"ieee ": [{"orden ":" 1 ", " número de artículo ":" 14 ", " número de publicación ":" 1 ", " año ":" 2018 ", " título ":" Artículo 14 "}, {"orden " : "2", "número de artículo": "105 ", "número de publicación": "2", "año": "2017", "título": "A rtículo 105"}], {"orden": "3", "número de artículo ":" 130 ", " publicaciónNumber ":" 4 ", " año ":" 2017 ", "título ":" Art ículo 130 "}, {"orden ":" 4 ", " número de artículo ":" 61 " , " número de publicación ":" 4 ", " año ":" 2019 ", " título " : " Artículo 61 " }, {"orden": "5", "número de artículo": "11 5", "número de publicación": "3", "año": "2015", "título": " Artículo 115"}, {"orden": "6", "número de artículo": "84", " número de publicación": "4", "año": "2015", "título": "Artíc ulo 84"}, {"orden": "7", "número de artículo" : "57", "publi caciónNumber": "2", "año": "2018", "título": "Artículo 57"}, {"orden ":" 8 ", " número de artículo ":" 96 ", " número de pu blicación ":" 4 ", " año ":" 2019 ", " título ":" Artículo 96 "}, {"orden ":" 9 ", "articleNumber": "9", "publishingNumbe r": "1", "year": "2017", "title": "Article 9"}, {"orden": "1 0", "articleNumber": "67", "publishingNumber": "4", "year": "2018", "title": "Article 67"}, {"orden": "11", "articleNumb er": "114", "publishingNumber": "2 ", " año ":" 2016 ", " títu lo ":" Artículo 114 "}, {"orden ":" 12 ", "articleNumber ":" 59 ", " publishingNumber ":" 1 ", " year ":" 2015 ", " title " : " Article 59 "}, {"orden ":" 13 ", " articleNumber ":" 118 " , "publishingNumber": "3", "year": "2015", "title": "Article 118"}, {"orden": "14", "articleNumber": "61", "publishingNum ber": "4" , "año": "2016", "título": "Artículo 61"}, {"orden ": "15", "número de artículo": "79", "número de publicación" : "4", "año": "2017 ", " title ":" Artículo 79 "}, {"orden " : " 16 ", " articleNumber ":" 83 ", "publishingNumber ":" 3 ", " year ":" 2018 ", " title ":" Article 83 "}, {"orden ":" 17 " , " articleNumber ":" 24 ", " publishingNumber ":" 4 ", "año": "2016", "título": "Artículo 24"}, {"orden": "18", "número de artículo": "48", "número de publicación": "3", "año": "2015" , "título": "Artículo 48"}, {"orden": "19", "número de artíc ulo": "11", "número de publicación": "4", "año": "2019", "tí tulo": "Artículo 11 "}, {"orden ":" 20 ", " número de artícu lo ":" 125 ", " número de publicación ":" 2 ", "año ":" 2019 " , " título ":" Artículo 125 "}, {"orden ":" 21 ", " número de artículo ":" 64 ", " número de publicación ":" 1 ", " año ":" 2016 ", "title": "Artículo 64"}, {"orden": "22", "número de artículo": "25", "número de publicación": "4", "año": "2016" }

Entrada

Salida

Explicación

```
16", "title": "Article 22"}, {"order": "28", "articleNumber":
: "113", "publishingNumber": "1 ", "año ":" 2015 ", " título
":" Artículo 113 "}}}}orden ":" 25 ", " número de artículo ":"
" 108 ", " número de publicación ":" 1 ", " año ":" 2018 ", " t
ítulo ":" Artículo 108 "}, {" orden ":" 26 ", "articleNumber
": "82", "publishingNumber": "2", "year": "2017", "title": "
Article 82"}, {"order": "27", "articleNumber": "22" , "publi
shingNumber": "2", "year": "2016", "title": "Article 22"}, {
"order": "28", "articleNumber": "113", "publishingNumber": "
1 ", " año ":" 2015 ", " título ":" Artículo 113 "}}}}"108", "
publishingNumber": "1", "year": "2018", "title": "Article 10
8"}, {"order": "26", "articleNumber": "82", "publishingNumbe
r" : "2", "año": "2017", "título": "Artículo 82"}, {"orden":
"27", "número de artículo": "22", "número de publicación": "
2", "año ":" 2016 ", " título ":" Artículo 22 "}, {" orden ":"
" 28 ", " número de artículo ":" 113 ", " número de publicació
n ":" 1 ", " año ":" 2015 ", " título ":" Artículo 113 "}}}}"1
08", "publishingNumber": "1", "year": "2018", "title": "Arti
cle 108"}, {"order": "26", "articleNumber": "82", "publishin
gNumber" : "2", "año": "2017", "título": "Artículo 82"}, {"o
rden": "27", "número de artículo": "22", "número de publicac
ión": "2", "año ":" 2016 ", " título ":" Artículo 22 "}, {" o
rden ":" 28 ", " número de artículo ":" 113 ", " número de pub
licación ":" 1 ", " año ":" 2015 ", " título ":" Artículo 113
"}}}}título ":" Artículo 108 "}, {" orden ":" 26 ", " número
de artículo ":" 82 ", " número de publicación ":" 2 ", " año "
:" 2017 ", " título ":" Artículo 82 " }, {"orden": "27", "núm
ero de artículo": "22", "número de publicación": "2", "año":
"2016", "título": "Artículo 22"}, {"orden": "28", "número de
artículo": "113", "número de publicación": "1", "año": "2015
", "título": "Artículo 113"}}}}título ":" Artículo 108 "}, {
" orden ":" 26 ", " número de artículo ":" 82 ", " número de p
ublicación ":" 2 ", " año ":" 2017 ", " título ":" Artículo 82
" }, {"orden": "27", "número de artículo": "22", "número de
publicación": "2", "año": "2016", "título": "Artículo 22"},
{"orden": "28", "número de artículo": "113", "número de publ
icación": "1", "año": "2015", "título": "Artículo 113"}}}}tí
tulo ":" Artículo 82 "}, {" orden ":" 27 ", " número de artíc
ulo ":" 22 ", " número de publicación ":" 2 ", " año ":" 2016
", " título ":" Artículo 22 " }, {"orden": "28", "número de a
rtículo": "113", "número de publicación": "1", "año": "2015"
, "título": "Artículo 113"}}}}título ":" Artículo 82 "}, {"
orden ":" 27 ", " número de artículo ":" 22 ", " número de pub
licación ":" 2 ", " año ":" 2016 ", " título ":" Artículo 22 "
}, {"orden": "28", "número de artículo": "113", "número de p
ublicación": "1", "año": "2015", "título": "Artículo 113"}}
}"Artículo 113"}}}"Artículo 113"}}}
```

```
{"editor": "IEEE", "título": "Título de publicación 3", "tip
o de contenido": "publicaciones periódicas", "ieeeCitationCo
unt": "4", "número de publicación": "23", "paperCitations":
```

