

Programame

Ctrl Alt Supr

Concurso Regional Programame
Zaragoza
26 de Abril de 2023

Cuaderno de Problemas



NTT DATA

[GO LIVE]

EDELVIVES

KPMG



integra

Índice de problemas

A. CONSTRUYENDO ACRÓNIMOS	3
B. CURIOSA PROPIEDAD LA DEL 9	4
C. DE PINCHOS POR LA LAUREL	5
D. ESPIRAL GALÁCTICA	6
E. SUPERCALIFRAGILISTICOESPIALIDOSO	8
F. UNO QUE VALE POR TODOS	9
G. RESOLVIENDO DERIVADAS SENCILLAS	11
H. LAS SERIES MÁS VISTAS	12
I. HACIENDO LA COMPRA	13
J. HACIENDO CUENTAS	14
K. MIDIENDO LOS ENTRENAMIENTOS	16
L. MANUALIDADES	17

Ejercicios realizados por:

- Antonio Pérez-Aradros
- Santiago Faci

A. CONSTRUYENDO ACRÓNIMOS



Un acrónimo es una palabra que se forma uniendo partes (en muchos casos, las iniciales) de varias palabras y que por el uso acaba frecuentemente por incorporarse al léxico habitual. Ejemplos muy conocidos son: LASER (Light Amplification by Stimulated Emission of Radiation), RENFE (Red nacional de ferrocarriles españoles), SIDA (síndrome de inmunodeficiencia adquirida) o AENA (Aeropuertos españoles y navegación aérea)

Hay también algunos acrónimos sospechosos, por ejemplo, SOS. Popularmente se cree que esta señal significa Save Our Ship (salven nuestro barco), Save Our Souls (salven nuestras almas) o Send Out Succour (envíen socorro). Sin embargo, la mayoría de especialistas afirman que SOS no es el acrónimo de ninguna frase, sino que fue seleccionada por su simplicidad para transmitirla por código morse: una sucesión de tres pulsos cortos, tres largos y otros tres cortos (... - - - ...)

Lo que vamos a hacer nosotros es construir nuestros propios acrónimos. Éstos se van a formar uniendo las iniciales de las palabras siempre y cuando éstas estén en mayúsculas.

Entrada

La entrada comienza con un primer número que indica cuántos casos de prueba deberán ser procesados.

Cada caso de prueba es una frase de no más de 100 letras. En la frase solo aparecerán letras del alfabeto inglés, en mayúsculas o en minúsculas y sin tildes. A veces no se ha tenido mucho cuidado y entre dos palabras puede haber más de un espacio. Se garantiza que todos los acrónimos formados tendrán al menos una letra.

Salida

Para cada caso de prueba se escribirá el acrónimo correspondiente según la regla dada con anterioridad.

Entrada de ejemplo

```
5
Light Amplification by Stimulated Emission of Radiation
Objeto Volador NO Identificado
Aeropuertos Españoles y Navegacion Aerea
Tren Articulado Ligero GOICOECHEA ORIOL
United Nations Educational Scientific and Cultural Organization
```

Salida de ejemplo

```
LASER
OVNI
AENA
TALGO
UNESCO
```

B. CURIOSIDAD PROPIEDAD LA DEL 9

Hoy nos ha contado el profe de Matemáticas una cosa que me ha parecido muy curiosa. Nos ha dicho:

“Pensaos cada uno un número, de cualquier número de cifras. A continuación, formad el mayor y el menor número posible con esas mismas cifras. Si ahora calculáis la diferencia entre el mayor y el menor, resulta que siempre sale un número que es divisible por 9”



Yo he pensado el 34187, así que el mayor y el menor son el 87431 y 13478 respectivamente. Los he restado y sale 73953, que efectivamente es divisible por 9: $73953 = 8217 \times 9$

Aunque ha funcionado como decía, la verdad es que me extraña un poco que salga siempre, así que he hablado con mi hermano, que está estudiando informática, y me va a hacer un programa para comprobarlo.

Entrada

La entrada está compuesta por diferentes casos de prueba.

