Torneo Argentino de Programación 2024

24 de agosto de 2024

Cuaderno de Problemas

Información General

Este cuaderno contiene 14 problemas; Las páginas están numeradas de 1 a 19, sin contar esta página. Verifique que su cuaderno está completo.

A) Sobre los nombres de los programas
1) Para soluciones en C/C++ y Python, el nombre del archivo de código fuente no es significativo, puede ser cualquier nombre.
2) Si su solución es en Java, el archivo debe ser llamado: *codigo_de_problema*`.java` donde *codigo_de_problema* es la letra mayúscula que identifica al problema. Recuerde que en Java el nombre de la clase principal debe ser igual que el nombre del archivo.
3) Si su solución es en Kotlin, el archivo debe ser llamado: *codigo_de_problema*`.kt` donde *codigo_de_problema* es la letra mayúscula que identifica al problema. Recuerde que en Kotlin el nombre de la clase principal debe ser llamado igual que el nombre del archivo

B) Sobre la entrada
1) La entrada de su programa debe ser leída de *entrada standard.*
2) La entrada está compuesta de un único caso de prueba, descrito mediante una o más líneas dependiendo del problema.
3) Cuando una línea de entrada contiene varios valores, estos están separados por un único espacio en blanco; la entrada no contiene ningún otro espacio en blanco.
4) Cada línea, incluyendo la última, contiene exactamente un caracter de final-de-línea.
5) El final de la entrada coincide con el final del archivo.

C) Sobre la salida
1) La salida de su programa debe ser escrita en *salida standard.*
2) Cuando una línea de salida contiene varios valores, estos deben ser separados por un único espacio en blanco; la salida no debe contener ningún otro espacio en blanco.
3) Cada línea, incluyendo la última, debe contener exactamente un caracter de final-de-línea.

Problema A
Alto ta-te-ti

El ta-te-ti es un juego de dos jugadores, que se juega sobre un tablero de 3×3 inicialmente vacío, donde los jugadores van alternando turnos. El primer jugador juega con la letra X, y el segundo con la letra O. En su turno, cada jugador debe elegir una casilla vacía del tablero y escribir su letra en ella. El primer jugador que logre formar una línea vertical, horizontal o diagonal con tres ocurrencias de su letra es el ganador. En caso de que no queden casillas vacías en el tablero donde se pueda jugar, el resultado se considera un empate. Las imágenes debajo muestran algunos ejemplos de posibles resultados finales en una partida de ta-te-ti.

O	X	O
O	X	X
X	O	X

Empate

O	X	X
X	X	O
O	X	O

Gana X

X		X
	X	
O	O	O

Gana O

Xavier y Olivia eran fanáticos del ta-te-ti, pero después de un tiempo ambos aprendieron a jugar de forma óptima, y ahora el juego se volvió aburrido porque todas las veces que juegan el resultado es un empate. Pero se les ocurrió una nueva variante del juego, donde se agregan N restricciones antes de comenzar, que se deben cumplir a lo largo del juego.

A cada casilla del tablero se le asigna un número de 1 a 9, como muestra la imagen de abajo, y luego cada restricción se define por dos enteros A y B, que indican que la casilla B del tablero no puede ser utilizada por ninguno de los jugadores si la casilla A se encuentra vacía.

1	2	3
4	5	6
7	8	9

Dadas N restricciones, si Xavier es el primero en jugar y ambos jugadores juegan de forma óptima, determinar quién será el ganador o si el resultado será un empate.

Entrada

Una línea con un entero N ($0 \leq N \leq 10^5$), la cantidad de restricciones.

Luego N líneas, donde la i-ésima de ellas contiene dos enteros A_i y B_i ($1 \leq A_i, B_i \leq 9$), que describen la i-ésima restricción del juego.

Salida

Una línea con una letra que indique el resultado del juego:

- “`X`” para indicar que gana Xavier
- “`O`” para indicar que gana Olivia
- “`E`” para indicar un empate

Ejemplo de entrada 1	**Ejemplo de salida 1**
`0`	`E`

Ejemplo de entrada 2	**Ejemplo de salida 2**
`1` `9 5`	`X`

Ejemplo de entrada 3	**Ejemplo de salida 3**
7 2 2 1 7 3 8 5 9 4 6 6 4 4 6	0

En el tercer ejemplo, si ambos jugadores juegan de forma óptima, el resultado final será el que muestra la tercera imagen del enunciado ("Gana O").

Problema B
Búsqueda del periodo

Adela y Bastian son fanáticos de las adivinanzas, y hace poco se les ocurrió un nuevo juego, pero necesitan de tu ayuda para poder jugarlo.

En este juego, Adela primero elige una palabra secreta $s = s_1 s_2 \ldots s_N$ de longitud N, formada por N letras minúsculas del alfabeto inglés $s_1, s_2, \ldots, s_N$. Bastian conoce la longitud N de la palabra secreta, pero no la palabra en sí.

Para ganar el juego, Bastian debe determinar si la palabra elegida por Adela es periódica o no. Para hacer esto, Bastian puede elegir dos enteros L y R, y realizar la siguiente pregunta: ¿Es $t = s_L s_{L+1} \ldots s_R$ un periodo de s? Adela siempre le va a responder cada pregunta con la verdad.

Adela y Bastian consideran que t es un periodo de una palabra s, si y sólo si s se obtiene concatenando $c \geq 2$ copias de t.