Entrada

Salida

Explicación

```
{
  "ieee ": [
    {
      "orden ": " 1 ",
      "número de artículo ": " 75 ",
      "número de publicación ": " 2 ",
      "año ": " 2016 ",
      "título ": " Artículo 75 "
    },
    {
      "orden " : "2",
      "número de artículo": "88",
      "número de publicación": "2",
      "año": "2018",
      "título": "Ar tículo 88"
    },
    {
      "orden": "3",
      "número de artículo ": " 12 ",
      "p ublicaciónNumber ": " 2 ",
      "año ": " 2016 ",
      "título ": "Artíc ulo 12"
    },
    {
      "orden": "4",
      "número de artículo": "90",
      "número de publicación": "2",
      "año": "2015",
      "título": "Artículo 90"
    },
    {
      "orden": "5",
      "número de artículo": "126",
      "número de pu blicación": "1",
      "año": "2017",
      "título": "Artículo 126"
    },
    {
      "orden": "6 ",
      "número de artículo ": " 12 ",
      "número de publ icación ": " 1 ",
      "año ": " 2018 ",
      "título ": " Artículo 12 "
    },
    {
      "orden ": " 7 ",
      "número de artículo ": " 7 ",
      "número de publicación ": " 1 ",
      "año ": " 2017 ",
      "título ": " Artículo 7 "
    },
    {
      "orden ": "8 ",
      "número de artículo ": " 49 ",
      "número d e publicación ": " 2 ",
      "año ": " 2015 ",
      "título ": " Artículo 49 "
    },
    {
      "orden ": " 9 ",
      "número de artículo ": " 54",
      "publi caciónNúmero": "2",
      "año": "2019",
      "título": "Artículo 54"
    },
    {
      "orden": "10",
      "número de artículo": "133",
      "número de publ icación " : "2",
      "año": "2019",
      "título": "Artículo 133"
    },
    {
      "orden": "11",
      "número de artículo": "3",
      "número de publicac ión": "1",
      "año ": " 2017 ",
      "title ": " Artículo 3 "
    },
    {
      "ord er ": " 12 ",
      "articleNumber ": "23 ",
      "número de publicación ": " 4 ",
      "año ": " 2015 ",
      "título ": " Artículo 23 "
    },
    {
      "ord en ": " 13 ",
      "número de artículo ": " 5 ",
      "número de publica ción ": " 1",
      "año": "2016",
      "título": "Artículo 5"
    },
    {
      "orden ": "14",
      "número de artículo": "8",
      "número de publicación": "3",
      "año " : "2019",
      "título": "Artículo 8"
    },
    {
      "orden": "15",
      "número de artículo": "1",
      "número de publicación": "3",
      "año": "2016",
      "título ": " Artículo 1 "
    },
    {
      "orden ": " 16 ",
      "número de artículo ": " 129 ",
      "número de publicación ": "4 ",
      "año ": " 2019 ",
      "título ": " Artículo 129 "
    },
    {
      "orden ": " 17 ",
      "número de artículo ": " 57 ",
      "número de publicación " : " 2 ",
      "año ": " 2019",
      "título": "Artículo 57"
    },
    {
      "orden": "18",
      "número de artículo": "106",
      "número de publicación": "1",
      "año": "2016",
      "título" : "Artículo 106"
    },
    {
      "orden": "19",
      "número de artículo": "67",
      "número de publicación": "3",
      "año": "2015",
      "título": "Artículo 67"
    },
    {
      "orden": "20",
      "número de artículo": "42",
      "número de publicación": "1",
      "añ o": "2019",
      "title": "Artículo 42"
    },
    {
      "orden": "21",
      "número de artículo": "14",
      "número de publicación": "1",
      "año": "20 15",
      "título": "Artículo 14 "
    },
    {
      "orden ": " 22 ",
      "número d e artículo ": " 76 ",
      "número de publicación ": " 4 ",
      "año ": " 2018 ",
      "título ": " Artículo 76 "
    },
    {
      "orden " : "23",
      "nú mero de artículo": "134",
      "número de publicación": "1",
      "año ": " 2016",
      "título": "Artículo 134"
    }
  ]
}
```

Entrada	Salida	Explicación
<pre> 34 "}}}}"title": "Artículo 14"}, {"orden": "22", "número de artículo": "76", "número de publicación": "4", "año": "2018" , "título": "Artículo 76 "}, {"orden": "23", "número de a rtículo": "134", "número de publicación": "1", "año": " 2016", "título": "Artículo 134 "}}}}"title": "Artículo 134 "}}}}"title": "Artículo 134"}}} {"publicador": "IEEE", "título": "Título de publicación 4", "tipo de contenido": "publicaciones periódicas", "ieeeCitati onCount": "1", "número de publicación": "21", "paperCitation s": {"ieee": [{"orden": "1", "número de artículo": "126 ", "número de publicación": "1", "año": "2015", "título ": "Artículo 126 "}, {"orden": "2", "número de artículo": "35", "número de publicación": "4", "año": "2017", "título": "Artículo 35"}, {"orden": "3", "número de artículo": "7", "publicaciónNumber": "1", "año": "2016", "título": "Art ículo 7 "}, {"orden": "4", "número de artículo": "116", "número de publicación": "2", "año": "2018", "título": "Artículo 116 "}, {"orden": "5", "número de artículo": "58 ", "número de publicación": "3", "año": "2015", "título": "A rtículo 58"}, {"orden": "6", "número de artículo": "24", "nú mero de publicación": "4", "año": "2018", "título": "Artícu lo 24"}, {"orden": "7", "número de artículo": "136", "public aciónNumber": "1", "año": "2015", "título": "Artículo 136"}, {"orden": "8", "número de artículo": "29", "número de pu blicación": "4", "año": "2019", "título": "Artículo 29 "}, {"orden": "9", "articleNumber": "118", "publishingNum ber": "2", "year": "2015", "title": "Article 118"}, {"orden" : "10", "articleNumber": "102", "publishingNumber": "2", "y ear": "2015", "title": "Article 102"}, {"orden": "11", "arti cleNumber": "41", "publishingNumber": "1", "año": "2019", "título": "Artículo 41"}, {"orden": "12", "articleNumbe r": "87", "publishingNumber": "1", "year": "2017", "ti tle": "Article 87"}, {"orden": "13", "articleNumber": " 61", "publishingNumber": "4", "year": "2019", "title": "Art icle 61"}, {"orden": "14", "articleNumber": "91", "publishin gNumber": "2", "año": "2017", "título": "Artículo 91"}, {"o rden": "15", "número de artículo": "88", "número de publicac ión": "2", "año": "2015", "title": "Artículo 88"}, {"ord er": "16", "articleNumber": "83", "publishingNumber": "1 ", "year": "2019", "title": "Article 83"}, {"orden": " 17", "articleNumber": "103", "publishingNumber": "1", " año": "2017", "título": "Artículo 103"}, {"orden": "18", "nú mero de artículo": "107", "número de publicación": "2", "año ": "2015", "título": "Artículo 107"}, {"orden": "19", "núme ro de artículo": "81", "número de publicación": "4", "año": "2016", "título": "Artículo 81"}, {"orden": "20", "númer o de artículo": "76", "número de publicación": "3", "año" : "2018", "título": "Artículo 76"}, {"orden": "21", "número </pre>		

Entrada

Salida

Explicación

de artículo": "31", "número de publicación": "1", "año": "2017", "title": "Artículo 31"]]]}

```
{
  "publicador": "IEEE",
  "título": "Título de publicación 5",
  "tipo de contenido": "publicaciones periódicas",
  "ieeeCitationCount": "2",
  "número de publicación": "15",
  "paperCitations": [
    {
      "ieee": [
        {
          "orden": "1",
          "número de artículo": "28",
          "número de publicación": "1",
          "año": "2017",
          "título": "Artículo 28"
        },
        {
          "orden": "2",
          "número de artículo": "1",
          "número de publicación": "1",
          "año": "2018",
          "título": "Artículo 1"
        },
        {
          "orden": "3",
          "número de artículo": "109",
          "número de publicación": "4",
          "año": "2018",
          "título": "Artículo 109"
        },
        {
          "orden": "4",
          "número de artículo": "82",
          "número de publicación": "1",
          "año": "2016",
          "título": "Artículo 82"
        },
        {
          "orden": "5",
          "número de artículo": "83",
          "número de publicación": "1",
          "año": "2017",
          "título": "Artículo 83"
        },
        {
          "orden": "6",
          "número de artículo": "136",
          "número de publicación": "4",
          "año": "2018",
          "título": "Artículo 136"
        },
        {
          "orden": "7",
          "número de artículo": "36",
          "número de publicación": "1",
          "año": "2018",
          "título": "Artículo 36"
        },
        {
          "orden": "8",
          "número de artículo": "83",
          "número de publicación": "4",
          "año": "2015",
          "título": "Artículo 83"
        },
        {
          "orden": "9",
          "número de artículo": "132",
          "publishingNumber": "3",
          "year": "2018",
          "title": "Article 132"
        },
        {
          "orden": "10",
          "articleNumber": "83",
          "publishingNumber": "4",
          "año": "2016",
          "título": "Artículo 83"
        },
        {
          "orden": "11",
          "número de artículo": "51",
          "número de publicación": "4",
          "año": "2015",
          "título": "Artículo 51"
        },
        {
          "orden": "12",
          "número de artículo": "37",
          "publishingNumber": "2",
          "year": "2015",
          "title": "Article 37"
        },
        {
          "orden": "13",
          "articleNumber": "112",
          "publishingNumber": "1",
          "año": "2016",
          "título": "Artículo 112"
        },
        {
          "orden": "14",
          "número de artículo": "16",
          "número de publicación": "1",
          "año": "2015",
          "título": "Artículo 16"
        },
        {
          "orden": "15",
          "número de artículo": "2",
          "número de publicación": "3",
          "año": "2019",
          "título": "Artículo 2"
        }
      ]
    },
    {
      "orden": "13",
      "número de artículo": "112",
      "número de publicación": "1",
      "año": "2016",
      "título": "Artículo 112"
    },
    {
      "orden": "14",
      "número de artículo": "16",
      "número de publicación": "1",
      "año": "2015",
      "título": "Artículo 16"
    },
    {
      "orden": "15",
      "número de artículo": "2",
      "número de publicación": "3",
      "año": "2019",
      "título": "Artículo 2"
    }
  ]
}
```

Entrada	Salida	Explicación
o de artículo": "2", "número de publicación": "3", "año": "2019", "título": "Artículo 2"}]]}}"Artículo 112"}, {"orden": "14", "número de artículo": "16", "número de publicación": "1", "año": "2015", "título": "Artículo 16"}, { "orden": "15", "número de artículo": "2", "número de publicación": "3", "año": "2019", "título": "Artículo 2"}]]}}		

Salida estándar

Liste el título de la publicación y su Factor de Impacto. Ordenar por factor de impacto descendente. Formatee la salida con un nombre de diario por línea y con el nombre y la puntuación (con dos decimales) separados por dos puntos. Ejemplo: Letters on IEEEExtreme: 1.78

En caso de que varias publicaciones tengan el mismo factor de impacto, imprimalas en orden alfabético.

