



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

game

Spanish — 1.0

Bazza y Shazza están jugando un juego. El tablero es una cuadrícula con R renglones numerados $0, \dots, R-1$ y C columnas numeradas $0, \dots, C-1$. Denotamos (P, Q) como la celda en el renglón P y columna Q . Cada casilla del tablero contiene un entero no negativo y al inicio del juego todos los enteros de las casillas valen cero.

El juego procede de la siguiente manera. En cualquier momento, Bazza puede hacer alguna de las siguientes cosas:

- Actualizar la casilla (P, Q) , cambiando el entero que contiene por otro.
- Preguntar a Shazza por el máximo común divisor (GCD, greater common divisor) de todos los enteros adentro de un bloque rectangular de casillas, representado por sus esquinas opuestas (P, Q) y (U, V) inclusive.

Bazza hará a lo más $N_U + N_Q$ acciones (actualizar N_U casillas y hacer N_Q preguntas) antes de que se aburra y se salga a jugar cricket.

Tu tarea es contestar de forma correcta las preguntas.

Ejemplo

Supongamos que $R = 2$, $C = 3$ y Bazza inicia con las siguientes actualizaciones:

- Actualizar la casilla $(0, 0)$ a 20;
- Actualizar la casilla $(0, 2)$ a 15;
- Actualizar la casilla $(1, 1)$ a 12.

20	0	15
0	12	0

Después de las acciones el tablero queda como la imagen de arriba. Bazza puede preguntar después por el máximo común divisor de los siguientes rectángulos:

- Esquina opuesta $(0, 0)$ y $(0, 2)$: Los tres enteros dentro de este rectángulo son 20, 0 y 15 por lo tanto su máximo común divisor es 5.
- Esquinas opuestas $(0, 0)$ y $(1, 1)$: Los cuatro enteros que están adentro de este rectángulo son 20, 0, 0 y 14, por lo tanto el máximo común divisor es 4.

Ahora supongamos que Bazza hace las siguientes actualizaciones:

- Actualizar la casilla $(0, 1)$ a 6;
- Actualizar la casilla $(1, 1)$ a 14.

20	6	15
0	14	0

El nuevo tablero queda como la imagen de arriba. Bazza puede volver a preguntar por el máximo común divisor de los mismos rectángulos:

- Esquina opuesta $(0,0)$ y $(0,2)$. Ahora los tres enteros dentro de este rectángulo son 20, 6 y 15 por lo tanto su máximo común divisor es 1.
- Esquinas opuestas $(0,0)$ y $(1,1)$. Ahora los cuatro enteros que están adentro de este rectángulo son 20, 6, 0 y 14, por lo tanto el máximo común divisor es 2.

En este caso Bazza realizó $N_U = 5$ actualizaciones y $N_Q = 4$ preguntas.

Implementación

Debes mandar un archivo que tenga las funciones `init()`, `update()` y `calculate()` de la siguiente forma:

Para ayudarte, el template que se encuentra en tu computadora (`game.c`, `game.cpp` y `game.pas`) cada uno incluye la función `gcd2(X, Y)` para calcular el máximo común divisor de dos enteros no negativos X e Y . Si $X = Y = 0$ entonces `gcd2(X, Y)` va a devolver `0`.

La función es lo suficiente rápida para obtener todos los puntos; particularmente, el tiempo de ejecución de la función $\text{gcd2}(X + Y)$ en el peor caso es proporcional a $\log(X + Y)$.

Tu función: `init()`

C/C++ `void init(int R, int C);`

Pascal `procedure init(R, C : LongInt);`

Descripción

Tu programa debe tener la siguiente función implementada.

Esta función te da el tamaño inicial del tablero y te permite inicializar variables globales y estructuras de datos. Esta función será llamada una sola vez y siempre antes de cualquier llamada a las funciones `update()` y `calculate()`.

Parámetros

- `R`: El número de renglones.
- `C`: El número de columnas.