Cada caso es una línea con un número N, positivo y no mayor que 10^9 , que representa el número que hemos pensado.

La entrada termina cuando N es 0. Este último caso no debe ser procesado.

Salida

Para cada caso de prueba se escribirán las operaciones realizadas para la comprobación. Más concretamente, deben aparecer el número mayor, el número menor, el resultado de su resta y el cociente de dividir entre 9. Delante y detrás de cada operador hay un espacio.

Comprueba los ejemplos de salida para ver cómo es exactamente el formato requerido

Entrada de ejemplo

```
34187
1709876
10
111
0
```

Salida de ejemplo

```
87431 - 13478 = 73953 = 8217 x 9
9877610 - 167789 = 9709821 = 1078869 x 9
10 - 1 = 9 = 1 x 9
111 - 111 = 0 = 0 x 9
```

Como puede observarse en la salida, los ceros no significativos no aparecen

C. DE PINCHOS POR LA LAUREL

Si le preguntas a cualquier persona de Logroño que te recomiende una zona para ir de pinchos, seguro que te habla de la calle Laurel.

Hay numerosas guías con la relación de bares que te puedes encontrar en la zona y las especialidades de cada uno de ellos. Estos bares suelen venir numerados, así que el otro día cuando unos amigos me preguntaron a qué bares podrían ir, yo les di la guía y como no tenía muchas ganas de elegir, les dije que fueran a todos entre el 10 y el 20. Ellos, muy obedientes, con la guía en la mano, fueron recorriendo, sin ningún orden, todos los bares que encontraron entre esos dos números.



A la mañana siguiente no paraban de alabar todo lo que habían tomado y me dijeron los bares a los que habían ido (habían tenido la precaución de ir apuntándolos). En la lista ponía 11, 13, 19, 14, 18, 10, 12, 15, 20 y 16. Se habían dejado uno, el 17, ¡el mejor de todos! ¿Qué hicimos? Por supuesto, fuimos al 17.

Entrada

La entrada consta de una serie de casos de prueba.

Cada caso de prueba es una línea en la que aparecen, separados por espacios, los bares a los que voy entrando. La relación de bares termina con un 0.

La lista no tiene por qué empezar siempre en el 1 y, desde luego, los puedo recorrer en cualquier orden. Lo que sí se asegura es que solo falta un bar y que éste nunca es ni el primero ni el último.

Cada vez hay más gente en la calle Laurel, y cada vez ponen más bares, así que vamos a pensar que puede haber hasta 100000 bares en la guía.

Salida

Para cada caso de prueba se escribirá el bar que falta por visitar.

Entrada de ejemplo

```
11 13 19 14 18 10 12 15 20 16 0
20 21 22 23 24 26 27 28 29 30 0
5 4 3 1 0
10 7 6 3 2 1 8 5 4 0
```

Salida de ejemplo

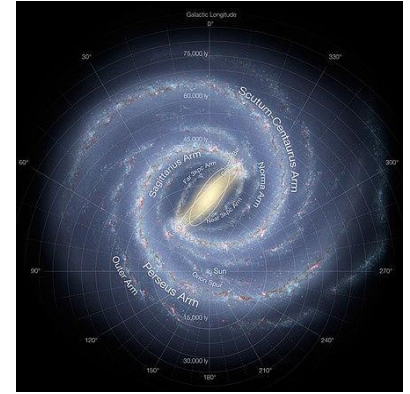
```
17
25
2
9
```

D. ESPIRAL GALÁCTICA

Nuestra galaxia, la Vía Láctea, tiene forma de espiral. Se estima que hay entre cien mil y cuatrocientos mil millones de estrellas en ella.

No te asustes, que no te vamos a mandar contarlas todas.

Resulta que hemos inventado un artilugio que, al enfocarlo al cielo, nos divide la porción de firmamento que vemos en casillas y nos cuenta las estrellas que hay en cada una de ellas. El artilugio se puede regular y podemos elegir el tamaño de la zona de cielo a examinar. Eso sí, esa zona siempre ha de ser cuadrada y con el lado impar.