Restricciones y notas

- El número máximo de registros es 1001.

Entrada	Salida	Explicación
<pre>" 4 "," número de artículo ":" 97 "," número de artículo ":" 61 "," número de publicación de artículo ":" 90 "," número de artículo ":" 116 "," número de artículo ":" 82 "," número de publicación de artículo ":" 82 "</pre>	<pre>Cartas en IEEEExtreme: 1.78 Diario de programación de 24 horas: 1.20</pre>	<p>Tomando la segunda línea de la entrada, debe generar el Factor de Impacto para publicaciones Cartas en IEEEExtreme y Diario de programación de 24 horas . Las letras en IEEEExtreme tuvieron el mayor factor de impacto, por lo que aparece primero en la salida.</p>



Alfa Universe Clasificación de peso

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Hay varias razas en el universo Alpha y cada una de ellas tiene un peso diferente. Para algunos pares de carreras, se te da el resultado de comparar su peso. A partir de la transitividad, puede deducir el orden de algunos de los pares que no recibe, pero no se conocen otros pares. Por ejemplo, si $A > B$ y $B > C$, puedes deducir que $A > C$.

Un *pedido válido* es una permutación de todas las razas, de modo que para cada par (A, B) , si $A < B$ entonces el índice de A en la permutación es menor que el índice de B .

Alguien le pide que compare el peso de dos razas que no se pueden deducir de los pares comparados conocidos. Desea responder proporcionando un número mínimo de *pedidos válidos*, de modo que:

- Por cada par (A, B) , donde no se puede deducir cuál de ellos es más ligero y cuál es más pesado, hay al menos un pedido donde A viene antes que B , y al menos un pedido donde B viene antes que A .

Por ejemplo, si hay cuatro razas: AA , BB , CC y D y sabes que:

1. El peso de CC es menor que el de BB ;
2. El peso de BB es menor que el de D ;
3. Y que el peso de CC es menor que el de AA .

En este caso, los seis pares se dividen en los cuatro órdenes conocidos:

1. $C < AC < A$
2. $B < DB < D$
3. $C < BC < B$
4. $C < DC < D$

Dos son desconocidos:

1. UNA ? siA ? si
2. UNA ? reA ? re

Puede proporcionar los siguientes dos *pedidos válidos* :

- $C < A < B < DC < A < B < D$
- $C < B < D < AC < B < D < A$

Donde todas estas permutaciones obedecen a las cuatro relaciones de orden de peso conocidas y para cualquier par desconocido existe al menos una permutación que cubre cada orden del par.

Su tarea es encontrar el número mínimo o permutaciones, *norteN* e imprima el *norteN pedidos válidos*.

Entrada estándar

La primera línea contiene el número de carreras, *norteN* y el número de comparaciones conocidas *METROM*.

El seguimiento *METROM* líneas contiene los pares en el formato *XXY* donde *XX* y *Y* son letras mayúsculas que representan la raza.

Salida estándar

La primera línea es el número mínimo. *KK* de permutaciones necesarias.

Cada uno de los siguientes *KL* las líneas *K* deben contener una permutación de la primera *norteN* letras del alfabeto.

- $K = 0$ $K = 0$ significa que el problema no se puede resolver (las comparaciones conocidas no son transitivas).
- $K = 1$ $K = 1$ significa que no hay pares indeterminados y la siguiente línea es la única permutación que se adhiere a todas las restricciones dadas.

Restricciones y notas

- $1 \leq N \leq 26$ Las carreras se denotan por el primer norte N letras en el alfabeto inglés.
- $0 \leq M \leq 3250$ Puede suponer que la entrada está en el formato correcto
- **Este es un problema de NP y es posible que nadie pueda resolver todos los casos de prueba dentro de los límites de tiempo**

Entrada	Salida	Explicación
<div>4 4</div> <div>Californi a</div> <div>BD</div> <div>CB</div> <div>CD</div>	<div>2</div> <div>CABD</div> <div>CBDA</div>	Esta muestra es el ejemplo dado anteriormente.
<div>3 2</div> <div>Californi a</div> <div>AB</div>	<div>1</div> <div>TAXI</div>	Las comparaciones dadas permiten solo una permutación: $C < A < B < D$
<div>10 10</div> <div>AB</div> <div>AJ</div> <div>DF</div>	<div>0</div>	Tome la tercera, quinta, segunda y décima comparación y obtenemos una contradicción: $D < F < A < J < D$

Entrada	Salida	Explicación
---------	--------	-------------

DC

FA

AC

EB

H

JD

AI

Salida estándar

La primera línea es el número mínimo. K de permutaciones necesarias.

Cada uno de los siguientes L as líneas K deben contener una permutación de la primera N letras del alfabeto.

- $K = 0$ significa que el problema no se puede resolver (las comparaciones conocidas no son transitivas).
- $K = 1$ significa que no hay pares indeterminados y la siguiente línea es la única permutación que se adhiere a todas las restricciones dadas.

Restricciones y notas

- $1 \leq N \leq 26$ Las carreras se denotan por el primero N letras en el alfabeto inglés.
- $0 \leq M \leq 325$ Puede suponer que la entrada está en el formato correcto
- Este es un problema de NP y es posible que nadie pueda resolver todos los casos de prueba dentro de los límites de tiempo

Entrada	Salida	Explicación
4 4 California BD CB CD	2 CABD CBDA	Esta muestra es el ejemplo dado anteriormente.

3 2
California
AB

1
TAXI

Las comparaciones dadas permiten solo una permutación: $C < A < B$

10 10
AB
AJ
DF
DC
FA
AC
EB
H
JD
AI

0

Tome la tercera, quinta, segunda y décima comparación y obtenemos una contradicción:
 $D < F < A < J < D$

<https://csacademy.com/ieeextreme13/task/alfa-universe-weight-sorting/>



Xranda y Tree

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Xranda y un hombre que no mastica viven en un **árbol** con n nodos, con etiquetas numéricas en los bordes. Su pasión por la treeología los determinó a responder a algunos problemas aparentemente imposibles como "¿cuántos árboles etiquetados se encuentran en la clase de isomorfismo de un árbol determinado?" o "¿cuál es la distancia de rotación generalizada de su árbol y alguna otra arbitraria?". Agotado por estas tareas, le piden que resuelva una más fácil.

Definimos la distancia $d(u, v)$ entre dos nodos u y v como la etiqueta más grande de un borde que pertenece a la ruta única entre ellos. Calcule la suma de las distancias a través de cada posible par de nodos, es decir:

$$\sum_{a=1}^n \sum_{b=1}^{a-1} d(a, b)$$

Dado que este número puede ser bastante grande, solo nos interesa su resto cuando se divide por $10^9 + 7$.

Entrada estándar

La primera línea contendrá el número de nodos del árbol, n .

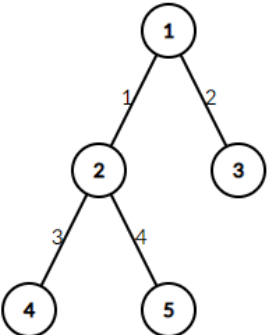
El siguiente $n-1$ líneas contendrán la descripción de los bordes del árbol, es decir, la línea $i + 1$ contendrá (en orden) números A_i , B_i y W_i , lo que significa que hay un borde entre nodos A_i y B_i con la etiqueta W_i entre ellos.

Salida estándar

La salida solo contendrá la suma deseada.

Restricciones y notas

- $2 \leq N \leq 10^5$
- $1 \leq W_i \leq 10^9$

Entrada	Salida	Explicación
5 1 2 1 1 3 2 2 4 3 2 5 4	30	

Tenemos las siguientes distancias:

$$\begin{aligned}d(1, 2) &= 1; d(1, 3) = 2; \\d(1, 4) &= 3; d(1, 5) = 4; \\d(2, 3) &= 2; d(2, 4) = 3; \\d(2, 5) &= 4; d(3, 4) = 3; \\d(3, 5) &= 4; d(4, 5) = 4\end{aligned}$$

The total sum is 30.