Una palabra s es periódica, si t es un periodo de s para algún t. Por ejemplo: `ab` es el único periodo de `abab`, y tanto `a` como `aa` son periodos de `aaaa`, mientras que por otro lado `ab` y `ababa` no son periódicas.

Para que el juego sea divertido, Adela debe definir un límite a la cantidad de preguntas que Bastian puede realizar. Este límite tiene que depender de la información inicial que Bastian obtiene: el valor de N. En particular, ella quiere que el límite de preguntas K sea el **mínimo** posible, pero de tal forma que Bastian pueda ganar el juego si realiza las K preguntas de forma óptima, teniendo **absoluta certeza** de siempre poder encontrar la respuesta correcta, sin importar qué palabra de longitud N elija Adela.

Ayudalos a que puedan jugar a su juego escribiendo un programa que, dado el valor de N, encuentre el límite de preguntas K que deben usar. Y además, para que Bastian no piense que Adela le hizo trampa en caso de que él pierda, el programa debe indicar un conjunto de K preguntas con las que Bastian podría ganar.

Entrada

Una línea con un entero N ($2 \leq N \leq 10^6$), la longitud de la palabra s que eligió Adela.

Salida

Una línea con el entero K, que indica el límite de preguntas que deben utilizar. Luego K líneas que indican un conjunto de preguntas con el que Bastian podría ganar. La i-ésima línea describe la i-ésima pregunta mediante los enteros L y R ($1 \leq L \leq R \leq N$), de acuerdo a lo explicado anteriormente.

Si existen múltiples conjuntos de preguntas óptimas, cualquiera de ellos será aceptado.

Ejemplo de entrada 1	**Ejemplo de salida 1**
`4`	`1` `1 2`

Ejemplo de entrada 2	**Ejemplo de salida 2**

`6`	`2`
	`3 4`
	`4 6`

Una palabra de longitud 4, $s = s_1 s_2 s_3 s_4$, es periódica si y sólo si $t = s_1 s_2$ es un periodo de s. Por eso, en el primer ejemplo, una pregunta es suficiente para determinar con certeza si s es una palabra periódica o no.

Problema C
Conociendo Ngipto

Un grupo de arqueólogos descubrió recientemente una civilización antigua en Egipto llamada Ngipto. La civilización vivía en un desierto, que se modela como un plano bidimensional infinito. En este desierto, los Ngipcios construyeron una pirámide con base poligonal. Dada una descripción de la pirámide y la posición del sol $(X_{\text{sol}}, Y_{\text{sol}}, Z_{\text{sol}})$, los arqueólogos quieren determinar si hay algún punto en el desierto donde puedan hacer su campamento a resguardo del sol.

Más precisamente, la base de la pirámide consiste en un polígono simple **no necesariamente convexo** formado por N puntos

$$(X_1, Y_1, 0), (X_2, Y_2, 0), \ldots, (X_N, Y_N, 0)$$

Donde las primeras dos coordenadas representan coordenadas cartesianas en el plano del desierto, y la tercera coordenada representa la altura, que para los puntos de la base es 0 pues están en el plano del desierto.

La pirámide también tiene una cúspide que es un punto $(X_{\text{cús}}, Y_{\text{cús}}, Z_{\text{cús}})$ ubicado a una altura $Z_{\text{cús}} > 0$ por encima del plano del desierto. Los puntos de la pirámide son exactamente los que se encuentran en algún segmento con un extremo en el interior o borde del polígono, y el otro extremo en la cúspide.

Se te pide determinar si existe algún punto en el plano del desierto al resguardo del sol, es decir algún punto en el plano del desierto en el exterior del polígono tal que el segmento que lo une al sol interseque la pirámide.

Entrada

Una línea con un entero N ($3 \leq N \leq 1000$), la cantidad de vértices en la base poligonal de la pirámide.

Luego una línea con tres enteros $X_{\text{cús}}, Y_{\text{cús}}, Z_{\text{cús}}$ ($-10^4 \leq X_{\text{cús}}, Y_{\text{cús}} \leq 10^4$, $1 \leq Z_{\text{cús}} \leq 10^4$), las coordenadas de la cúspide de la pirámide.

Luego una línea con tres enteros $X_{\text{sol}}, Y_{\text{sol}}, Z_{\text{sol}}$ ($-10^4 \leq X_{\text{sol}}, Y_{\text{sol}} \leq 10^4$, $Z_{\text{cús}} < Z_{\text{sol}} \leq 10^4$), las coordenadas del sol.

Luego N líneas que describen los vértices de la base poligonal de la pirámide. La i-ésima de estas líneas contiene los enteros X_i, Y_i ($-10^4 \leq X_i, Y_i \leq 10^4$). Estos puntos se dan en sentido horario y forman un polígono simple.

Salida

Una línea con la letra "`S`" si existe algún punto en el plano del desierto al resguardo del sol, o la letra "`N`" en caso contrario.