Solo nos falta programarlo para que sea capaz de contar las estrellas que pertenecen a nuestra Galaxia, es decir, las que están en la espiral. La espiral siempre comienza en la casilla central y sus brazos van avanzando y creciendo en la dirección de las agujas del reloj (arriba, derecha, abajo, izquierda).

2	4	4	4	4
2	3	3	4	5
5	2	2	7	1
2	1	3	5	3
2	4	3	1	2

Supongamos que estamos viendo una porción de firmamento de tamaño 5. Esta porción está dividida en 25 zonas y tenemos anotadas la cantidad de estrellas que hay en cada una de ellas. Se trata de contar las estrellas que pertenecen a la espiral (la zona sombreada). En este caso, 41

2	1	2	2	3	2	3
4	2	4	4	4	4	4
3	2	3	3	4	5	3
4	5	2	2	7	1	2
5	2	1	3	5	3	6
4	2	4	3	1	2	0
3	9	9	9	9	2	9

Si la porción que se ve es de tamaño 7, estaría dividida en 49 zonas y podría ser ésta. En este caso las estrellas serían 54

1	2	4	6	2	3	5	7	0
2	2	1	2	2	3	2	3	1
1	4	2	4	4	4	4	4	1
0	3	2	3	3	4	5	3	2
3	4	5	2	2	7	1	2	3
1	5	2	1	3	5	3	6	1
2	4	2	4	3	1	2	0	2
2	3	9	9	9	9	2	9	3
7	8	4	2	1	0	1	7	5

Si ampliamos la porción a un tamaño 9, la visión podría ser como la de la figura y el recuento nos daría 114

Entrada

Cada caso de prueba son varias líneas. La primera es un número N entre 1 y 100 que indica el tamaño de la porción del Universo que estamos viendo. Recuerda que esta porción siempre es un cuadrado.

A continuación, aparecen N líneas, cada una con N números entre 1 y 10^9 y separados por espacios, indicando el número de estrellas que se ven en esa casilla.

El último caso de prueba es un 0 y no se debe procesar.

Salida

Para cada caso de prueba el programa escribirá el número de estrellas que pueden verse en la espiral galáctica.

Entrada de ejemplo

```
3
3 3 4
2 2 7
1 3 5
5
2 4 4 4 4
2 3 3 4 5
5 2 2 7 1
2 1 3 5 3
2 4 3 1 2
7
2 1 2 2 3 2 3
4 2 4 4 4 4 4
3 2 3 3 4 5 3
4 5 2 2 7 1 2
5 2 1 3 5 3 6
4 2 4 3 1 2 0
3 9 9 9 9 2 9
9
1 2 4 6 2 3 5 7 0
2 2 1 2 2 3 2 3 1
1 4 2 4 4 4 4 4 1
0 3 2 3 3 4 5 3 2
3 4 5 2 2 7 1 2 3
1 5 2 1 3 5 3 6 1
2 4 2 4 3 1 2 0 2
2 3 9 9 9 9 2 9 3
7 8 4 2 1 0 1 7 5
0
```

Salida de ejemplo

```
9
41
54
114
```

E. SUPERCALIFRAGILISTICOESPIALIDOSO

Una de mis películas favoritas de todos los tiempos siempre ha sido Mary Poppins. Me encanta desde el principio hasta el final, pero si tuviera que elegir un solo momento, me quedaría con la canción tras ganar la carrera de caballos. En ella aparece esa palabra mágica que es casi impronunciable y a la vez de las más bonitas que se pueden formar: *SUPERCALIFRAGILISTICOESPIALIDOSO*.



y

La primera vez que la oyes te parece imposible de repetir, sin embargo luego resulta de lo más sencillo. Bueno, por lo menos eso es lo que parece al oír a Julie Andrews cantarla. A mí me ha bastado con 1000 repeticiones para aprendérmela.

Lo que no he podido conseguir es lo que hace ella en la canción: decirla al revés. Anda, prueba a decir *OSODILAI PSEOCITSILIGARFILACREPUS*. Imposible, ¿verdad? Pues ella lo hace, y como si nada.