<https://csacademy.com/ieeextreme13/task/xrand-and-tree/>



Reorganizar equipos

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Recientemente, Sara inventó un nuevo juego de mesa y ha invitado a todos sus amigos a jugar en su casa. Todos ya están sentados en una gran mesa circular. Sara atribuye aleatoriamente un equipo, representado por una carta de Aa D, a cada amigo. Para agrupar equipos alrededor de la mesa, un subconjunto de jugadores debería levantarse y cambiar de lugar.

¿Cuál es el número mínimo de personas que necesitan cambiar su asiento para que todos los jugadores de cualquier equipo ocupen una secuencia contigua de sillas?

Entrada estándar

La primera línea de la entrada contiene el número de veces, T , que necesitas resolver el desafío.

Cada uno de la siguiente T líneas contiene una cadena S_i de longitud L_i .

Salida estándar

Para cada prueba, envíe una línea que contenga un número entero único que represente el número mínimo de personas que tienen que cambiar su asiento.

Restricciones y notas

- $1 \leq T \leq 10$

- $1 \leq L_i \leq 10^5$
- Los equipos están representados por caracteres (A, B, C o D)

Entrada	Salida	Explicación
5	0	CBBACC -> 0 (los equipos ya están sentados uno al lado del otro).
CBBACC	0	DBCA -> 0 (los equipos ya están sentados uno al lado del otro).
DBCA	2	CCACACC -> CCCAACC: 2 letras fuera de lugar
CCACACC	6	ABCABCABC -> AAABBBCCC: 6 letras fuera de lugar
ABCABCABC	12	ABCDABCDABCDABCD -> AAAABBBBCCCCDDDD: 12 letras fuera de lugar
ABCDABCDABCDABCD		

Salida estándar

Para cada prueba, envíe una línea que contenga un número entero único que represente el número mínimo de personas que tienen que cambiar su asiento.

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq L_i \leq 10^5$
- Los equipos están representados por caracteres (A, B, C o D)

Entrada	Salida	Explicación
5	0	CBBACC -> 0 (los equipos ya están sentados uno al lado del otro).
CBBACC	0	DBCA -> 0 (los equipos ya están sentados uno al lado del otro).
DBCA	2	CCACACC -> CCCAACC: 2 letras fuera de lugar
CCACACC	6	ABCABCABC -> AAABBBCCC: 6 letras fuera de lugar
ABCABCABC	12	ABCDABCDABCDABCD -> AAAABBBBCCCCDDDD: 12 letras fuera de lugar
ABCDABCDABCDABCD		

<https://csacademy.com/ieeextreme13/task/reshuffle-teams/>



Equipos de intercambio de parejas

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

En un universo paralelo, Sara también inventó un nuevo juego de mesa y ha invitado a todos sus amigos a jugar en su casa. Todos ya están sentados en una gran mesa circular. Sara atribuye aleatoriamente un equipo, representado por una carta de Aa D, a cada amigo. Para agrupar equipos alrededor de la mesa, un subconjunto de jugadores debería levantarse y cambiar de lugar.

Al contrario del otro universo, tienen que intercambiar lugares en parejas.

¿Cuál es el número mínimo de intercambios para que todos los jugadores de cualquier equipo ocupen una secuencia contigua de sillas?

Entrada estándar

La primera línea de la entrada contiene el número de veces, T , que necesitas resolver el desafío.

Cada uno de la siguiente T líneas contiene una cadena S_i de longitud L_i .

Salida estándar

Para cada prueba, envíe una línea que contenga un número entero único que represente el número mínimo de intercambios de asientos.

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq L_i \leq 10^5$
- Los equipos están representados por caracteres (A, B, C o D)
- El mismo jugador puede participar en intercambios de asientos múltiples

Entrada

Salida

```
5
CBBACC
DBCA
CCACACC
ABCABCABC
ABCDABCDABCDABCD
```

```
0
0
1
3
6
```

Salida estándar

Para cada prueba, envíe una línea que contenga un número entero único que represente el número mínimo de intercambios de asientos.

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq L_i \leq 10^5$
- Los equipos están representados por caracteres (A, B, C o D)
- El mismo jugador puede participar en intercambios de asientos múltiples

Entrada

Salida

```
5
CBBACC
DBCA
CCACACC
ABCABCABC
ABCDABCDABCDABCD
```

```
0
0
1
3
6
```

<https://csacademy.com/ieeextreme13/task/pair-swap-teams/>



Unicornosaurus

Límite de tiempo: 800 ms Límite de memoria: 256 MB

Tiranca el unicornosaurus rompió el duckinator de Sherbot *. Después de una inspección minuciosa, se dio cuenta de que habían n intervalos $[L_i, R_i]$ de tableros rotos. En cada intervalo, todos los tableros numerados de L_i a R_i fueron completamente destruidos.

Afortunadamente, Tiranca es un unicornosaurio y posee M acciones de superpotencia para reparar el duckinator. Cada superpotencia cuesta C_i arco iris y se puede utilizar para reparar todas las tablas en el intervalo $[A_i, B_i]$. Para arreglar el duckinator, tiene que arreglar todos los tableros rotos y, debido a que los arcoirises son caros, debe usar la menor cantidad posible de arcoirises.

1 * *Duckinator* : valla larga hecha de tableros anchos utilizados para atrapar patos. Las tablas están dispuestas en una línea, numeradas de 1 a S .

Entrada estándar

La primera línea contiene tres enteros, N , M y S . Los i -ésimas siguientes n líneas contienen dos enteros, L_i y R_i , describiendo el daño hecho al duckinator. The i -th following M lines contain three integers A_i , B_i and C_i , describing Tiranca's possible superpower actions.

Standard output

The output should contain the minimum number of rainbows required to repair the duckinator.
In case it is impossible to fix, the output should be -1.

Constraints and notes

- $1 \leq S \leq 10^5$
- $1 \leq C_i \leq 10^9$
- $1 \leq L_i \leq R_i \leq S$
- $1 \leq N, M \leq 10^5$
- The intervals of broken planks **may overlap**.
- The author made up the term **duckinator**.

Input	Output	Explanation
<pre> 1 3 15 5 10 3 7 2 6 12 5 2 11 6 </pre>	6	<p>There is one broken segment: [5, 10][5,10].</p> <p>We choose [2, 11][2,11] with the total cost of 66 to fix it.</p> <p>Note that another solution would be choosing [3, 7][3,7] and [6, 12][6,12], but the total cost of that would be 77.</p>
<pre> 2 4 15 3 7 9 10 2 6 10 3 9 15 5 12 13 8 10 30 </pre>	23	<p>There are two broken segments: [3, 7][3,7] and [9, 10][9,10].</p> <p>We choose [2, 6][2,6] and [5, 12][5,12] with the total cost of 2323 to fix it.</p>

Salida estándar

La salida debe contener el número mínimo de arcos necesarios para reparar el duckinator. En caso de que sea imposible de arreglar, la salida debería serlo `-1`.

Restricciones y notas

- $1 \leq S \leq 10^5$
- $1 \leq C_i \leq 10^9$
- $1 \leq L_i \leq R_i \leq S$
- $1 \leq N, M \leq 10^5$
- Los intervalos de tablas rotas **pueden superponerse**.
- El autor creó el término **duckinator**.

Entrada	Salida	Explicación
<pre>1 3 15 5 10 3 7 2 6 12 5 2 11 6</pre>	<pre>6</pre>	<p>Hay un segmento roto: $[5, 10]$.</p> <p>Nosotros elegimos $[2, 11]$ con el costo total de 6 para arreglarlo.</p> <p>Tenga en cuenta que otra solución sería elegir $[3, 7]$ y $[6, 12]$, pero el costo total de eso sería 7.</p>
<pre>2 4 15 3 7 9 10 2 6 10 3 9 15 5 12 13 8 10 30</pre>	<pre>23</pre>	<p>Hay dos segmentos rotos: $[3, 7]$ y $[9, 10]$.</p> <p>Nosotros elegimos $[2, 6]$ y $[5, 12]$ with the total cost of 23 to fix it.</p>

<https://csacademy.com/ieeextreme13/task/unicornosaurus/>



Pares estrictamente convexos

Límite de tiempo: 3000 ms Límite de memoria: 256 MB

Hay un polígono con n vértices en coordenadas cartesianas 2D estándar. El polígono es **estrictamente convexo**. Es decir, el polígono es convexo y no hay tres vértices colineales (es decir, que se encuentran en la misma línea recta).

Existen M otros puntos distintos que se encuentran estrictamente dentro o fuera del polígono (no en los bordes del polígono). Andy quiere elegir dos puntos diferentes a y b de esos M puntos para que pueda conectarse a y b con un **segmento de línea** recta sin intersectar o tocar el polígono. Andy tiene curiosidad acerca de cuántos pares de puntos desordenados diferentes a y b he can pick. Can you help Andy to count that?

