IOI '13 P6 - Game

Bazza and Shazza are playing a game. The board is a grid of cells, with R rows number $0, \ldots, R-1$, and C columns numbered $0, \ldots, C-1$. We let (P, Q) denote the cell in row P and column Q. Each cell contains a non-negative integer, and at the beginning of the game all of these integers are zero.

The game proceeds as follows. At any time, Bazza may either:

- update a cell (P, Q), by assigning the integer that it contains;
- ask Shazza to calculate the greatest common divisor (GCD) of all integers within a rectangular block of cells, with opposite corners (P, Q) and (U, V) inclusive.

Bazza will take $N_U + N_Q$ actions (updating cells N_U times and asking question N_Q times) before he gets bored and goes outside to play cricket.

Your task is to work out the correct answers.

Example

Suppose $R = 2$ and $C = 3$, and Bazza begins with the following updates:

- Update cell $(0, 0)$ to 20;
- Update cell $(0, 2)$ to 15;
- Update cell $(1, 1)$ to 12.

The resulting grid is shown in the picture above. Bazza might then ask for GCDs in the following rectangles:

- Opposite corners $(0, 0)$ and $(0, 2)$: The three integers in this rectangle are 20, 0 and 15, and their GCD is 5.
- Opposite corners $(0, 0)$ and $(1, 1)$: The four integers in this rectangle are 20, 0, 0, and 12, and their GCD is 4.

Now suppose Bazza makes the following updates:

- Update cell $(0, 1)$ to 6;
- Update cell $(1, 1)$ to 14.

The new grid is shown in the picture above. Bazza might then ask for GCDs in the following rectangles again:

- Opposite corners $(0,0)$ and $(0,2)$: Now the three integers in this rectangle are 20, 6 and 15, and their GCD is 1.
- Opposite corners $(0,0)$ and $(1,1)$: Now the four integers in this rectangle are 20, 6, 0 and 14, and their GCD is 2.

Here Bazza has preformed $N_U = 5$ updates and $N_Q = 4$ questions.

Implementation

You should submit a file implementing the procedures `init()` and `update()` and the function `calculate()`, and described below.

Your procedure: `init()`

C/C++

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

Pascal

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

Description

> Your submission must implement this procedure.
>
> This procedure give you the initial size of the grid, and allows you to initialise any global variables and data structures. It will only be called once, before any calls to `update()` or `calculate()`.

Parameters

- `R` : The number of rows.
- `C` : The number of columns.

Your Procedure: `update()`

C/C++

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

Pascal

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

Description

> Your submission must implement this procedure.
>
> This procedure will be called when Bazza assigns the number in some grid cell.

Parameters

- `P` : The row of the grid cell $(0 \leq P \leq R - 1)$
- `Q` : The column of the grid cell $(0 \leq Q \leq C - 1)$
- `K` : The new integer in this grid cell $(0 \leq K \leq 10^{18})$. May be the same as the current value.

Your Function: `calculate()`

C/C++

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

Pascal

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

Description

> Your submission must implement this function.
>
> This function should calculate the greatest common divisor of all integers in the rectangle with opposite corners (P, Q) and (U, V). This range is inclusive, i.e., the cells $(P, Q$P and (U, V) are included in the rectangle.
>
> If all the integers in this rectangle are zero, then this function should return zero also.

Parameters

- `P` : The row of the top-left cell in the rectangle $(0 \leq P \leq R - 1)$

- $\boxed{Q}$: The column of the top-left cell in the rectangle $(0 \leq Q \leq C - 1)$
- $\boxed{U}$: The row of the bottom-right cell in the rectangle $(P \leq U \leq R - 1)$.
- $\boxed{V}$: The column of the bottom0right cell in the rectangle $(Q \leq V \leq C - 1)$
- *Returns*: The GCD of all integers in the rectangle, or 0 if all those integers are zero.

Sample Session

The following session describes the example above:

Function Call	Returns
init(2, 3)	
update(0, 0, 20)	
update(0, 2, 15)	
update(1, 1, 12)	
calculate(0, 0, 0, 2)	5
calculate(0, 0, 1, 1)	4
update(0, 1, 6)	
update(1, 1, 14)	
calculate(0, 0, 0, 2)	1
calculate(0, 0, 1, 1)	2

Constraints

- $1 \leq R, C \leq 10^9$
- $0 \leq K \leq 10^{18}$, where K is any integer that Bazza places in a grid cell.

Subtasks

Subtask	Points	R	C	N_U	N_Q
1	10	≤ 100	≤ 100	≤ 100	≤ 100
2	27	≤ 10	$\leq 100\,000$	$\leq 10\,000$	$\leq 250\,000$
3	26	$\leq 2\,000$	$\leq 2\,000$	$\leq 10\,000$	$\leq 250\,000$
4	17	$\leq 10^9$	$\leq 10^9$	$\leq 10\,000$	$\leq 250\,000$

| 5 | 20 | $\leq 10^9$ | $\leq 10^9$ | $\leq 22\,000$ | $\leq 250\,000$ |