Pero yo tengo un arma secreta: sé programar en Java, así que voy a hacer un programa que le dé la vuelta a la palabra que yo quiera.

Entrada

La entrada está compuesta por diferentes casos de prueba.

Cada caso es una línea con una sola palabra, como mucho de 50 letras, todas mayúsculas y del alfabeto inglés.

Salida

Para cada caso de prueba se escribirá la palabra inicial invertida, como hace Mary Poppins en la canción.

Entrada de ejemplo

```
SUPERCALIFRAGILISTICOESPIALIDOSO
PERRO
GATO
CONCURSO
```

Salida de ejemplo

```
OSODILAI PSEOCITSILIGARFILACREPUS
ORREP
OTAG
OSRUCNOC
```


F. UNO QUE VALE POR TODOS

Al disponer una serie de números en un cuadrado hay infinidad de reglas que éstos pueden cumplir, dando lugar a diversos cuadrados famosos: cuadrados mágicos, cuadrados diabólicos, cuadrados esotéricos,

A mí se me ocurre uno que no sé si tiene nombre, así que se lo pongo yo: *cuadrado doblado*. Un cuadrado es doblado cuando uno de los números que forma el cuadrado es igual a la suma del resto de números.

1	2	3
4	5	6
7	8	36

Por ejemplo

1	2	3
4	5	6
7	8	36

Es doblado

1	1	1	1
1	1	1	1
1	1	1	1
15	1	1	1

Es doblado

1	1	1	1
1	5	1	1
1	1	1	1
1	1	1	1

NO es doblado

Entrada

La entrada comienza con un primer número que indica cuántos casos de prueba deberán ser procesados.

Cada caso de prueba comienza con una línea con un número N ($1 \leq N \leq 1000$) indicando el lado del cuadrado. A continuación, aparecen N líneas con N números positivos y menores que 10^9 y que corresponden a cada una de las celdas del cuadrado.

Salida

Para cada caso de prueba se escribirá SI si el cuadrado es doblado, según la definición dada anteriormente, o NO en caso contrario.

Entrada de ejemplo

```

4
3
1 2 3
4 5 6
7 8 36
4
1 1 1 1
1 1 1 1
1 1 1 1
15 1 1 1
4
1 1 1 1
1 5 1 1
1 1 1 1
1 1 1 1
5

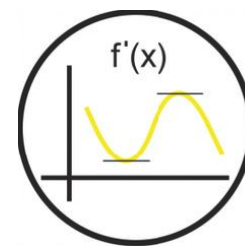
```

```
1 2 3 4 5
6 7 8 9 10
11 12 300 13 14
15 16 17 18 19
20 21 22 23 24
```

Salida de ejemplo

```
SI
SI
NO
SI
```

G. RESOLVIENDO DERIVADAS SENCILLAS



Diana estudia bachillerato y, por tanto, ya se encuentra aprendiendo a hacer derivadas. Como nos ha pasado un poco a todos, aún no sabe muy bien para qué valen, pero tiene que aprender a calcularlas si quiere aprobar Matemáticas.

Practica mucho para sacar buena nota en la asignatura y se ha dado cuenta de que pierde mucho tiempo en comprobar los resultados. Así, como también está aprendiendo a programar, ha decidido hacer un pequeño programa que le permita resolverlas y obtener directamente el resultado. Para comenzar, se va a centrar primero en que el programa resuelva las más sencillas, que son las que corresponden a polinomios sencillos, con la forma $ax^n + \dots + bx^2 + cx + d$ como $3x^3 + 4x^2 + x + 5$

Las reglas necesarias para esta primera versión de su programa son sencillas:

- La derivada de un término con la forma $a \cdot x^n$ será $n \cdot a \cdot x^{n-1}$. Así, la derivada de x^3 es $3x^2$ y la de $3x^2$ sería $6x$
- La derivada del término que no contiene x será 0 y ese valor nunca se indicará en la salida como parte de la solución
- Hay que tener en cuenta que a veces el coeficiente de la x puede ser 1 y cuando eso ocurre no se suele indicar. Lo mismo ocurre cuando está elevado a ese valor. Eso tampoco se indica.