Ejemplo de entrada 1	Ejemplo de salida 1
4 2 2 4 2 2 6 0 0 0 4 4 4 4 0	N

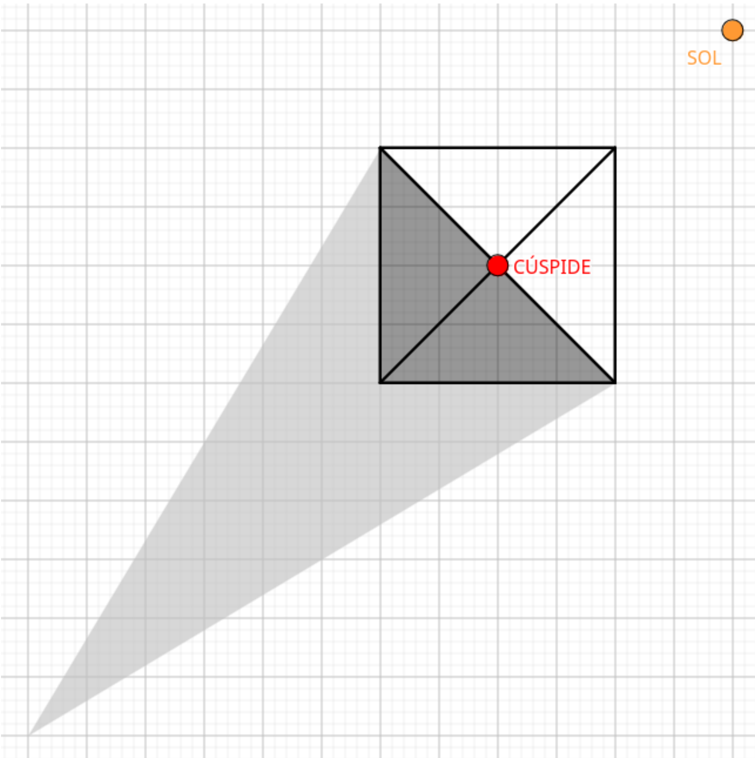
Ejemplo de entrada 2	**Ejemplo de salida 2**

`4`
`2 2 4`
`6 6 6`
`0 0`
`0 4`
`4 4`
`4 0`
	`S`

Ejemplo de entrada 3	**Ejemplo de salida 3**

`4`
`2 2 4`
`2 2 6`
`0 0`
`0 4`
`1 1`
`4 0`
	`S`

En la figura se muestra una vista desde arriba del ejemplo 2. Las partes sombreadas claras son los puntos del desierto donde los arqueólogos pueden acampar.



Problema D
Dupla

Ana, Beto y Carla están jugando un Torneo de Alianzas con Puntos. Cada uno tiene un puntaje, dos de ellos van a aliarse y formar una dupla. Si la suma de los puntos de los aliados supera a los puntos del tercer jugador, gana la dupla. Si no, gana el otro jugador solo.

Conociendo los puntajes de los jugadores, ¿puede alguien ganar solo?

Entrada

Una línea con 3 enteros positivos A, B y C ($1 \leq A, B, C \leq 100$) que indican los puntajes de Ana, Beto y Carla respectivamente.

Salida

Una línea con la letra "`S`" si alguien puede asegurarse ganar solo, o la letra "`N`" en caso contrario.

Ejemplo de entrada 1	**Ejemplo de salida 1**
`1 2 3`	`S`

Ejemplo de entrada 2	**Ejemplo de salida 2**
`4 5 6`	`N`

Ejemplo de entrada 3	**Ejemplo de salida 3**
`100 10 10`	`S`

Ejemplo de entrada 4	**Ejemplo de salida 4**
`50 50 50`	`N`

En el primer ejemplo el jugador que tiene 3 puntos puede ganar solo, ya que la suma de los puntajes de los otros dos jugadores no lo supera.

Problema E
Enfrentamiento final

Dentro de un conocido videojuego, nuestro héroe se enfrenta a “La Bestia”, el jefe final del juego y quiere saber si está en condiciones de derrotarla o no.

El héroe cuenta con P puntos de poder, y dispone de N armas diferentes. Cada arma se caracteriza por tres enteros A, B y C. Al atacar con un arma, si en ese momento el héroe tiene P puntos de poder, pasa a tener $\left\lfloor \frac{P-B}{A} \right\rfloor$ puntos de poder luego del ataque, infligiendo un daño a La Bestia de modo que esta pierde C puntos de vida. Luego de golpear a La Bestia las armas se rompen. Por esta razón, cada arma se puede utilizar como máximo una vez.

El héroe gana el enfrentamiento si logra derrotar a La Bestia. Para lograr esto, sus puntos de poder deben ser no negativos y los puntos de vida de La Bestia tienen que ser negativos o cero.

¿Cuál es la máxima cantidad de puntos de vida V que puede tener La Bestia inicialmente para que el héroe pueda derrotarla?

Nota: $\lfloor x \rfloor$ es el mayor entero tal que $\lfloor x \rfloor \leq x$. Por ejemplo, $\lfloor 2.5 \rfloor = 2, \lfloor \pi \rfloor = 3, \lfloor -2.75 \rfloor = -3$.

Entrada

Una línea con dos enteros N y P ($1 \leq N \leq 200, 1 \leq P \leq 10^5$), la cantidad de armas y la cantidad inicial de puntos de poder del héroe.

Luego N líneas que describen las armas mediante tres enteros A, B y C ($1 \leq A, B, C \leq 10^5$).

Salida

Una línea con un entero V, la máxima cantidad de puntos de vida que puede tener La Bestia.