Note: An unordered pair (a, b) is the same as the pair (b, a) .

Standard input

The first line of the input contains two integers, N and M .

The next N lines describe the convex polygon. Each line contains two integers indicating x -coordinate and y -coordinate of one vertex. The vertices are given in counter-clockwise order.

The next M lines describe points that Andy can pick. Each line contains two integers indicating x -coordinate and y -coordinate of one point.

Standard output

Print one integer representing the number of pairs.

Constraints and notes

- $3 \leq N \leq 2 \times 10^5$
- $2 \leq M \leq 2 \times 10^5$
- $-10^9 \leq x\text{-coordinate}, y\text{-coordinate} \leq 10^9$
- The polygon is strictly convex.
- All M points are distinct and not located on any polygon edges.

Input	Output	Explanation
4 6 4 0 0 4 -4 0 0 -4 5 0 1 0 -1 0 3 3 4 -2 4 4	5	-4-20246810-4-2024(5,0)(1,0)(-1,0)(3,3)(4,-2)(4,4) The five pairs that can be picked are: <ul style="list-style-type: none">point (5, 0)(5,0) and point (3, 3)(3,3)point (5, 0)(5,0) and point (4, -2)(4,-2)point (5, 0)(5,0) and point (4, 4)(4,4)point (1, 0)(1,0) and point (-1, 0)(-1,0)point (3, 3)(3,3) and point (4, 4)(4,4)

Entrada estándar

La primera línea de la entrada contiene dos enteros, N y M .

El siguiente N líneas describen el polígono convexo. Cada línea contiene dos enteros que indican x -coordenada y y -coordenada de un vértice. Los vértices se dan en orden antihorario.

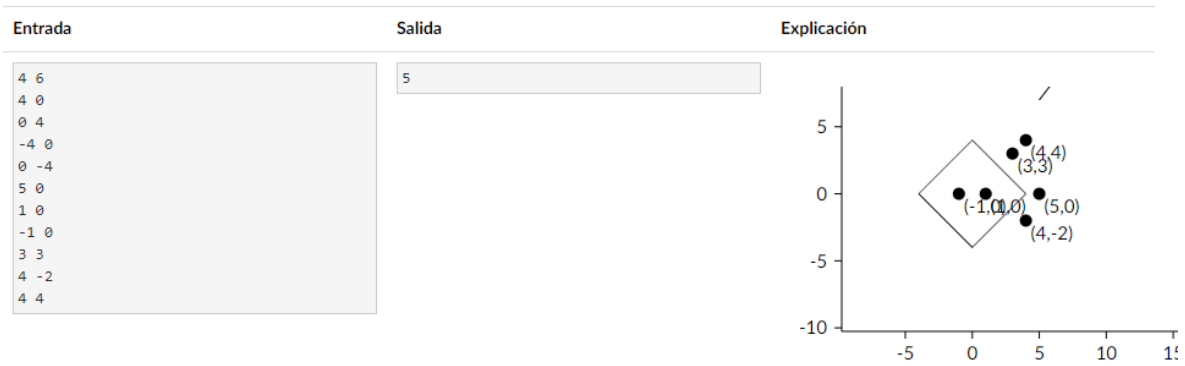
El siguiente M líneas describen puntos que Andy puede elegir. Cada línea contiene dos enteros que indican x -coordenada y y -coordenada de un punto.

Salida estándar

Imprime un número entero que representa el número de pares.

Restricciones y notas

- $3 \leq N \leq 2 \times 10^5$
- $2 \leq M \leq 2 \times 10^5$
- $-10^9 \leq x\text{-coordinate}, y\text{-coordinate} \leq 10^9$
- The polygon is strictly convex.
- All M points are distinct and not located on any polygon edges.



Los cinco pares que se pueden elegir son:

- punto (5 , 0) y punto(3, 3)
- punto (5 , 0) y punto(4, -2)
- punto (5 , 0) y punto(4, 4)
- punto (1 , 0) y punto(-1, 0)
- punto (3, 3) and point (4, 4)

<https://csacademy.com/ieeextreme13/task/strictly-convex-pairs/>



Charlie nueva PC

Límite de tiempo: 2000 ms Límite de memoria: 256 MB

Charlie ha recibido dinero para construir una nueva PC. Necesita su ayuda para seleccionar componentes, utilizando la mayor cantidad de dinero posible, sin exceder el presupuesto. Hay varios tipos de componentes para elegir (por ejemplo, RAM, disco duro, procesador, etc.), y debe seleccionar exactamente uno de cada tipo de componente para la computadora.

Entrada estándar

La primera línea de la entrada especifica un número entero TT , el número de casos de prueba. Cada caso de prueba comienza con un número entero BB , que representa la cantidad máxima que Charlie puede gastar. La siguiente línea contiene un número entero $norteN$, que representa el número de tipos de componentes. La siguiente línea contiene $norteN$ enteros separados por espacios que representan la cantidad de opciones K_i , for each component type $1 \leq i \leq N$. Each of the next NN lines contain the costs for the components. The i^{th} line contains a space separated list of K_i integers indicating the costs of each component option.

Standard output

For each test case, you should output a single line with the maximum cost of the computer, using exactly one of each component types. The cost needs to be less or equal to the maximum amount available BB . If there is no configuration that can be constructed for the given budget, the output should be \emptyset .

Constraints and notes

- $1 \leq T \leq 10$
- $1 \leq B \leq 2 * 10^9$
- $1 \leq N \leq 10$
- $1 \leq K_i \leq 100$
- The sum of all options $K_i \leq 100$
- The costs of the components are integers between 11 and $2 * 10^9$

Input	Output	Explanation
<pre> 1 50 4 3 2 1 2 15 10 49 11 17 10 13 23 </pre>	50	<p>In the sample input, there is one test case.</p> <p>This test case has a b budget of 50, and there are 4 types of components.</p> <ul style="list-style-type: none"> • Three options for component 1, with cost 15, 10, and 49; • Two options for component 2, with cost 11 and 17; • One options for component 3, with costs 10; • Two options for component 4, with cost 13 and 23. <p>The most expensive computer that Charlie can build will cost 50, choosing the 10 option for component 1, the 17 option for component 2, the 10 option for component 3, and the 13 option for component 4.</p>

Salida estándar

Para cada caso de prueba, debe generar una sola línea con el costo máximo de la computadora, utilizando exactamente uno de cada tipo de componente. El costo debe ser menor o igual a la cantidad máxima disponible B . Si no hay una configuración que se pueda construir para el presupuesto dado, la salida debería ser 0.

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq B \leq 2 * 10^9$
- $1 \leq N \leq 10$
- $1 \leq K_i \leq 100$
- The sum of all options $K_i \leq 100$
- The costs of the components are integers between 1 and $2 * 10^9$

Input	Output	Explanation
<pre> 1 50 4 3 2 1 2 15 10 49 11 17 10 13 23 </pre>	<pre> 50 </pre>	<p>In the sample input, there is one test case.</p> <p>This test case has a b budget of 50, and there are 4 types of components.</p> <ul style="list-style-type: none"> • Three options for component 1, with cost 15, 10, and 49; • Two options for component 2, with cost 11 and 17; • One options for component 3, with costs 10; • Two options for component 4, with cost 13 and 23. <p>The most expensive computer that Charlie can build will cost 50, choosing the 10 option for component 1, the 17 option for component 2, the 10 option for component 3, and the 13 option for component 4.</p>



Charlie nueva PC

Límite de tiempo: 2000 ms Límite de memoria: 256 MB

Charlie ha recibido dinero para construir una nueva PC. Necesita su ayuda para seleccionar componentes, utilizando la mayor cantidad de dinero posible, sin exceder el presupuesto. Hay varios tipos de componentes para elegir (por ejemplo, RAM, disco duro, procesador, etc.), y debe seleccionar exactamente uno de cada tipo de componente para la computadora.

Entrada estándar

La primera línea de la entrada especifica un número entero TT , el número de casos de prueba. Cada caso de prueba comienza con un número entero BB , que representa la cantidad máxima que Charlie puede gastar. La siguiente línea contiene un número entero $norteN$, que representa el número de tipos de componentes. La siguiente línea contiene $norteN$ enteros separados por espacios que representan la cantidad de opciones K_i , para cada tipo de componente $1 \leq i \leq N$. Cada uno de los siguientes $norteN$ líneas contienen los costos de los componentes. i^{th} línea contiene una lista espacio separado de K_i enteros que indican los costos de cada opción de componentes.

Salida estándar

Para cada caso de prueba, debe generar una sola línea con el costo máximo de la computadora, utilizando exactamente uno de cada tipo de componente. El costo debe ser menor o igual a la cantidad máxima disponible BB . Si no hay una configuración que se pueda construir para el presupuesto dado, la salida debería ser 0.