Entrada

Como entrada se recibe una lista de funciones para las que hay que calcular su derivada en base al funcionamiento explicado en el enunciado. Para simplificar la entrada Diana ha decidido emplear el carácter ^ para indicar el exponente de la x cuando éste sea distinto de 1 (valor que se omite).

Puedes considerar que todos los casos de entrada tendrán, al menos, un término con x .

Salida

Como salida se genera la derivada para cada una de las funciones recibidas en la entrada.

Entrada de ejemplo

```
3x^2+2x+3
3x^4+x
5x^4+3x^3+2x^2+10x+4
x
```

Salida de ejemplo

```
6x+2
12x^3+1
20x^3+9x^2+4x+10
1
```

H. LAS SERIES MÁS VISTAS



Netflix está tratando de mejorar su algoritmo de recomendación y para ello necesita enterarse bien de cuáles son nuestras series favoritas.

Es por eso que, desde hace un tiempo, monitorizan el tiempo que cada usuario pasa viendo una serie determinada. Al final del mes procesan esa información y tratan de generar el listado con el top 3 series para dicho usuario. En base a eso tratarán de recomendar series la próxima vez que el usuario se conecte.

A todo esto han añadido un caso especial que consiste en no considerar para este cálculo aquellas series que el usuario pasa menos de 30 minutos viendo. Tras algunos estudios previos aprendieron que es el tiempo mínimo para considerar que alguien está “enganchado” a alguna serie. No parece información relevante a tener en cuenta para el ranking.

Entrada

Como entrada se recibe un listado, en orden cronológico, con todas las series vistas por un usuario y el tiempo (en minutos) que ha pasado viéndolas.

Salida

La salida será el ranking de, cómo máximo, las 3 series más vistas en base al tiempo que el usuario les ha dedicado. El ranking estará ordenado de mayor a menor tiempo de visualización. Lo que podemos asegurar es que siempre habrá, como mínimo, una película a mostrar en ese ranking.

Entrada de ejemplo

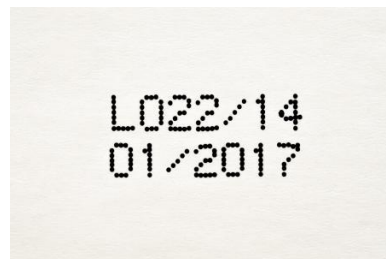
```
Fargo 10
Dexter 20
Chicago PD 20
You 30
Dexter 10
The Wire 100
Chicago PD 25
Dexter 20
You 20
Dexter 20
Fargo 210
Dexter 5
```

Salida de ejemplo

```
Fargo
The Wire
You
```

I. HACIENDO LA COMPRA

Muchas veces, cuando hacemos la compra, nos encontramos con el mismo producto, pero con diferente fecha de caducidad. Es entonces cuando empezamos a rebuscar para llevarnos justo los que caduquen más tarde (que suelen estar escondidos al fondo).



Aunque a veces no queda más remedio y nos tenemos que llevar también los que caducan un poco antes, ya que no tenemos suficiente con los otros. Y así hasta que conseguimos todos los que queremos.

Pero siempre es un poco lío tener que ir rebuscando. Nunca sabes si todavía quedan de los que te durarán más en la nevera o si tienes que empezar ya a buscar de otra fecha. Y para la tienda es otro lío porque acaba todo revuelto y sus trabajadores no saben cuántos quedan para cada fecha por si habría que reponer o retirar de cara a los días siguientes. Hay que revisarlo manualmente, y es por eso que muchos supermercados están tratando de encontrar una forma de automatizarlo y actualizar el conteo cuando el cliente pasa los artículos por caja.

Entrada

Como entrada se recibe, en la primera línea, las unidades que un cliente ha comprado para un producto determinado y, a continuación, cuántas unidades había disponibles de dicho producto, precedida de los días que quedan para que éstas caduquen. Hay que tener en cuenta que este listado no tiene por qué estar ordenado.

