



Estado Plurinacional de Bolivia

The 2014 Bolivia Preliminary Contest

ACM - ICPC

Índice

A. Shakespeare	1
B. Shakespeare Tormenta	2
C. Balas	3
D. Area de region	4
E. Nuevas Palabras	5
F. Deudas	7
G. Bloques	8

Las paginas estumeradas desde el 1 al 8.

Los Problem-setters damos un cordial saludo a todos los participantes del evento de programacion mas grande en Bolivia.

El ICPC Bolivia nacio a partir de competidores que poco a poco nos fuimos convirtiendo en organizadores, coachs y problem-setters. Esperamos en un futuro poder contar con su apoyo realizando problemas para eventos a nivel nacional.

Mucho exito en el concurso y que el entrenamiento realizado de frutos.

Problem Setters

- Alvaro Rojas
- Edwin Guzman
- Jorge Teran
- Kevin Bauer
- Mauricio Poppe
- Raul Ronald Zarate Murillo

Problema - A. Shakespeare

Shakespeare tenia un dicho que dice as "mientras mas gotas caen a nuestro alrededor mejor".

En este problema asumimos que las gotas caen en forma uniforme de arriba hacia abajo en linea recta.

Usted tiene una lista de plataformas representadas en un plano cartesiano, con coordenadas $X1$, $X2$ y Y . Estas plataformas inician en $(X1, Y)$ y terminan en $(X2, Y)$ como puede ver forman un segmento a una altura Y .

Este segmento puede retener la lluvia y no deja pasar las gotas a las plataformas por debajo de esta. Necesitamos saber por cada plataforma cuanta agua esta reteniendo.

Entrada

Este problema tiene mltiples casos de entrada.

Cada caso de entrada comienza con un nmero N ($1 \leq N \leq 3$) la cantidad de plataformas.

Cada una de las siguiente N lineas contiene tres nmeros enteros $X1$ ($1 \leq X1 < 1000$), $X2$ ($X1 < X2 \leq 1000$), Y ($1 \leq Y \leq 10^9$) representando a la plataforma.

La entrada termina cuando N es igual a 0 (cero).

Se garantiza que las plataformas no interceptan en ningn punto.

Salida

Por cada plataforma ingresada imprimir la cantidad de agua que esta recibe en unidades.

Ejemplos de entrada	Ejemplos de salida
3	2
1 3 1	4
3 6 2	1
4 7 1	5
2	2
1 5 5	
3 7 2	
0	

Problema - B. Shakespeare Tormenta

Shakespeare tenia un dicho que dice as "El perdn cae como lluvia suave desde el cielo a la tierra. Es dos veces bendito; bendice al que lo da y al que lo recibe".

En este problema asumimos que las gotas caen en forma uniforme de arriba hacia abajo en linea recta.

Usted tiene una lista de plataformas representadas en un plano cartesiano, con coordenadas $X1$, $X2$ y Y . Estas plataformas inician en $(X1, Y)$ y terminan en $(X2, Y)$ como puede ver forman un segmento a una altura Y .

Este segmento puede retener la lluvia y no deja pasar las gotas a las plataformas por debajo de esta. Necesitamos saber por cada plataforma cuanta agua esta reteniendo.

Entrada

Este problema tiene mltiples casos de entrada.

Cada caso de entrada comienza con un nmero N ($1 \leq N \leq 100000$) la cantidad de plataformas.

Cada una de las siguiente N lineas contiene tres nmeros enteros $X1$ ($1 \leq X1 < 10^8$), $X2$ ($X1 < X2 \leq 10^8$), Y ($1 \leq Y \leq 10^9$) representando a la plataforma.

La entrada termina cuando N es igual a 0 (cero).

Se garantiza que las plataformas no interceptan en ningn punto.

Salida

Por cada plataforma ingresada imprimir la cantidad de agua que esta recibe en unidades.