Ejemplo de entrada 1	Ejemplo de salida 1

`3 66`
`8 8 6`
`8 8 9`
`2 5 2`
	`11`

Ejemplo de entrada 2	Ejemplo de salida 2

`2 10`
`3 1 4`
`2 8 6`
	`10`

Ejemplo de entrada 3	Ejemplo de salida 3
`1 6` `3 7 5`	`0`

En el primer ejemplo, si La Bestia tiene 11 puntos de vida, el héroe puede usar la segunda arma, reduciendo sus puntos de poder de 66 a $\lfloor \frac{66-8}{8} \rfloor = 7$ y los puntos de vida de La Bestia a 2. Luego, puede usar la tercera arma, reduciendo sus puntos de poder de 7 a $\lfloor \frac{7-5}{2} \rfloor = 1$ y los puntos de vida de La Bestia a 0. Si La Bestia inicia con 12 puntos de vida o más, entonces es imposible derrotarla.

En el segundo ejemplo, si La Bestia tiene 10 puntos de vida, el héroe puede usar primero la segunda arma y luego la primera, reduciendo sus puntos de poder a 0 y los puntos de vida de La Bestia a 0.

En el tercer ejemplo, la única forma de que el héroe gane es si La Bestia tiene 0 puntos de vida inicialmente.

Problema F

Fixture

Mati tiene en orden cronológico los resultados de los N partidos que obtuvo durante la última liga de computenis, un deporte individual donde no hay empates, es decir que en cada partido un jugador o bien gana o bien pierde.

La organización de la liga tiene una cierta fascinación por las rachas, y la forma de obtener el puntaje final de un jugador es un tanto peculiar. A un jugador se le suma un punto por cada partido ganado y se le resta un punto por cada partido perdido. Además, si luego de finalizar un partido un jugador lleva al menos 3 partidos ganados consecutivamente incluyendo este último partido, entonces recibe un punto adicional.

¿Podés ayudar a Mati a calcular cuántos puntos obtuvo?

Entrada

Una línea con un entero N ($1 \leq N \leq 100$), la cantidad de partidos que Mati jugó en la liga de computenis.

Luego una línea con N números $R_1, R_2, \ldots, R_N$ ($R_i \in \{0, 1\}$), donde $R_i = 0$ si Mati perdió el i-ésimo partido en orden cronológico y $R_i = 1$ si Mati ganó el i-ésimo partido.

Salida

Una línea con un entero que indica los puntos que obtuvo Mati al finalizar sus partidos de la liga.

Ejemplo de entrada 1	Ejemplo de salida 1
`8` `1 0 1 0 1 0 1 0`	`0`

Ejemplo de entrada 2	Ejemplo de salida 2
`5` `0 0 0 0 0`	`-5`

Ejemplo de entrada 3	Ejemplo de salida 3
`5` `1 1 1 1 1`	`8`

Ejemplo de entrada 4	Ejemplo de salida 4
`7` `1 1 1 0 1 1 1`	`7`

En el primer ejemplo no hay ninguna racha de 3 partidos o más y Mati gana y pierde la misma cantidad de partidos, obteniendo 0 puntos.

En el segundo ejemplo Mati pierde todos los partidos, obteniendo -5 puntos.

En el tercer ejemplo Mati gana todos los partidos obteniendo 8 puntos.

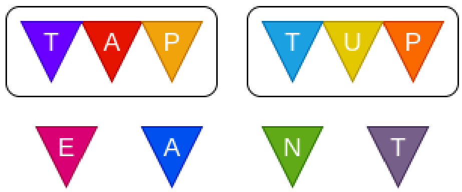
En el cuarto ejemplo Mati obtiene 7 puntos, pues obtiene 2 puntos adicionales por los partidos consecutivos ganados al finalizar el tercer y séptimo partido.

Problema G
Guirnaldas

Lucía está organizando dos competencias muy importantes: el Torneo Argentino de Programación (TAP) y el Torneo Universitario de Pasapalabra (TUP). Estos torneos se realizan de forma simultánea en múltiples sedes.

Para decorar estas sedes, Lucía compró banderines triangulares de papel con letras, como los que se pueden ver en la siguiente imagen:

Lucía se dio cuenta de que puede reordenar los banderines para formar diferentes palabras. Una vez ordenados, planea unir algunos banderines para formar guirnaldas, que se enviarán a las distintas sedes. Todas estas guirnaldas deben tener exactamente tres banderines, o bien formando la palabra `"TAP"` o bien la palabra `"TUP"`. Si una guirnalda no cumple estas características, entonces no puede enviarse a las sedes. A continuación hay un ejemplo de cómo puede hacerlo con los banderines anteriores:



¿Cuál es la máxima cantidad de guirnaldas que puede enviarle a las sedes?

Entrada

Una línea con una cadena S ($3 \leq |S| \leq 300$), las letras de los banderines de Lucía.
La notación $|S|$ denota la cantidad de letras de S.
Se garantiza que todos sus caracteres son letras mayúsculas del alfabeto inglés.

Salida

Una línea con un entero, la máxima cantidad de guirnaldas que puede enviar.

Ejemplo de entrada 1	Ejemplo de salida 1
`APPUNTATTE`	`2`

Ejemplo de entrada 2	Ejemplo de salida 2
`TULIPAN`	`1`

Ejemplo de entrada 3	Ejemplo de salida 3
`TAPTUPTAP`	`3`

Ejemplo de entrada 4	Ejemplo de salida 4
`TOP`	`0`

Las imágenes del enunciado corresponden al primer ejemplo.
En el segundo ejemplo, puede ordenarlas de esta forma: `TUPNILA`.