Salida

La salida devuelta deberá mostrar cuántas unidades quedan disponibles para cada tiempo de caducidad tras la compra. Como última línea se devolverán las unidades que Alberto no ha podido comprar por no haber suficientes (o 0 cuando consiga comprar todo lo que necesitaba). El sistema aprovecha y devuelve la salida ordenada por el tiempo que queda para que caduque el producto, de más a menos tiempo.

Entrada de ejemplo

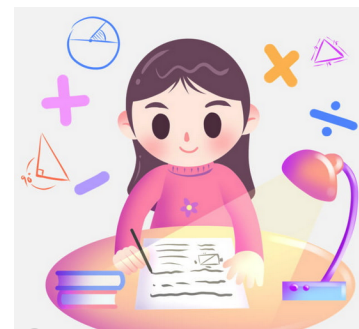
9
2 2
1 2
5 2
3 5

Salida de ejemplo

5 0
3 0
2 0
1 2
0

J. HACIENDO CUENTAS

Diana es una chica muy organizada y cada Navidad organiza sus cuentas para todo el año que viene. Aprovechando las vacaciones escolares dedica algunos ratos a anotar en un cuaderno los ingresos o gastos que cree que tendrá que hacer durante el año siguiente (ir al cine, cenar un día con sus amigos, su paga mensual, el dinero que su tío le da, . .).



Al final resulta que lo apunta todo en una libreta según le viene a la cabeza, por lo que realmente estos movimientos que ella prevé no están del todo ordenados. Cuando se acuerda de un gasto que cree que tendrá para una fecha, lo anota y sigue con lo que estuviera haciendo en ese momento. Antes de acabar el año, su última tarea es organizar toda esa información que ha ido anotando.

El resultado final de todo esto es una hoja donde Diana anota el dinero del que dispondrá a final de cada mes (haya o no haya ingresado o gastado algo dicho mes). Es así como ella se organiza para saber si su saldo siempre será positivo y también para saber si dispondrá de dinero disponible para los gastos no previstos. Por supuesto, su punto de partida es siempre el dinero con el que ya cuenta, que será con el que empiece el nuevo año. Como ella es muy conservadora con los gastos, podemos estar seguros de que su saldo nunca llegará a ser negativo.

Entrada

Como entrada se recibe, en la primera línea la cantidad con la que Diana comienza el año y, a continuación, los movimientos previstos que tiene para todo el año. Para cada movimiento se indica la fecha y si éste es un ingreso o un gasto (indicando + o - junto a la cantidad). Recuerda que no tienen porque estar del todo ordenados.

Salida

Como salida se mostrará el balance de la cuenta de Diana a final de cada mes, independientemente de si ha habido o no movimientos para un mes determinado.

Entrada de ejemplo

```
150
01-01-2023 +30 Regalo tío
01-02-2023 -10 Cenar con mis amigos
05-02-2023 -12 Ir al cine
02-02-2023 +45 Paga
03-03-2023 -12 Ir al cine
04-04-2023 -45 Invitar a mis amigos
05-05-2023 +45 Paga
06-06-2023 -12 Ir al cine
01-09-2023 -30 De compras
01-10-2023 -10 Comprar un libro
01-11-2023 -12 Ir al cine
12-12-2023 -20 Comprar un libro
11-12-2023 -35 Ropa
01-11-2023 -45 Gimnasio
01-10-2023 +45 Paga
01-02-2023 -12 Ir al cine
01-03-2023 +45 Paga
```

Salida de ejemplo

Enero	180
Febrero	191
Marzo	224
Abril	179
Mayo	224
Junio	212
Julio	212
Agosto	212
Septiembre	182
Octubre	217
Noviembre	160
Diciembre	105

K. MIDIENDO LOS ENTRENAMIENTOS

Ya llega el buen tiempo y los ciclistas amateur empiezan a preparar la temporada de cicloturistas. Es tiempo de endurecer los entrenamientos para llegar al verano en plena forma para las carreras importantes.

Antonio ya ha empezado con las series, así que en muchos de sus entrenamientos se marca tramos a diferentes ritmos para mejorar su forma física en las condiciones más duras.