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq B \leq 2 * 10^9$
- $1 \leq N \leq 10$
- $1 \leq K_i \leq 100$
- La suma de todas las opciones. $K_i \leq 100$
- Los costos de los componentes son enteros entre 1 y $2 * 10^9$

Entrada Salida Explicación

1
50
4
3 2 1 2
15 10 4
9
11 17
10
13 23

50

En la entrada de muestra, hay un caso de prueba.

This test case has a **b** budget of 50, and there are 4 types of components.

- Three options for component 1, with cost 15, 10, and 49;
- Two options for component 2, with cost 11 and 17;
- One options for component 3, with costs 10;
- Two options for component 4, with cost 13 and 23.

The most expensive computer that Charlie can build will cost 50, choosing the 10 option for component 1, the 17 option for component 2, the 10 option for component 3, and the 13 option for component 4.

Salida estándar

Para cada caso de prueba, debe generar una sola línea con el costo máximo de la computadora, utilizando exactamente uno de cada tipo de componente. El costo debe ser menor o igual a la cantidad máxima disponible. B . Si no hay una configuración que se pueda construir para el presupuesto dado, la salida debería ser 0 .

Restricciones y notas

- $1 \leq T \leq 10$
- $1 \leq B \leq 2 * 10^9$
- $1 \leq N \leq 10$
- $1 \leq K_i \leq 100$
- La suma de todas las opciones. $K_i \leq 100$
- Los costos de los componentes son enteros entre 1 y $2 * 10^9$

Entrada	Salida	Explicación
<pre> 1 50 4 3 2 1 2 15 10 49 11 17 10 13 23 </pre>	50	<p>En la entrada de muestra, hay un caso de prueba.</p> <p>This test case has a b budget of 50, and there are 4 types of components.</p> <ul style="list-style-type: none"> • Three options for component 1, with cost 15, 10, and 49; • Two options for component 2, with cost 11 and 17; • One options for component 3, with costs 10; • Two options for component 4, with cost 13 and 23. <p>The most expensive computer that Charlie can build will cost 50, choosing the 10 option for component 1, the 17 option for component 2, the 10 option for component 3, and the 13 option for component 4.</p>

<https://csacademy.com/ieeextreme13/task/charlie-new-pc/>



Puzzle of Rooks

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Algunos predicen que por año 2050 el AI podrá resolver la mayoría de los desafíos que enfrentamos hoy. Eres fanático de la predicción, pero tu amigo Lenny aún no está convencido. ¡Por lo tanto, debes demostrarle el poder de la IA resolviendo el *Rompecabezas de Rooks*!

Hay un tablero de ajedrez de tamaño infinito. Cada celda se identifica en función de sus coordenadas horizontales y verticales utilizando un par entero (x, y) . Existen n torres ubicadas en n celdas iniciales en el tablero de ajedrez. Hay adicionalmente m celdas objetivo. Para resolver el rompecabezas, uno debe mover n torres, de modo que cada una de las celdas objetivo contiene exactamente una torre, y no hay dos torres que se ataquen entre sí en cualquier momento. Dos torres se atacan entre sí si comparten una misma fila o columna, es decir, sus celdas tienen el mismo x o igual y coordenada. Se pueden realizar las siguientes operaciones:

- Mueva una torre en el tablero de ajedrez a una de sus celdas adyacentes horizontal o verticalmente. Es decir, mover una torre a la celda (x, y) izquierda, derecha, arriba o abajo cambia su celda a $(x-1, y)$, $(x+1, y)$, $(x, y+1)$, $(x, y-1)$ respectively.
- Registre la posición de una torre y retírela temporalmente del tablero de ajedrez. La celda de la torre se vacía. Como máximo, una torre puede eliminarse temporalmente a la vez. En otras palabras, el tablero de ajedrez siempre tiene al menos $n - 1$ torres sobre ella.

- Vuelva a colocar la torre eliminada en su posición registrada en el tablero de ajedrez. Se puede eliminar una torre nueva después de volver a colocar una torre eliminada anteriormente.

Debes crear una IA que resuelva el Rompecabezas de Rooks. La IA encontrará una secuencia de operaciones que moverá n torres a n posiciones objetivo sin dejar que las torres se ataquen entre sí durante todo el proceso. Lenny también es muy exigente. Solo estará convencida si tu IA es lo suficientemente eficiente. Más específicamente, tu IA resolverá el rompecabezas con no más de 20502050 operations!

Standard input

The input has one integer n on the first line.

Each of the next n lines has two integers describing the initial cell of a rook.

Each of the next n lines has two integers describing a target cell.

Standard output

Output a single integer k ($k \leq 20502050$) on the first line, the number of operations your AI needs. Then print k lines, each line is in the format $x\ y\ op$. op is a single character describing the type of operation performed on cell (x, y) :

- L, R, U, D: Move the rook currently at cell (x, y) left, right, up, or down.
- T: Temporarily remove the rook at cell (x, y) .
- P: Put the temporarily removed rook back to its recorded position (x, y) .

Note that as the chessboard is infinitely large, it is allowed that a rook is moved to a cell with negative coordinates.

If the output sequence contains more than 20502050 operations, or any of the operation provided is invalid (e.g. an operation results in two rooks attacking each other; attempting to move a rook at (x, y) but the cell (x, y) is empty; putting a removed rook back to a position different from what was recorded), your solution will receive *Wrong Answer*.

Constraints and notes

- $1 \leq n \leq 10$
- All coordinates of the initial and target cells are between 11 and 9999 inclusive.
- No two initial cells share a row or column.
- No two target cells share a row or column.
- At least one target cell is not among the initial cells.
- It can be proved that for any initial and target cells satisfying the given constraints a sequence of no more than 20502050 operations exists to solve the puzzle.

Input**Output**

2

1 1

2 2

1 2

2 1

6

2 2 T

1 1 R

2 1 R

2 2 P

2 2 L

3 1 L

3

1 2

2 3

3 1

3 2

2 1

1 3

18

2 3 U

1 2 L

2 4 T

3 1 L

2 1 L

2 4 P

0 2 T

1 1 U

1 2 U

0 2 P

1 3 T

0 2 D

2 4 D

2 3 R

3 3 D

0 1 R

1 1 R

1 3 P

Salida estándar

Salida de un solo entero k ($k \leq 2050$) on the first line, the number of operations your AI needs. Then print k lines, each line is in the format $x\ y\ op$. op is a single character describing the type of operation performed on cell (x, y) :

- **L**, **R**, **U**, **D**: Move the rook currently at cell (x, y) left, right, up, or down.
- **T**: Temporarily remove the rook at cell (x, y) .
- **P**: Put the temporarily removed rook back to its recorded position (x, y) .

Note that as the chessboard is infinitely large, it is allowed that a rook is moved to a cell with negative coordinates.

If the output sequence contains more than 2050 operations, or any of the operation provided is invalid (e.g. an operation results in two rooks attacking each other; attempting to move a rook at (x, y) but the cell (x, y) is empty; putting a removed rook back to a position different from what was recorded), your solution will receive *Wrong Answer*.

Constraints and notes

- $1 \leq n \leq 10$
- All coordinates of the initial and target cells are between 1 and 99 inclusive.
- No two initial cells share a row or column.
- No two target cells share a row or column.
- At least one target cell is not among the initial cells.
- It can be proved that for any initial and target cells satisfying the given constraints a sequence of no more than 2050 operations exists to solve the puzzle.

Input

```
2
1 1
2 2
1 2
2 1
```

Output

```
6
2 2 T
1 1 R
2 1 R
2 2 P
2 2 L
3 1 L
```

```
3
1 2
2 3
3 1
3 2
2 1
1 3
```

```
18
2 3 U
1 2 L
2 4 T
3 1 L
2 1 L
2 4 P
0 2 T
1 1 U
1 2 U
0 2 P
1 3 T
0 2 D
2 4 D
2 3 R
3 3 D
0 1 R
1 1 R
```



Minimum Permutation

Time limit: 1000 ms
Memory limit: 256 MB

You are given an array AA of size NN and a set SS with MM elements. Each number from 11 to $N + MN + M$ occurs exactly once in either AA or SS .

You need to insert the elements from the set SS into the array AA in order to obtain the lexicographically smallest permutation.

Note: A Sequence $X_{\{1..n\}}X_{1..n}$, of the same length as a sequence $Y_{\{1..n\}}Y_{1..n}$, is considered lexicographically smaller than $Y_{\{1..n\}}Y_{1..n}$ if, and only if, there exists an index jj for which $X_{\{i\}}X_i = Y_{\{i\}}Y_i$ for $1 \leq i \leq j-1$, and $X_j < Y_j$.

Standard input

The first line contains the numbers NN and MM . The second line contains NN integers corresponding to the elements of AA . The third line contains MM integers corresponding to the elements of SS .

