



## International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

# game

Norwegian — 1.1

Bazza og Shazza spiller et spill. Brettet er et rutenett av celler, med  $R$  rader som er nummerert  $0, \dots, R - 1$ , og  $C$  kolonner som er nummerert  $0, \dots, C - 1$ . Vi lar  $(p, q)$  betegne cellen i rad  $p$  og kolonne  $q$ . Hver celle inneholder et ikke-negativt heltall, og ved starten av spillet er alle disse heltallene 0.

Spillet foregår som følger. Når som helst kan Bazza enten:

- Oppdatere en celle  $(p, q)$  ved å endre tallet den inneholder
- Be Shazza om å regne ut den største fellesnevneren (greatest common divisor / GCD) av alle heltallene innenfor en rektangulær blokk av celler, med motstående hjørner  $(p, q)$  og  $(u, v)$  (som begge er med i blokken)

Bazza vil utføre inntil  $N_u + N_q$  handlinger ( $N_u$  oppdateringer av celler og  $N_q$  stillinger av spørsmål) før han begynner å kjede seg og går ut for å spille cricket.

Oppgaven din er å finne ut de rette svarene.

### Eksempel

Anta at  $R = 2$  og  $C = 3$ , og at Bazza begynner med de følgende oppdateringene:

- Oppdater celle  $(0, 0)$  til 20;
- Oppdater celle  $(0, 2)$  til 15;
- Oppdater celle  $(1, 1)$  til 12.

20	0	15
0	12	0

Det resulterende rutenettet vises i bildet ovenfor. Bazza kan da komme til å spørre etter GCD'ene i de følgende rektanglene:

- Motstående hjørner  $(0, 0)$  og  $(0, 2)$ : De tre heltallene i dette rektangelet er 20, 0 og 15, og GCD'en deres er 5.
- Motstående hjørner  $(0, 0)$  og  $(1, 1)$ : De fire heltallene i dette rektangelet er 20, 0, 0 og 12, og GCD'en deres er 4.

La oss nå si at Bazza gjør de følgende oppdateringene:

- Oppdater celle  $(0, 1)$  til 6;
- Oppdater celle  $(1, 1)$  til 14.

20	6	15
0	14	0

Det nye rutenettet vises i bildet ovenfor. Bazza kan da komme til å spørre etter GCD'ene i de følgende rektanglene:

- Motstående hjørner  $(0, 0)$  og  $(0, 2)$ : Nå er de tre heltallene i dette rektangelet 20, 6 og 15, og GCD'en deres er 1.
- Motstående hjørner  $(0, 0)$  og  $(1, 1)$ : Nå er de fire heltallene i dette rektangelet 20, 6, 0 og 14, og GCD'en deres er 2.

Her har Bazza utført totalt  $N_U = 5$  oppdateringer og stilt  $N_Q = 4$  spørsmål.

---

## Implementasjon

Du skal sende inn en fil som implementerer funksjonene `init()`, `update()` og `calculate()` som beskrevet nedenfor.

For å hjelpe deg inneholder alle løsningsmalene på maskinen din (`game.c`, `game.cpp` og `game.pas`) en funksjon `gcd2(X, Y)` for å regne ut den største fellesnevneren av to gitte ikke-negative heltall  $X$  og  $Y$ . Hvis  $X = Y = 0$  vil `gcd2(X, Y)` returnere 0.

Denne funksjonen er rask nok til å få full poengsum; mer spesifikt er kjøretiden i verste fall proporsjonal med `log(X + Y)`.

## Din funksjon: `init()`

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

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

### Beskrivelse

Besvarelsen din må implementere denne funksjonen.

Denne funksjonen gir deg den initielle størrelsen til rutenettet, og lar deg initialisere globale variabler og datastrukturer som du trenger. Den vil bli kalt bare én gang, før alle kall til `update()` og `calculate()`.

### Parametre

- `R`: Antall rader.
- `C`: Antall kolonner.

## Din funksjon: `update()`

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

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

### Beskrivelse

Besvarelsen din må implementere denne funksjonen.

Denne funksjonen vil bli kalt når Bazza endrer tallet i en celle.

### Parametre

- `P`: Raden til cellen ( $0 \leq P \leq R - 1$ ).
- `Q`: Kolonnen til cellen ( $0 \leq Q \leq C - 1$ ).
- `K`: Det nye heltallet i denne cellen ( $0 \leq K \leq 10^{18}$ ). Det kan være den samme verdien som den som er i cellen fra før av.

## Din funksjon: `calculate()`

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

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

### Beskrivelse

Besvarelsen din må implementere denne funksjonen.

Denne funksjonen skal beregne største fellesnevner for alle heltallene i rektangelet med de motstående hjørnene  $(P, Q)$  og  $(U, V)$ . Dette området er inklusivt, dvs. at cellene  $(P, Q)$  og  $(U, V)$  er med i rektangelet.

Hvis alle heltallene i dette rektangelet er 0, skal denne funksjonen returnere 0.

### Parametre

- $P$ : Raden til øverste venstre celle i rektangelet ( $0 \leq P \leq R - 1$ ).
- $Q$ : Kolonnen til øverste venstre celle i rektangelet ( $0 \leq Q \leq C - 1$ ).
- $U$ : Raden til nederste høyre celle i rektangelet ( $P \leq U \leq R - 1$ ).
- $V$ : Kolonnen til nederste høyre celle i rektangelet ( $Q \leq V \leq C - 1$ ).
- *Returnerer*: GCD av alle heltallene i rektangelet, eller 0 hvis alle de heltallene er 0.

---

## Sample Session

The following session describes the example above:

Function Call	Returns
<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

---

## Constraints

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

---

## Subtasks

See English version for subtask parameters.

Subtask	Points	R	C	$N_U$	$N_Q$	Time limit	Memory limit

---

## Experimentation

The sample grader on your computer will read input from the file `game.in`. This file must be in the following format:

- line 1: `R C N`
- next `N` lines: one action per line, in the order in which actions occur

The line for each action must be in one of the following formats:

- to indicate `update(P, Q, K)`: `1 P Q K`
- to indicate `calculate(P, Q, U, V)`: `2 P Q U V`

For instance, the example above should be provided in the following format:

```
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
```

---

## Language Notes

**C/C++** You must `#include "game.h"`.

**Pascal** You must define the `unit Game`. All arrays are numbered beginning at `0` (not `1`).

Because the integers in grid cells could be very large, C/C++ users are advised to use the `long long` type, and Pascal users are advised to use the `Int64` type.