Tu función: `update()`

C/C++ `void update(int P, int Q, long long K);`

Pascal `procedure update(P, Q : LongInt; K : Int64);`

Descripción

Tu programa debe tener la siguiente función implementada.

Esta función será llamada cuando Bazza actualice el valor de alguna casilla.

Parámetros

- `P`: El renglón de la casilla ($0 \leq P \leq R - 1$).
- `Q`: La columna de la casilla ($0 \leq Q \leq C - 1$).
- `K`: El nuevo valor de la casilla ($0 \leq K \leq 10^{18}$). Este valor podría ser el mismo que el valor actual.

Tu Función: `calculate()`

C/C++ `long long calculate(int P, int Q, int U, int V);`

Pascal `function calculate(P, Q, U, V : LongInt) : Int64;`

Descripción

Tu envío deberá implementar esta función.

Esta función debe calcular el máximo común divisor de todos los enteros que se encuentran adentro del rectángulo con equinas opuestas (P, Q) y (U, V) . Este rango es inclusivo, es decir, las casillas (P, Q) y (U, V) están incluidas en el rectángulo.

Si el valor de todos los enteros que se encuentran adentro del rectángulo es cero entonces esta función debe regresar cero.

Parámetros

- P : El renglón de la casilla en la esquina superior-izquierda del rectángulo ($0 \leq P \leq R - 1$).
- Q : La columna de la casilla en la esquina superior-izquierda del rectángulo ($0 \leq Q \leq C - 1$).
- U : El renglón de la casilla en la esquina inferior-derecha del rectángulo ($P \leq U \leq R - 1$).
- V : La columna de la casilla en la esquina inferior-derecha del rectángulo ($Q \leq V \leq C - 1$).
- *Returns*: El máximo común divisor de todos los enteros que se encuentran adentro del rectángulo, o 0 si el valor de todos esos enteros es cero.

Sesión de ejemplo

La siguiente sesión describe el ejemplo mencionado anteriormente:

Llamada de función	Regresa
<code>init(2, 3)</code>	
<code>update(0, 0, 20)</code>	
<code>update(0, 2, 15)</code>	
<code>update(1, 1, 12)</code>	
<code>calculate(0, 0, 0, 2)</code>	5
<code>calculate(0, 0, 1, 1)</code>	4
<code>update(0, 1, 6)</code>	
<code>update(1, 1, 14)</code>	
<code>calculate(0, 0, 0, 2)</code>	1
<code>calculate(0, 0, 1, 1)</code>	2

Restricciones

- Tiempo límite: Ver sub-tareas.
- Límite de memoria: Ver sub-tareas.
- $1 \leq R, C \leq 10^9$
- $0 \leq K \leq 10^{18}$, donde K es cualquier entero que Bazza puede poner en el tablero.

Subtareas

Ver la versión en inglés para los parámetros de los sub-problemas.

Sub-problemas	Puntos	R	C	N_U	N_Q	Límite de tiempo	Límite de memoria

Experimentación

El evaluador de ejemplo de tu computadora leerá el archivo de entrada `game.in`. Este archivo debe tener el siguiente formato:

- línea 1: `R C N`
- siguientes `N` líneas: una acción por línea, en el orden que las acciones ocurren

La línea para cada acción debe estar en alguno de los siguientes formatos:

- para indicar `update(P, Q, K) : 1 P Q K`
- para indicar `calculate(P, Q, U, V) : 2 P Q U V`

Por ejemplo, el caso de arriba debiera ser provisto en el siguiente formato:

```
2 3 9
1 0 0 20
1 0 2 15
1 1 1 12
2 0 0 0 2
2 0 0 1 1
1 0 1 6
1 1 1 14
2 0 0 0 2
2 0 0 1 1
```

Notas de lenguaje

C/C++ Debes incluir `#include "game.h"`.

Pascal Debes definir `unit Gamr`. Todos los arreglos están numerados desde `0` (no `1`).

Como los enteros en cada casilla pueden ser muy largos, se aconseja que los usuarios de C/C++ usen `long long`, y los usuarios de Pascal usen el tipo `Int64`.