Standard output

The output should contain the elements of the lexicographically smallest permutation, separated by single spaces.

Constraints and notes

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 10^5$
- $2 \leq N + M \leq 10^5$
- All the elements are unique and between 1 and $N + M$

Input	Output	Explanation
-------	--------	-------------

3 2
3 1 5
4 2

2 3 1 4 5

The array is $[3, 1, 5]$ and the set is $\{4, 2\}$.

The first operation is to insert 2 at the beginning of the array, obtaining $[2, 3, 1, 5]$

The second operation is to insert 4 right after 1, obtaining $[2, 3, 1, 4, 5]$

Standard input

The first line contains the numbers N and M . The second line contains N integers corresponding to the elements of A . The third line contains M integers corresponding to the elements of S .

Standard output

The output should contain the elements of the lexicographically smallest permutation, separated by single spaces.

Constraints and notes

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 10^5$
- $2 \leq N + M \leq 10^5$
- All the elements are unique and between 1 and $N + M$

Input	Output	Explanation
<div>3 2 3 1 5 4 2</div>	<div>2 3 1 4 5</div>	<p>The array is $[3, 1, 5]$ and the set is $\{4, 2\}$.</p> <p>The first operation is to insert 2 at the beginning of the array, obtaining $[2, 3, 1, 5]$.</p> <p>The second operation is to insert 4 right after 1, obtaining $[2, 3, 1, 4, 5]$.</p>

<https://csacademy.com/ieeextreme13/task/minimum-permutation/>



Asociación de comerciantes

Límite de tiempo: 3000 ms Límite de memoria: 512 MB

Como líder de la Asociación de Comerciantes de Xtreme, está tratando de expandir su negocio comercial ya lucrativo. Existen n pueblos en el estado, numerados 1 a n . Las ciudades están conectadas por $n - 1$ caminos bidireccionales. Todas las ciudades compran y venden el mismo tipo de golosinas. Pueblo i compra y venta de una golosina por un precio de P_i dólares.

Dentro de la asociación, hay k comerciantes ubicados en k diferentes pueblos de inicio. Cada comerciante puede elegir una ciudad de destino y viajar a ella desde su ciudad de inicio. El comerciante debe visitar cada ciudad en el camino elegido exactamente una vez y no puede regresar. Para evitar la competencia dentro de la Asociación Xtreme, no hay rutas de viaje de dos comerciantes que puedan incluir la misma ciudad (en particular, un comerciante no puede visitar la ciudad inicial de otro comerciante). Un comerciante puede comprar exactamente un obsequio en cualquier ciudad en el camino de viaje, y vender ese obsequio en la ciudad de destino para obtener ganancias. Cada comerciante solo puede comprar y vender una vez, pero puede optar por no viajar y quedarse en la ciudad inicial si no se pueden obtener ganancias.

El objetivo de la asociación es que todos los comerciantes juntos obtengan el mayor beneficio posible. Observe detenidamente los precios de las golosinas e instruya a los comerciantes sobre dónde deberían estar sus destinos y cómo deberían comprar y vender. ¿Cuál es el beneficio total óptimo que pueden obtener los comerciantes?

Entrada estándar

La entrada tiene un entero n en la primera línea que representa el número de ciudades.

La segunda línea tiene n enteros que describen los precios en cada ciudad. p_i el i -ésimo entero es p_i .

Cada uno de los siguientes $n - 1$ líneas tiene dos enteros x, y , y que describe una ciudad bidireccional que conecta la ciudad x a la ciudad y .

Por último hay una línea con un número entero k , el número de comerciantes. Esto es seguido por una línea con k enteros distintos que dan las ciudades iniciales de cada comerciante.

Salida estándar

Produce un solo entero, el mayor beneficio total que los comerciantes pueden obtener.

Restricciones y notas

- $1 \leq n \leq 2 \times 10^5$
- $1 \leq p_i \leq 10^9$
- $1 \leq k \leq n$
- Se garantiza que todas las ciudades están conectadas por carreteras.

Entrada	Salida	Explicación
12 3 1 2 3 5 6 7 2 2 4 10 11 1 2 1 3 4 2 5 2 3 6 6 7 6 8 6 9 8 10 9 11 9 12 5	12	<ul style="list-style-type: none">El mercader empezando en la ciudad 4 viaja a la ciudad 5, comprando en la ciudad 2 para 1 dólar y vendiendo en la ciudad 5 para 5 dólares (ganando 4 dólares).El mercader empezando en la ciudad 3 viaja a la ciudad 1, comprando en la ciudad 3 para 2 dólares y venta en el pueblo 1 para 3 dólares (ganando 1 dólar).El mercader empezando en la ciudad 8 viaja a la ciudad 10, comprando en la ciudad 8 para 2 dólares y venta en el pueblo 10 para 4 dólares (ganando 2 dólares).El mercader empezando en la ciudad 11 viaja a la ciudad 7, comprando en la ciudad 9 para 2 dólares y venta en el pueblo 7 para 7 dólares (ganando 5 dólares).El mercader empezando en la ciudad 12 no viaja.

Entrada	Salida	Explicación
---------	--------	-------------

3 8 4 11 12

Restricciones y notas

- $1 \leq n \leq 2 \times 10^5$
- $1 \leq p_i \leq 10^9$
- $1 \leq k \leq n$
- Se garantiza que todas las ciudades están conectadas por carreteras.

Entrada	Salida	Explicación
<div> 12 3 1 2 3 5 6 7 2 2 4 10 11 1 2 1 3 4 2 5 2 3 6 6 7 6 8 6 9 8 10 9 11 9 12 5 3 8 4 11 12 </div>	<div> 12 </div>	<ul style="list-style-type: none"> El mercader empezando en la ciudad 4 viajes a la ciudad5 , comprando en la ciudad2 para1 dólar y vendiendo en la ciudad5 para5 dólares (ganando4 dolares). El mercader empezando en la ciudad 3 viajes a la ciudad1 , comprando en la ciudad3 para2 dolares y venta en el pueblo1 para3 dólares (ganando1 dólar) El mercader empezando en la ciudad 8 viajes a la ciudad1 0 , comprando en la ciudad8 para2 dolares y venta en el pueblo1 0 para4 dólares (ganando2 dolares). El mercader empezando en la ciudad 1 1 viaja a la ciudad7 , comprando en la ciudad9 para2 dolares y venta en el pueblo7 para7 dólares (ganando5 dolares). El mercader empezando en la ciudad 1 2 no viaja.

<https://csacademy.com/ieeextreme13/task/merchant-association/>



Raperos Xtreme

Límite de tiempo: 100 ms Límite de memoria: 256 MB

La rapera CrissyChris y su amiga, JaCe, participarán en Gold Level, uno de los eventos de estilo libre más importantes del mundo. Están entrenando improvisando frases de 3 palabras, donde cada una tiene que aportar al menos una palabra.

Quieren improvisar la cantidad máxima de frases sin repetir palabras (una palabra se puede usar solo en una frase), CrissyChris y JaCe seleccionan una variedad de palabras de la base de datos de Ctoke (Ctoke es uno de los raperos más antiguos en estilo libre y su base de datos contiene solo palabras únicas) CrissyChris y JaCe ambos seleccionan `KK` y `JJ` palabras de la base de datos respectivamente. CrissyChris y JaCe nunca tienen palabras en común.

CrissyChris es una leyenda real dentro de IEEE y ha apoyado la competencia IEEE Xtreme durante los últimos 13 años de una forma u otra; ¡ya sea durante todas esas noches en las competiciones o comprando y enviando todos los 100 mejores premios! Recientemente se mudó dentro de IEEE a otro departamento. ¡Este desafío es un homenaje a ella y todo lo que ha hecho para enriquecer la competencia a través de los años! ¡Adiós Christine Eldridge, te echaremos de menos!

Entrada estándar

Una línea con dos enteros. `KK` y `JJ`, que representa el número de palabras disponibles para cada rapero.

Salida estándar

Imprima la cantidad máxima de frases de 3 palabras que pueden improvisar juntas.

Restricciones y notas

- $0 \leq K, J \leq 10^{10}$

Entrada	Salida	Explicación
4 3	2	Con la entrada de muestra, podemos formar como máximo dos frases de 3 palabras. Una frase puede tener 2 palabras de JaCe y 1 palabra de CrissyChris y la otra frase puede tener 2 palabras de CrissyChris y 1 de JaCe.
0 10000000000000	0	Sin palabras para CrissyChris no se pueden crear frases.

Entrada estándar

Una línea con dos enteros. K y J , que representa el número de palabras disponibles para cada rapero.

Salida estándar

Imprima la cantidad máxima de frases de 3 palabras que pueden improvisar juntas.