Lleva ya un mes practicando estas series y, a la hora de volcar la información del velocímetro que lleva instalado en su bicicleta, se ha dado cuenta de que los datos están corruptos y ha perdido información. Concretamente ha perdido el valor que le indicaba la distancia recorrida para cada serie de cada entrenamiento. En su lugar aparece el valor 00. Sí que ha conseguido recuperar el tiempo dedicado a cada serie y la velocidad media que mantuvo en cada una, por lo que parece que podrá recuperar la información perdida con un poco de código. Y precisamente, como Antonio es un gran programador, se pone manos a la obra con la solución a este problema.

Entrada

Como entrada se recibe, en la primera línea, el número de entrenamientos que vendrán indicados. A continuación, para cada entrenamiento se reciben los datos para cada una de las series realizadas durante el mismo: la duración (H:mm), un valor 00 donde debería venir indicada la distancia recorrida y, por último, la velocidad media expresada en km/h.

Hay que tener en cuenta que la duración de sus entrenamientos siempre será múltiplo de 30 minutos y que la distancia total recorrida siempre resultará ser un número entero, ya que a Antonio le gustan mucho las cifras redondas y organiza siempre así los entrenamientos.

Salida

Como salida se calculará, para cada entrenamiento, la distancia recorrida en kms.

Entrada de ejemplo

```
6
1:30 00 24 0:30 00 40
2:30 00 20
5:00 00 28 1:30 00 24
1:30 00 34 0:30 00 40 1:00 00 25
3:00 00 30 1:00 00 24
1:00 00 25 0:30 00 40 1:00 00 25 0:30 00 40 1:00 00 25 0:30 00 40
```

Salida de ejemplo

```
56
50
176
96
114
135
```


L. MANUALIDADES

A Diana le gustan mucho las manualidades y últimamente le ha dado por coleccionar los palillos de los helados para hacer marcos de fotos. Los tiene de varios colores y los combina para hacer cuadros donde los palillos verticales (los que quedan a izquierda y derecha de la foto) son de diferente color que los horizontales (los que quedan encima y debajo de la foto).



Para cada foto que quiere enmarcar reúne todos los palillos que tiene disponibles y los junta en grupos de 2 colores diferentes sin repetir color en ningún momento. A continuación comprueba qué grupos de palillos le pueden servir para hacer un marco y que éste encaje con la foto. Finalmente decide cuál le convence más de entre todos los posibles.

A Diana no le importa que el marco de la foto sea más grande que la foto pero, como no le gusta nada que luego le queden palillos sueltos, solamente hace marcos cuando puede aprovechar todos los que ella ha preparado previamente por colores. Y los marcos solamente pueden ser cuadrados o rectángulos perfectos.

Entrada

Al inicio de la entrada se recibe una matriz de caracteres x que representa el tamaño en palillos (una x por cada palillo necesario a lo ancho y alto) de la foto que Diana quiere enmarcar. A continuación, se recibe un número que representa el número de parejas de colores de palillos disponibles. Por último, en cada línea se recibe el número de palillos que hay para las partes de arriba y abajo ($-$), y derecha e izquierda ($|$) del marco (en ese orden).

Salida

Como salida se indicará, para cada uno de los grupos de palillos:

- NO VALE: Cuando no sea posible formar un marco válido usando todos los palillos de ese grupo
- NO CABE: Cuando la foto no quepa en el marco que se podría formar con esos palillos
- CABE JUSTO: Cuando la foto encaje perfectamente con el marco que se podría formar
- CABE PERO SOBRAN n DE ANCHO Y m DE ALTO: Cuando la foto quepa en el marco pero éste sea más grande (n y m representan la anchura y altura sobrantes para cada caso)

Entrada de ejemplo

```
xxxx
xxxx
xxxx
xxxx
5
----- | | | | | | | |
--- --- | | | |
---- ---- | | |
---- ---- | | |
---- ---- | | | | |
```

Salida de ejemplo

```
CABE PERO SOBRAN 1 DE ANCHO Y 1 DE ALTO  
NO CABE  
NO CABE  
NO VALE  
CABE JUSTO
```