Ejemplos de entrada	Ejemplos de salida
3	2
1 3 1	4
3 6 2	1
4 7 1	5
2	2
1 5 5	
3 7 2	
0	

Problema - C. Balas

Te encuentras en el espacio con una pistola muy especial. Tira balas en una direccija, y como no hay ni gravedad ni friccis balas van en linea recta y con una velocidad constante. Las balas tienen otra particularidad m si dos o me encuentran al mismo tiempo, revientan y desaparecen.

Tiras una bala cada segundo, pero cada bala tiene su propia velocidad. Cuantas balas vuelan por siempre sin reventar nunca?

Entrada

La primera linea contiene el nmero t ($1 \leq t \leq 100$) de casos.

Cada caso empieza con una linea que contiene el nmeros entero n ($1 \leq n \leq 1000$) de balas. Le sigue una linea con n numeros enteros que denominan las velocidades $v_0, v_1, \dots, v_{n-1}$ ($1 \leq v_i \leq 1000$) (en metros por segundo) de las balas que salen a los segundos $0, 1, \dots, n-1$ respectivamente.

Salida

Por cada caso debes poner una linea "Caso X: Y" (sin comillas) en el que X es el nmero de caso (comenzando con 1) e Y es la cantidad de balas que nunca revientan.

Ejemplos de entrada	Ejemplos de salida
4	Caso #1: 2
2	Caso #2: 0
1 1	Caso #3: 3
2	Caso #4: 0
1 2	
3	
3 2 1	
3	
500 600 750	

Problema - D. Area de region

Tienes un pequebot que solo se puede mover hacia una de las cuatro direcciones: este, oeste, norte o sur. Y cada vez se mueve una unidad. Al moverse deja un rastro de su camino recorrido. Al final vuelve a su punto inicial sin jamisar su camino recorrido, con excepcin de su ltimo movimiento. Al mirar el rastro de su recorrido te das cuenta que forma un polgono que encierra una regi su interior. Te preguntas cual ser a de esa regi

Entrada

La primera la de cada entrada es el nmero de casos c ($1 \leq c \leq 100$).

Cada caso consta de dos las. La primera la contiene el nmero m ($2 \leq m \leq 2000$) de movimientos del robot. Y la segunda es una cadena de m caracteres que solo contiene los cuatro caracteres 'e', 'o', 'n' y 's' sealando a los cuatro tipos de movimientos del robot: este, oeste, norte y sur.

Salida

Por cada caso debes poner una la "Caso X: Y" donde X es el nmero de caso (empezando por 1) e Y es el a de la regi

Ejemplos de entrada	Ejemplos de salida
3	Caso #1: 1
4	Caso #2: 2
neso	Caso #3: 5
6	
neesoo	
12	
nonenesesoso	

Problema - E. Nuevas Palabras

Tu hermanita esta aprendiendo a hablar y te das cuenta de que existen algunas palabras que son muy difíciles para que ella las aprenda, al saber que tu hermanita sabe como pronunciar n palabras se te ocurre una idea: *que tal si escojo algunas palabras de mi libro favorito que estn formadas por la concatenacin de 2 de las palabras que mi hermanita sabe? as ella sabra como pronunciar la nueva palabra :)*, tu idea es genial pero tediosa porque tienes que analizar todas las palabras de tu libro favorito as que decidiste hacer un programa que te ayude a encontrar estas palabras, una vez que obtienes una palabra que tu hermanita puede pronunciar se la haces conocer inmediatamente y ella la aprende en ese instante pero debido a que ella es muy pequeña y tiene muy poca memoria *solo puede anadir la nueva palabra que aprendi a partir de otras 2 palabras a la lista de palabras que ya sabe pronunciar si la longitud de la palabra es $\leq 10^2$*

Entrada

Un número n , $0 < n \leq 10^4$ indicando la cantidad de palabras que tu hermanita sabe pronunciar, n líneas siguen cada una con una palabra p que tendr como mximo 10^2 letras (solo se consideran letras minsculas)

Sigue un número q , $0 < q \leq 5 \cdot 10^4$ indicando la cantidad de palabras que estn en tu libro, siguen q líneas, cada línea tendr una palabra $newWord$ que es la palabra que tienes que analizar y decidir si tu hermanita la puede pronunciar, $2 \leq longitud(newWord) \leq 10^2$ (solo se consideran letras minsculas)

Salida

Imprime por cada $newWord$ la cadena SI si es que tu hermanita puede pronunciar la nueva palabra a partir de la concatenacion de otras 2 palabras o NO si es imposible que tu hermanita pueda pronunciar la nueva palabra.