Restricciones y notas

- $0 \leq K, J \leq 10^{10}$

Entrada	Salida	Explicación
4 3	2	Con la entrada de muestra, podemos formar como máximo dos frases de 3 palabras. Una frase puede tener 2 palabras de JaCe y 1 palabra de CrissyChris y la otra frase puede tener 2 palabras de CrissyChris y 1 de JaCe.
0 10000000000000	0	Sin palabras para CrissyChris no se pueden crear frases.

<https://csacademy.com/ieeextreme13/task/xtreme-rappers/>



Adición léxica

Límite de tiempo: 1000 ms Límite de memoria: 256 MB

Michael es un niño que ama las calculadoras. Está realmente obsesionado con cualquier cosa sobre dígitos y enteros. También tiene mucha curiosidad por las operaciones, como la suma, la resta, la división y la multiplicación. Todos los días, juega con una calculadora, ingresa algunos números, realiza algunas operaciones y luego la calculadora da el resultado. Repite este patrón varias veces hasta que se aburre.

Un día, Michael descubre algo interesante sobre la suma de los enteros. Se da cuenta de que se pueden sumar diferentes pares de números para obtener el mismo resultado. Pares como $5 + 75 + 7$ y $8 + 48 + 4$, que son iguales a 121_2 . Para profundizar aún más este descubrimiento fortuito, se da cuenta de que esta propiedad interesante se aplica a grupos de números mayores que dos, como $4 + 6 + 64 + 6 + 6$ y $5 + 8 + 35 + 8 + 3$, que son iguales a 161_6 . Se pregunta si se puede observar un comportamiento tan extraño en grupos de cuatro o cinco enteros y, para su asombro, ¡sigue aumentando el tamaño de las secuencias y esta propiedad siempre está presente!

Se pregunta si hay una manera de revertir todo el proceso. Propone la existencia de un número entero $norteN$, y adivina la secuencia de enteros que, cuando se agrega, da como resultado $norteN$. Encontró que para algún número entero $norteN$, su experimento mental podría tener una solución múltiple.

Por ejemplo, 161_6 se puede expresar a partir de la suma de estas secuencias:

- $[1, 15]_{[1,15]}$
- $[1, 1, 14]_{[1,1,14]}$

- [5, 5, 6][5,5,6]
- [4, 6, 6][4,6,6]
- [5, 8, 3][5,8,3]
- y así.....

Debido a que el número de configuraciones diferentes es tan grande, Michael quiere restringir los elementos utilizados en las secuencias. Elige otros 2 enteros AA y BB , de modo que cada elemento de la secuencia es un número entero mayor o igual que AA y también menor o igual que BB . Más formalmente, para cada elemento XX en la secuencia, $A \leq X \leq B$. El elemento no tiene que ser único.

Teniendo en cuenta esta restricción, si $N = 16, A = 4, B = 6$ algunas de las secuencias válidas con la suma N son:

- [4, 4, 4, 4][4,4,4,4]
- [4, 6, 6][4,6,6]
- [6, 4, 6][6,4,6]
- [6, 6, 4][6,6,4]
- [5, 5, 6][5,5,6]
- y así...

Para unos tres enteros N, AA y BB , se pregunta, ¿pueden N siempre se expresará como la suma de una secuencia válida de enteros que cumplan todas las restricciones anteriores? Si hay varias secuencias, quiere la secuencia más corta. Si hay múltiples secuencias con la longitud más corta, quiere la secuencia lexicográfica más pequeña.

Nota: una secuencia $X_{1..n}$, de la misma longitud que una secuencia $Y_{1..n}$, es considerado lexicográficamente más pequeña que $Y_{1..n}$ si, y solamente si, existe un índice j para el cual $X_i = Y_i$ para $1 \leq i < j$, y $X_j < Y_j$.

Entonces para:

- $N = 4, A = 2, B = 3$, es posible encontrar una secuencia: $4 = 2 + 2 = 2 + 2$;
- $N = 10, A = 6, B = 9$, es imposible encontrar una secuencia;
- $N = 251, A = 40, B = 51$, hay múltiples secuencias, pero de todas las más cortas la lexicográficamente más pequeña es $251 = 47 + 51 + 51 + 51 + 51$

Entrada estándar

Hay tres enteros N, AA y BB en una sola línea, separada por espacios individuales.

Salida estándar

Si es imposible, envíe una sola línea NO. De lo contrario, salida YES en la primera línea. Luego, genera la secuencia de enteros en la segunda línea, separados por espacios individuales.

Restricciones y notas

- $1 \leq N \leq 10^{15}$
- $1 \leq A \leq B \leq 10^{15}$
- Se garantiza que si existe una secuencia, no contiene más de 10^5 elementos.

Entrada	Salida	Explicación
4 2 3	SI 2 2	Sólo hay una solución: 2 2
59 8 10	SI 9 10 10 1 0 10 10	<p>Hay varias secuencias posibles:</p> <ul style="list-style-type: none"> • 10 10 10 10 10 9 • 9 10 10 10 10 10 • 8 8 8 9 9 9 8 • 8 8 8 8 9 9 9 • ... <p>Las dos primeras secuencias tienen la longitud más corta con 66 elementos cada uno. La segunda secuencia es lexicográficamente más pequeña que la primera secuencia.</p>
10 6 9	NO	No existe secuencia para esta entrada.

Restricciones y notas

- $1 \leq N \leq 10^{15}$
- $1 \leq A \leq B \leq 10^{15}$
- Se garantiza que si existe una secuencia, no contiene más de 10^5 elementos.

Entrada	Salida	Explicación
4 2 3	SI 2 2	Sólo hay una solución: 2 2
59 8 10	SI 9 10 10 10 10 10	Hay varias secuencias posibles: <ul style="list-style-type: none">• 10 10 10 10 10 9• 9 10 10 10 10 10• 8 8 8 9 9 9 8• 8 8 8 8 9 9 9• ... Las dos primeras secuencias tienen la longitud más corta con 6 elementos cada uno. La segunda secuencia es lexicográficamente más pequeña que la primera secuencia.
10 6 9	NO	No existe secuencia para esta entrada.

<https://csacademy.com/ieeextreme13/task/lexical-addition/>



Bienvenido a IEEE Xtreme 13.0

Límite de tiempo: 1280 ms Límite de memoria: 264 MB

En nombre del comité ejecutivo, ¡BIENVENIDO al 13º concurso de programación IEEE Xtreme! Nuestro lema para IEEE Xtreme 13.0 es "No estás solo" y con más de 10,000 participantes y voluntarios, esto nunca ha sido más cierto.

Durante las próximas 24 horas, se lanzarán desafíos a intervalos regulares. Esté atento a la página de anuncios para actualizaciones. ¡Es importante que disfrutes este evento tanto como puedas! ¡Cientos de voluntarios han estado trabajando durante meses para preparar esta fiesta de programación para usted! Asegúrese de descansar lo suficiente, mantenerse hidratado y no olvide comer. Por encima de todo, ¡DIVERTIRSE LO MÁS POSIBLE!

El equipo técnico estará en línea en todo momento. Haremos todo lo posible para responder preguntas, pero tenga en cuenta que su código no puede revisarse bajo ninguna circunstancia. Nuestro primer objetivo es mantener la integridad y la equidad de la competencia. Lo alentamos a que intente resolver todos los problemas, incluso si su solución no es óptima o incluso completa. Podrá obtener un puntaje parcial resolviendo los casos de prueba simples en cualquier desafío dado.

Todos los desafíos han sufrido un ciclo de desarrollo muy riguroso. Hacemos todo lo posible para garantizar su corrección, pero, en caso de que encuentre un error, tenga en cuenta que los desafíos son los mismos para todos los concursantes. Si ha descubierto un error, ¡puede considerar formar parte de nuestro equipo de Garantía de calidad para futuras competencias!

Un agradecimiento especial a todos los voluntarios que han hecho posible este Xtreme. Nos gustaría expresar nuestro más sincero agradecimiento a nuestros patrocinadores y socios por creer en nuestras ideas y apoyar este gran evento.

Su primera tarea es bastante simple, escríbanos una nota. Escriba un programa que genere texto con sus comentarios y envíelo al sistema. Esta tarea no se puntuará y no es obligatoria, sin embargo, ¡esperamos escuchar de usted!

¡Buena suerte y diviértete!

Entrada estándar

Nada en la entrada estándar.

Salida estándar

Envíe un mensaje de texto con su mensaje.

Restricciones y notas

- Palabras ≤ 200

Entrada estándar

Nada en la entrada estándar.

Salida estándar

Envíe un mensaje de texto con su mensaje.

Restricciones y notas

- Palabras ≤ 200

<https://csacademy.com/ieeextreme13/task/welcome-ieeeextreme-13/>