



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR
CORRESPONDENCIA

CIIC '09

REQUISITOS TÉCNICOS

Nombre del problema	PARCELAS	REUNION	PIZZA	SISEG
<i>Archivo solución</i>	PARCELAS.cpp PARCELAS.c PARCELAS.pas	REUNION.cpp REUNION.c REUNION.pas	PIZZA.cpp PIZZA.c PIZZA.pas	SISEG.cpp SISEG.c SISEG.pas
<i>Archivo de entrada</i>	PARCELAS.ENT	REUNION.ENT	PIZZA.ENT	SISEG.ENT
<i>Archivo de salida</i>	PARCELAS.SAL	REUNION.SAL	PIZZA.SAL	SISEG.SAL
<i>Tiempo de ejecución en una Pentium IV a 2.5 GHz</i>	1 segundo.	1segundo.	1 segundo.	1 segundo.
<i>Máxima cantidad de memoria disponible</i>	64 MB	64 MB	64 MB	64 MB
<i>Puntuación máxima</i>	100	100	100	100



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

Diferencia productiva.

Los concursantes de la CIIC, con el objetivo de vincular el estudio con el trabajo, cuentan con un enorme campo dividido en N parcelas de iguales dimensiones. Cada parcela puede ser identificada con un entero positivo que representa su producción total. De esta manera, es posible modelar el campo mediante un conjunto de N enteros positivos, donde cada elemento representa una parcela.

Debido a complicadas razones de naturaleza técnica, es necesario seleccionar un subconjunto del conjunto original que contenga la mayor cantidad posible de elementos (parcelas), satisfaciendo la restricción de que la diferencia entre el mayor y el menor elemento de dicho subconjunto no sea mayor que un entero positivo D fijado de antemano.

Tarea

Hacer un programa que permita:

- Leer desde el fichero de entrada **PARCELAS.ENT** la cantidad de parcelas, la diferencia máxima permitida y el valor de la producción de cada parcela.
- Determinar cuál es la cantidad máxima de parcelas que pueden ser seleccionada, satisfaciendo la restricción anterior.
- Escribir hacia el fichero de salida **PARCELAS.SAL** dicha cantidad.

Entrada

El fichero de entrada **PARCELAS.ENT** contiene:

Línea 1: Dos enteros N y D (separados por un espacio en blanco) que representan la cantidad de parcelas y la diferencia máxima permitida respectivamente.

Líneas 2.. $N+1$: en la línea $i+1$ aparecerá el valor P_i , que representa el valor de la producción de la i -ésima parcela.

Salida

El fichero de salida **PARCELAS.SAL** debe contener un entero, que represente la cantidad máxima de parcelas que puede ser seleccionada.



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

Ejemplo de entrada y salida

PARCELAS.ENT PARCELAS.SAL

```
9 10
8
1
100
20
1
50
2
10
1000
```

```
5
```

Explicación: Un subconjunto de 5 elementos que cumple la restricción está formado por las producciones {8 , 1 , 10 , 2 , 1}. Como no existen subconjuntos con más de 5 elementos que cumplan la restricción, la respuesta es 5.

Restricciones

$$1 \leq N \leq 10^5$$

$$1 \leq D \leq 10^9$$

$$1 \leq P_i \leq 10^9$$

Para el 50% de los juegos de datos que serán utilizados en la evaluación, se cumple, además, que $1 \leq N \leq 10^3$. Adicionalmente, para el 20% se cumple que $1 \leq N \leq 20$.



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

Reunión

Los concursantes de la CIIC que asistirán a la próxima IOI están probando un sistema de movimientos que les permita reunirse en algún punto y no perderse en ciudades extrañas para ellos. Utilizaremos para la representación de las ciudades un tablero de $N \times N$ casillas donde cada uno de los concursantes está en diferentes casillas. En una unidad de tiempo cada concursante tiene que moverse de cualquiera de las siguientes cuatro maneras:

- b casillas al oeste y b casillas al norte.
- b casillas al este y b casillas al sur.
- b casillas al oeste y b casillas al sur.
- b casillas al este y b casillas al norte.

Nótese que todos los concursantes se mueven a la vez y aunque inician en distintas posiciones pueden eventualmente compartir alguna posición; además está prohibido que un concursante permanezca sin moverse durante una unidad de tiempo.

El tablero está periodificado, esto quiere decir que si un concursante salta de una posición (a, b) , x cuadros al este (x puede ser negativo) y y cuadros al sur (y también puede ser negativo), entonces su posición final será $((a+x) \bmod N, (b+y) \bmod N)$, el operador \bmod representa la división entera entre dos números enteros. Nótemos que $a \bmod b$ es el residuo no negativo de dividir a por b . Es decir, el menor entero no negativo c tal que $a-c$ es múltiplo de b . Por ejemplo: $-1 \bmod 9 = 8$, $-9 \bmod 7 = 5$, $17 \bmod 3 = 2$. En otras palabras como nuestro tablero no tiene posiciones negativas, cuando nos movemos y llegamos a un lado del tablero entonces continuamos el mismo movimiento en el lado opuesto a este.