Ejemplos de entrada	Ejemplos de salida
5	SI
ma	SI
pa	NO
ge	NO
la	SI
tina	SI
6	
mapa	
mama	
gato	
gelatina	
latina	
gelatina	

Problema - F. Deudas

Se encontr un problema con las cobranzas en la empresa de telefonos. Se tiene un detalle de los montos que cada persona adeuda por el servicio telefonico. Se perdieron los registros de las personas que pagaron y solo se tiene el monto que falta cobrar. Se tiene un importe que no se cobra y que corresponde a la suma de lo que deben una cantidad de personas. La pregunta es cuales son los grupos de personas que pueden ser los que no pagaron por el servicio?. Adicionalmente se conoce que este importe corresponde a un grupo contiguo de personas.

Veamos un ejemplo:

Persona	0	1	2	3	4	5
Deuda	50	70	30	50	25	25

Si el monto adeudado es de 100 Bs., las posibles respuestas son la suma desde la persona 1 hasta la persona 2 ($70 + 30 = 100$), y la suma desde la persona 3 hasta la 5 ($50 + 25 + 25 = 100$). No puedo considerar la suma de la persona 0 y 3 porque no conforman un rango continuo.

Entrada

La entrada consiste de varios casos de prueba.

Cada caso comienza con un numero entero n ($10 \leq n \leq 100$) cantidad de personas. En la segunda linea existen n numeros d ($2 \leq d \leq 100$) los montos adeudados.

La siguiente linea tiene el numero entero c ($1 \leq c \leq 100$) la cantidad de consultas sobre estos datos. Seguida de los c montos x para las consultas ($1 \leq x \leq 200$).

La entrada termina cuando n es igual a cero.

Salida

Imprima en la salida una linea por cada rango de numeros que den el valor buscado x . Si dicho valor no existe imprima -1 .

Pueden existir multiples rangos por cada valor x imprima todos.

Ejemplos de entrada	Ejemplos de salida
6	1 2
50 70 30 50 25 25	3 5
2	-1
100	
180	

Problema - G. Bloques

Jacinto es un muchacho que le encanta resolver problemas algoritmicos, pero en su tiempo libre le gusta armar torres conformada de varios bloques en diferentes alturas, entonces un da como es de costumbre despues de su sesin diaria algoritmica decidi armar nuevamente las torres, pero al ver las torres de bloques tuvo la idea de calcular la superficie visible de todas las torres. La superficie de un solo bloque es 6 (esto es contando todas las caras del bloque).

Por ejemplo si Jacinto solo arma 1 torre de un solo bloque la superficie ser de 6, ahora si arma 1 torre de altura 2 la respuesta ser 10 (ya que al estar juntos los 2 bloques pierdes 2 caras y entonces como superficie visible solo nos queda 10). Como es predecible Jacinto se aburri de calcular manualmente la superficie de las torres as que decidi hacer un programa que dada las alturas de cada torre en bloques, puedas tu imprimir la superficie de estas.

Entrada

La primera linea de entrada contiene un entero n ($2 \leq n \leq 20$), donde n es el nmero de edificios que Jacinto armara (Solo para condiciones del problema un edificio puede tener altura 0).

A continuacin n enteros indicando la altura de cada torre en bloques. La entrada de datos termina con un entero de valor 0.

Salida

Imprimir la superficie total de todas las torres conformadas por lo bloques.

Ejemplos de entrada	Ejemplos de salida
2	6
1 0	40
10	14
0 2 3 3 2 0 0 0 0 1	
3	
1 1 1	
0	