Problema H

Hilo eléctrico para ganado

En un campo existen N cercos rectangulares, con los lados de los rectángulos alineados a los ejes de coordenadas. Dos cercos distintos nunca se intersecan entre sí, ni siquiera en un punto, y en particular ni siquiera en un vértice. Es posible sin embargo que algunos cercos estén "rodeados" o "cercados" por otros cercos.

Existen M posibles puntos en el campo donde puede aterrizar un paracaidista. Un paracaidista dado siempre tiene la misma probabilidad de aterrizar en cada uno de los M puntos. Más concretamente, en este campo aterrizarán dos paracaidistas, y la posición en la que aterriza cada uno es independiente de la del otro. En otras palabras, todas las $M \times M$ combinaciones de pares de ubicaciones de aterrizaje para el primer y el segundo paracaidista respectivamente son equiprobables. Ninguna de las M ubicaciones se encuentra sobre un cerco electrificado, ni siquiera en un vértice.

Los cercos están hechos de hilo electrificado, que abarca todo el perímetro del rectángulo y por lo cual no es fácil cruzarlos sin recibir un shock eléctrico. Cruzar un cerco significa pasar por el perímetro de su rectángulo correspondiente, es decir, pasar por un punto que pertenece a alguno de los 4 lados del rectángulo. Para cumplir su misión, luego de aterrizar los paracaidistas pueden caminar por el campo hasta encontrarse ambos en un mismo punto del campo. Como cruzar un cerco es muy complicado y peligroso, caminarán de tal forma de atravesar la mínima posible cantidad de cercos hasta encontrarse.

Tu tarea consiste en calcular la cantidad esperada de cercos que deben cruzar en total para poder encontrarse.

Entrada

Una línea con dos enteros N y M ($1 \leq N, M \leq 2 \cdot 10^5$), la cantidad de cercos y la cantidad de puntos de aterrizaje.

Luego N líneas, cada una de las cuales describe uno de los N cercos. Cada línea contiene 4 enteros X_1, Y_1, X_2, Y_2 ($0 \leq X_1 < X_2 \leq 10^9$, $0 \leq Y_1 < Y_2 \leq 10^9$), de manera tal que el cerco tiene esquinas en (X_1, Y_1), (X_1, Y_2), (X_2, Y_1), y (X_2, Y_2).

Luego M líneas, cada una de las cuales describe uno de los M puntos de aterrizaje. Cada línea contiene 2 enteros X, Y ($0 \leq X, Y \leq 10^9$).

Salida

Una única línea con un único número que indica la cantidad esperada de cercos que los paracaidistas deben cruzar en total para poder encontrarse.

Esta respuesta será aceptada si tiene un error relativo o absoluto menor o igual a 10^{-9}.

Formalmente, sea a tu respuesta y b la respuesta del jurado. Entonces, tu respuesta será aceptada si y solo si $\frac{|a-b|}{\max(1,|b|)} \leq 10^{-9}$.

Ejemplo de entrada 1	**Ejemplo de salida 1**

`1 2`
`0 0 10 30`
`1 1`
`0 31`
	`0.5`

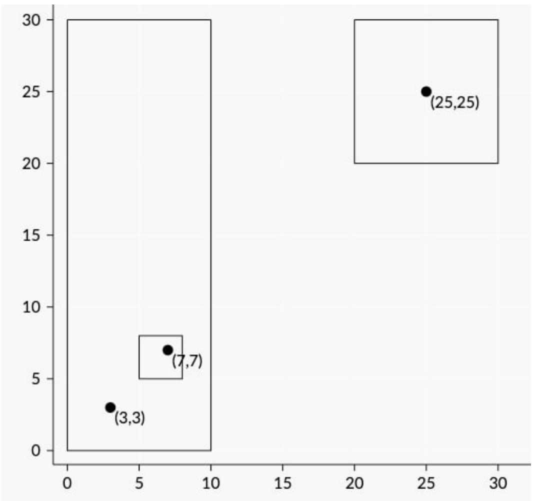
Ejemplo de entrada 2	**Ejemplo de salida 2**

`3 3`
`0 0 10 30`
`5 5 8 8`
`20 20 30 30`
`3 3`
`7 7`
`25 25`
	`1.333333333333`

Ejemplo de entrada 3	**Ejemplo de salida 3**

`1 4`
`10 15 100 200`
`1000 2000`
`3000 4000`
`5000 6000`
`7000 8000`
	`0.0`

La siguiente figura ilustra el segundo ejemplo:



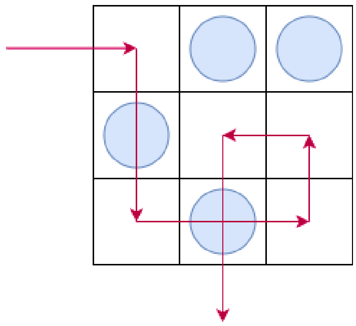
Problema I
Innovaciones robóticas

Franco sigue aprendiendo robótica y está muy contento porque construyó su segundo robot. Su plan original era que este robot fuera capaz de jugar a la Rayuela, pero como esto es muy difícil, se conforma con una versión más simple de este juego.