Los concursantes quieren saber si es posible que ellos puedan reunirse en un mismo lugar después de cierto número de unidades de tiempo, y de ser posible, que dicho número de unidades de tiempo sea el menor.

Tarea

Hacer un programa que permita:

- Leer desde el fichero de entrada **REUNION.ENT** la ubicación de los concursantes en el tablero y la cantidad de casillas que se pueden mover.
- Determinar, si es posible, el mínimo número de unidades de tiempo que los concursantes necesitan para reunirse en un mismo lugar.
- Escribir hacia el fichero de salida **REUNION.SAL** el mínimo de tiempo encontrado o un cartel indicando que es imposible reunirse todos los concursantes en un mismo punto.

Entrada



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

El fichero de entrada **REUNION.ENT** contiene:

Línea 1: Dos enteros: N y b, separados entre sí por un espacio en blanco, los cuales representan respectivamente las dimensiones del tablero y la cantidad de casillas que se pueden desplazarse los concursantes.

Línea 2..N+1: En cada una de ellas aparecerán N dígitos(sin espacios) que pueden ser 1 ó 0 representando posiciones de la cuadrícula, donde 1 significa que hay un estudiante en la casilla y 0 que la casilla está vacía. La esquina superior izquierda del tablero es la (0, 0).

Salida

El fichero de salida **REUNION.SAL** contiene en una sola línea un entero indicando el mínimo número de unidades de tiempo, si no es posible reunir los concursantes se debe imprimir "INFINITO".

Ejemplo de Entrada y Salida

REUNION.ENT

```
3 2
101
000
000
```

REUNION.SAL

```
1
```

Explicación

El estudiante que está en (0, 0) da un salto de la forma (2, -2) llegando a $(2 \bmod 3, -2 \bmod 3) = (2, 1)$ y el que está en (0, 2) da un salto de la forma (2, 2) llegando a $((0+2) \bmod 3, (2+2) \bmod 3) = (2, 1)$

Restricciones

- Para todos los casos $2 \leq N \leq 500$.



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

PIZZA

La capital ideal de la CIIC, puede ser representada como una grilla donde solo es posible desplazarse paralelamente a los ejes X e Y. La compañía CIICPizza quiere un programa para mantener un registro de todas sus pizzerías y además que pueda determinar el tiempo mínimo de entrega para un determinado punto (o sea la pizzería más cercana).

La compañía necesita que su programa sea capaz de procesar órdenes de la siguiente forma:

I x y : agregar una pizzería en las coordenadas enteras (x,y). Inicialmente no hay pizzerías disponibles.

D x : imprimir el tiempo de entrega mínimo (de las pizzerías disponibles) a las coordenadas enteras (x, 0), o sea sobre el eje x.

Tarea

Hacer un programa que permita:

- Leer desde el fichero de entrada **PIZZA.ENT** la cantidad y las órdenes a procesar.
- Determinar el tiempo de entrega mínimo para cada orden D.
- Escribir hacia el fichero de salida **PIZZA.SAL** todos los tiempos mínimos para cada orden D.

Entrada

El fichero de entrada **PIZZA.ENT** contiene:
Línea 1: Un entero O, la cantidad de "órdenes".

Línea 2..O+1: cada una con una orden de acuerdo al formato especificado anteriormente.

Salida

El fichero de salida **PIZZA.SAL** contiene para cada orden **D x**, el tiempo de entrega mínimo. Se garantiza que siempre habrá al menos 1 pizzería disponible.

Restricciones

- $2 \leq O \leq 100000$.
- $0 \leq x, y \leq 100000$.

Para el 30% de los puntos se garantiza que $2 \leq O \leq 2000$.

Ejemplos de Entrada y Salida

PIZZA.ENT

5
I 4 3
D 1
I 2 2
D 1
D 4

PIZZA.SAL

6
3
3



XI COMPETENCIA IBEROAMERICANA DE INFORMÁTICA POR CORRESPONDENCIA

CIIC '09

Línea 1: Dos enteros N y K , separados entre sí por un espacio en blanco, los cuales representan respectivamente la cantidad de nodos del árbol y la cantidad de nodos “rojos” que deben quedar a lo sumo en T .

Línea 2: N valores 1 (azul) ó 2 (rojo) separados entre sí por un espacio en blanco, los cuales representan el color de cada uno de los nodos. El i -ésimo valor representa el color del i -ésimo nodo.

Línea 3: $N-1$ enteros. El i -ésimo entero V_i representa el número del padre del $i+1$ -ésimo nodo.

Salida

El fichero de salida **SISEG.SAL** contiene la cantidad mínima de nodos “azules” que debe ser eliminado.

Restricciones:

$$1 \leq N \leq 100.$$

$$0 \leq K \leq V.$$

$$1 \leq V_i \leq N.$$

Para el 20% de los juegos de datos que serán utilizados en la evaluación, se cumple, además, que $1 \leq N \leq 20$.

Ejemplo de Entada y Salida

SISEG.ENT

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

SISEG.SAL

```
1
```