Sobre el piso hay dibujado un tablero de $N \times M$ casillas. Sea (i, j) la casilla que se encuentra en la fila i y en la columna j. Debido a que estuvo lloviendo mucho últimamente, algunas de estas casillas están mojadas.

El robot puede arrancar su recorrido en cualquier punto exterior al tablero y solamente puede moverse horizontalmente y verticalmente, es decir, en las direcciones paralelas a los bordes del tablero. El robot puede caminar tanto por casillas secas como por casillas mojadas. Sin embargo, para cambiar su dirección de movimiento de vertical a horizontal (o viceversa) el robot debe estar en el interior de una casilla **seca**. El robot puede cambiar su dirección una vez como máximo en cada casilla, pero puede pasar por ella la cantidad de veces que quiera. En ningún momento puede cambiar su dirección en 180°. El recorrido debe terminar en el exterior del tablero.

Para que el recorrido sea válido, el robot tiene que haber cambiado su dirección en **todas** las casillas secas. A continuación se muestra un recorrido válido de un tablero de 3×3, donde las casillas que contienen un círculo son casillas mojadas y las demás son casillas secas:



Se desea determinar algún recorrido válido en caso de existir, para lo cual se debe escribir en cada casilla seca un número entero distinto entre 1 y s (donde s es la cantidad de casillas secas), de modo que los números indiquen ordenadamente los s cambios de dirección que realizará el robot sobre las casillas secas.

Entrada

Una línea con dos enteros N y M ($1 \leq N, M \leq 1000$), la cantidad de filas y columnas del tablero.

Luego, la i-ésima de las siguientes N líneas contiene M caracteres separados entre sí por un espacio. Cada uno de estos es "`.`" o "`0`". Si el caracter en la posición j es "`.`" entonces la casilla (i, j) está seca, y si es "`0`" está mojada.

Se garantiza que hay al menos una casilla seca.

Salida

Si no existe un recorrido válido, una única línea que contenga "`*`".

En caso contrario, N líneas que describan el tablero con los números anotados en las casillas secas.

Formalmente, la i-ésima de las N líneas debe contener M enteros. El j-ésimo de ellos debe ser "`0`" si la casilla (i, j) estaba mojada, y un número entero entre 1 y s si esa casilla estaba seca, como se describe en el enunciado.

Si existen múltiples recorridos válidos, cualquiera de ellos será aceptado.

Ejemplo de entrada 1	**Ejemplo de salida 1**

`3 3`
`. 0 0`
`0 . .`
`. 0 .`
	`1 0 0`
	`0 5 4`
	`2 0 3`

Ejemplo de entrada 2	**Ejemplo de salida 2**

`3 2`
`0 .`
`0 0`
`. .`
	`0 3`
	`0 0`
	`1 2`

Ejemplo de entrada 3	**Ejemplo de salida 3**
`1 3` `. . .`	`*`

Problema J
Jamás suman X

A Máximo le gusta mucho coleccionar peluches. Tiene N peluches y a cada uno de ellos le asignó un valor de belleza, que es un número entero positivo. Él quiere ubicarlos a todos en su estante formando una fila.

Como Máximo cree que el número X trae mala suerte, quiere organizarlos en su estante de forma tal que no haya dos peluches cuya suma de bellezas sea X y que sean adyacentes en la fila.

¿Puede Mámixo lograr su objetivo? Si la respuesta es que sí, se debe indicar cómo puede hacerlo.

Entrada

Una línea con dos enteros N y X ($2 \leq N \leq 3 \cdot 10^5, 1 \leq X \leq 10^9$), la cantidad de peluches que tiene Máximo y el número que cree que trae mala suerte.

Luego una línea con N enteros $B_1, B_2, \ldots, B_N$ ($1 \leq B_i \leq X$), los valores de belleza de cada uno de los peluches de Máximo.

Salida

Si Máximo no puede lograr su objetivo, una única línea que contenga "*".

En caso contrario, una única línea que contenga las bellezas de los N peluches, en el orden en el que Máximo los coloca en su estante.

Si existen múltiples soluciones, cualquiera de ellas será aceptada.

Ejemplo de entrada 1	**Ejemplo de salida 1**
`6 7` `1 2 5 4 6 7`	`1 7 5 4 2 6`

Ejemplo de entrada 2	**Ejemplo de salida 2**
`2 5` `2 3`	`*`

Ejemplo de entrada 3	**Ejemplo de salida 3**
`3 10` `2 6 2`	`2 6 2`

En el primer ejemplo, si Máximo coloca los peluches de forma que las bellezas sean $[1, 7, 5, 4, 2, 6]$ entonces los pares de peluches adyacentes suman $[8, 12, 9, 6, 8]$. Ninguno de estos números es igual a $X = 7$.

En el segundo ejemplo, Máximo no puede lograr su objetivo, ya que las únicas formas de ubicarlos serían $[2, 3]$ y $[3, 2]$ y $2 + 3 = 3 + 2 = 5 = X$.

En el tercer ejemplo, cualquier ordenamiento de los peluches es válido.

Problema K
Kaleidoscopio tipográfico

Se tiene una imagen "ASCII Art" formada por los caracteres '#' y '.', como por ejemplo:

```
.............
.###.###.###.
..#..#.#.#.#.
..#..###.###.
..#..#.#.#...
..#..#.#.#...
.............
```

Se sabe que la imagen está de hecho formada por varias traslaciones de las letras T, A y P que vemos en el dibujo de arriba. Estas letras no pueden estirarse ni escalarse, deben tener exactamente la misma forma que las que se muestran. Cada caracter '#' de la imagen debe pertenecer a exactamente una letra, de modo tal que las letras cubran todos los '#' sin superponerse ni cubrir un '.'.

Por ejemplo, la siguiente imagen:

```
###...
.####.
.##.#.
.####.
.#####
..#.#.
....#.
....#.
....#.
```

contiene dos T's y una P. A modo ilustrativo, cada letra individual se indica como A, B y C a continuación:

```
AAA...
.ABBB.
.AB.B.
.ABBB.
.ABCCC
..B.C.
....C.
....C.
....C.
```

Dada una imagen válida, ¿cuántas T, cuántas A y cuántas P hay?

Entrada

Una línea con dos enteros, N y M ($1 \leq N, M \leq 1000$), el número de filas y columnas. Las siguientes N líneas conforman la imagen, y cada una contiene una cadena de M caracteres.

La imagen dada siempre será válida, es decir, se puede formar con letras T, A, P siguiendo las reglas explicadas en el enunciado.

Salida

Una única línea con tres enteros que indiquen las cantidades de T, A y P.
Si hay más de una terna de valores posible, cualquiera de ellas será aceptada.

Ejemplo de entrada 1

```
7 13
.............
.###.###.###.
..#..#.#.#.#.
..#..###.###.
..#..#.#.#...
..#..#.#.#...
.............
```

Ejemplo de salida 1

```
1 1 1
```

Ejemplo de entrada 2

```
9 6
###...
.####.
.##.#.
.####.
.#####
..#.#.
....#.
....#.
....#.
```

Ejemplo de salida 2

```
2 0 1
```

Ejemplo de entrada 3

```
11 7
.###...
..####.
..##.#.
..####.
..#####
####.#.
.#####.
.##.##.
.#####.
.##.#..
..#.#..
```

Ejemplo de salida 3

```
3 1 1
```

En el primer ejemplo, la imagen contiene exactamente una T, una A y una P.
El segundo ejemplo se explica en el enunciado.

Problema L
Logística lúdica

Agustín y Brian juegan todos los fines de semana muchas partidas del siguiente juego:

Dada una lista de m números enteros positivos $x_1, x_2, \ldots, x_m$, cada jugador comienza con 0 puntos, y alternadamente comenzando por Agustín cada jugador en su turno realiza una de estas dos acciones:

1. Elegir uno de los elementos de la lista que no haya sido elegido anteriormente, y sumar el número de la lista elegido a su puntaje.

2. Pasar el turno.

Si en dos turnos consecutivos ambos jugadores pasan, el juego finaliza. Notar que el hecho de que un jugador no tenga ningún número disponible para elegir en su turno no implica necesariamente que el juego finalice en ese turno.

Cuando finaliza el juego, si un jugador obtuvo un puntaje mayor que el otro, ese jugador resulta ganador de la partida, y si ambos jugadores obtienen el mismo puntaje, entonces el resultado de la partida es un empate.

Ellos están cansados de jugar siempre partidas de este mismo juego, entonces en los últimos fines de semana empezaron a considerar variantes. En particular, este fin de semana decidieron que Agustín solo puede elegir valores que sean una potencia de 2 (es decir, valores x para los que existe un entero no negativo k tal que $x = 2^k$, como 1, 2, 4, 8, ..., etcétera) y Brian solo puede elegir valores impares.

Para no jugar siempre con los mismos números, lo que hacen es jugar Q partidas, cada una con una lista de números que se genera a partir de una lista original de N valores $A_1, A_2, \ldots A_N$. Cada partida se describe con dos números L y R, y se juega con la lista de números que están entre las posiciones L y R de la lista original, es decir $A_L, A_{L+1}, \ldots, A_R$ (incluyendo ambas posiciones extremas).

Dados los N valores de la lista original A_1, A_2, ..., A_N y la descripción de las Q partidas, decidir cuál será el resultado para cada una de las Q partidas si Agustín y Brian juegan de manera óptima.

Entrada

Una línea con dos valores N y Q ($1 \le N, Q \le 2 \cdot 10^5$), que denotan el largo de la lista y la cantidad de partidas que se van a jugar respectivamente.

Una línea con los N valores A_i de la lista original ($1 \le A_i \le 10^4$).

Finalmente siguen Q líneas que describen las partidas. Cada línea describe una partida mediante dos valores L y R ($1 \le L \le R \le N$) que denotan que esa partida utiliza los valores $A_L, A_{L+1}, \ldots, A_R$ de la lista original.

Salida

Q líneas, donde la j-ésima línea denota el resultado de la j-ésima partida, que debe ser "`A`", "`B`" o "`E`" según si al jugar de forma óptima el ganador es Agustín, Brian o la partida termina en empate respectivamente.

Ejemplo de entrada 1	**Ejemplo de salida 1**

`8 3`
`4 2 2 2 3 3 1 6`
`1 3`
`2 6`
`5 8`
	`A`
	`E`
	`B`

En la primera partida del ejemplo, juegan con la lista $[4, 2, 2]$. Jugando de forma óptima, Agustín logra conseguir todos los números, y por lo tanto Agustín resulta ganador de la primera partida.

Problema M
Mercado de globos

Este año Pedro fue invitado a participar en un torneo de programación que se realizará en un país muy muy lejano. Afortunadamente, ya consiguió su pasaje, que incluye N escalas en diferentes países.

Para cubrir los costos del viaje, decidió llevar una valija en la que puede cargar hasta un máximo de K globos que le sobraron de una competencia anterior, con el fin de venderlos en las escalas del viaje. Como conseguir globos en otros países no es tan fácil, Pedro decidió que no va a cargar más globos en su valija en ninguna de las escalas.

Pedro sabe que en el país que visitará en la i-ésima escala puede vender cada globo por V_i pesos, pero hay un problema: Solo puede ingresar con hasta C_i globos sin pagar impuestos. Por cada globo excedente deberá pagar el Impuesto al Comercio de Productos de Carnaval (ICPC), que tiene un costo de P_i pesos por globo excedente.

Pedro quiere saber cuál es la máxima ganancia neta que puede obtener.

Entrada

Una línea con dos enteros N y K ($1 \leq N \leq 2 \cdot 10^5$, $1 \leq K \leq 10^9$), que representan el número de escalas del viaje y la máxima cantidad de globos que Pedro puede llevar en su valija.

Luego una línea con N enteros: $V_1, V_2, \ldots, V_N$ ($0 \leq V_i \leq 10^9$), el precio de venta de un globo en el país que visita en la i-ésima escala, en pesos.

Luego una línea con N enteros: $C_1, C_2, \ldots, C_N$ ($0 \leq C_i \leq 10^9$), la cantidad de globos que se pueden ingresar de manera gratuita en la i-ésima aduana.

Finalmente, una línea con N enteros: $P_1, P_2, \ldots, P_N$, ($0 \leq P_i \leq 10^9$), el impuesto que debe pagar por cada globo excedente al ingresar al i-ésimo país, en pesos.

Salida

Una línea con un entero, la máxima cantidad de dinero (en pesos) que puede ganar Pedro con la venta de globos.

Ejemplo de entrada 1	**Ejemplo de salida 1**
5 6 5 7 8 8 10 2 0 3 0 2 2 2 2 1 2	27

En el ejemplo, Pedro podría cargar en la valija $6 \leq K$ globos, para vender tres en la primera ciudad a un valor de \$5 cada uno, uno en la tercera ciudad por \$8 y los dos restantes en la última por \$10.

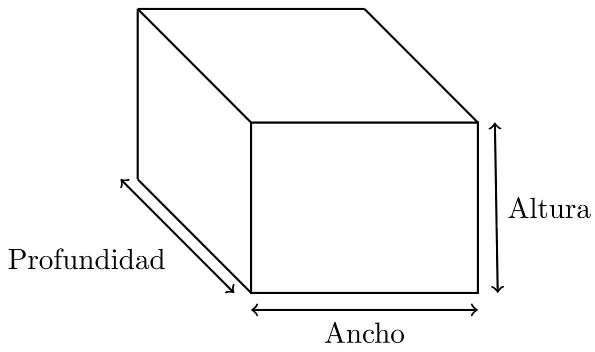
De esta manera, obtendría ingresos brutos por $3 \cdot \$5 + 1 \cdot \$8 + 2 \cdot \$10 = \$43$ y pagaría en total $4 \cdot \$2 + 3 \cdot \$2 + 0 \cdot \$2 + 2 \cdot \$1 + 0 \cdot \$2 = \$16$ en impuestos.

Así tendría $\$43 - \$16 = \$27$ de ganancia neta, que es la máxima posible en este ejemplo.

Problema N

Nuevas dimensiones

Camila vende cajas con forma de paralelepípedos rectangulares, es decir, la forma habitual de una caja.



Si la altura, ancho y profundidad de una caja son a, b y c respectivamente, Camila la vende a un precio de $a^2 + b^2 + c^2$ pesos. Las cajas son huecas, y solo están formadas por la superficie. El material con el que las fabrica cuesta medio peso por unidad cuadrada. Por lo tanto, fabricar una caja cuya altura, ancho y profundidad son a, b y c le cuesta $ab + bc + ca$ pesos.

Camila tiene una lista de N valores posibles para altura, ancho o profundidad de las cajas. Solamente fabrica cajas tales que cada una de sus tres medidas pertenezca a esta lista de N medidas que trabaja. No hay ningún problema en utilizar un mismo valor de la lista para más de una de las tres dimensiones de una caja.

Si elige las medidas a, b, c de una caja de modo de maximizar su ganancia neta (diferencia entre precio de venta y costo), ¿Cuánto puede ganar como máximo en una venta?

Entrada

Una línea con un entero N ($1 \leq N \leq 5000$), el número de valores permitidos para las medidas de la caja.

Luego una línea con N enteros positivos V_i ($1 \leq V_i \leq 10^6$), los valores posibles para la altura, ancho o profundidad. No habrá valores repetidos.

Salida

Una única línea con un único entero, la máxima cantidad de dinero que Camila puede ganar al vender una caja.

Ejemplo de entrada 1	**Ejemplo de salida 1**
`1` `1000000`	`0`

Ejemplo de entrada 2	**Ejemplo de salida 2**
`5` `1734 69384 16 22338 320`	`4811919424`