



International Olympiad in Informatics 2013

6-13 July 2013

Brisbane, Australia

Day 2 tasks

game

Latvian — 1.1

Baize un Šaize spēlē spēli uz milzīga rūtiņu laukuma ar R rindām numurētām $0, \dots, R-1$, un C kolonām numurētām $0, \dots, C-1$. (P, Q) apzīmē rūtiņu P . rindā un Q . kolonā. Katra rūtiņa satur nenegatīvu veselu skaitli, un spēles sākumā visās rūtiņās ir nulles.

Spēle notiek sekojoši: jebkurā brīdī Baize var izdarīt vienu no diviem gājieniem:

- ierakstīt rūtiņā (P, Q) skaitli;
- paprasīt Šaizei aprēķināt lielāko kopīgo dalītāju (LKD) visiem skaitļiem rūtiņu taisnstūrī, kura pretējo stūru rūtiņas ir (P, Q) un (U, V) (arī šīs rūtiņas pieder taisnstūrim).

Baize izdarīs $N_U + N_Q$ darbības (N_U reizes rūtiņās ierakstot skaitļus un N_Q reizes prasot aprēķināt LKD).

Jūsu uzdevums ir noskaidrot atbildes uz jautājumiem.

Piemērs

Ja $R = 2$, $C = 3$ un Baize sāk ar šādu skaitļu ierakstīšanu:

- rūtiņā $(0, 0)$ skaitli 20;
- rūtiņā $(0, 2)$ skaitli 15;
- rūtiņā $(1, 1)$ skaitli 12.

20	0	15
0	12	0

Laukuma rūtiņu saturs ir parādīts augstāk esošā attēlā. Tad Baize var pajautāt LKD sekojošos blokos:

- Pretējie stūri ir $(0, 0)$ un $(0, 2)$: trīs skaitļi šajā taisnstūrī ir 20, 0 un 15; to LKD ir 5.
- Pretējie stūri ir $(0, 0)$ un $(1, 1)$: četri skaitļi šajā taisnstūrī ir 20, 0, 0, 12; to LKD ir 4.

Tagad Baize rūtiņās ieraksta šādus skaitļus:

- rūtiņā $(0, 1)$ skaitli 6;
- rūtiņā $(1, 1)$ skaitli 14.

20	6	15
0	14	0

Rūtiņu saturs ir parādīts augstāk esošajā attēlā. Tad Baize var atkal pajautāt LKD sekojošos blokos:

- Pretējie stūri ir $(0, 0)$ un $(0, 2)$: tagad trīs skaitļi šajā taisnstūrī ir 20, 6 un 15; to LKD ir 1.
- Pretējie stūri ir $(0, 0)$ un $(1, 1)$: tagad četri skaitļi šajā taisnstūrī ir 20, 6, 0, 14; to LKD ir 2.

Šeit Baize izpildīja $N_U = 5$ skaitļu ierakstīšanas darbības un uzdeva $N_Q = 4$ jautājumus.

Implementācija

Jums jāiesūta fails, kurā implementētas procedūras `init()` un `update()`, kā arī funkcija `calculate()`, kā aprakstīts tālāk:

Lai jums palīdzētu, uz jūsu datora dotajos šablona risinājumos (`game.c`, `game.cpp` un `game.pas`) katrā ir iekļauta funkcija `gcd2(X, Y)` divu nenegatīvu veselu skaitļu X un Y lielākā kopīgā dalītāja aprēķināšanai. Ja $X = Y = 0$, tad `gcd2(X, Y)` rezultāts arī būs 0.

Šī funkcija ir pietiekami ātra, lai dabūtu visus punktus; precīzāk, izpildes laiks sliktākajā gadījumā ir propocionāls $\log(X + Y)$.

Jūsu procedūra: `init()`

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

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

Apraksts

Jūsu iesūtījumam jārealizē šī procedūra.

Šī procedūra parametros saņem laukuma izmērus, un jūs varat inicializēt jebkurus globālos mainīgos un datu struktūras. Tā tiks izsaukta tikai vienu reizi, pirms jebkādiem `update()` vai `calculate()` izsaukumiem.

Parametri

- `R` : Rindu skaits.
- `C` : Kolonu skaits.

Jūsu procedūra: `update()`

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

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

Apraksts

Jūsu iesūtījumam jārealizē šī procedūra.

Šī procedūra tiks izsaukta, kad Baize kādā laukuma rūtiņā ierakstīs skaitli.

Parametri

- `P` : Rūtiņas rinda ($0 \leq P \leq R - 1$).
- `Q` : Rūtiņas kolona ($0 \leq Q \leq C - 1$).
- `K` : Šajā rūtiņā ierakstāmais skaitlis ($0 \leq K \leq 10^{18}$), kas var sakrist ar jau ierakstīto skaitli.

Jūsu funkcija: `calculate()`

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

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

Apraksts

Jūsu iesūtījumam jārealizē šī funkcija.

Šai funkcijai jānoskaidro visu skaitļu taisnstūrī ar pretējiem stūriem (P, Q) un (U, V) lielākais kopīgais dalītājs. Šis diapazons ir iekļaujošs, t.i. rūtīņas (P, Q) un (U, V) ir iekļautas taisnstūrī.

Ja visi skaitļi šajā taisnstūrī ir nulles, tad funkcijai arī jāatgriež nulle.

Parametri

- P : Rūtīņas rinda taisnstūra augšējā kreisajā stūrī ($0 \leq P \leq R - 1$).
- Q : Rūtīņas kolona taisnstūra augšējā kreisajā stūrī ($0 \leq Q \leq C - 1$).
- U : Rūtīņas rinda taisnstūra apakšējā labajā stūrī ($0 \leq U \leq R - 1$).
- V : Rūtīņas kolona taisnstūra apakšējā labajā stūrī ($0 \leq V \leq C - 1$).
- *Rezultāts*: Visu skaitļu taisnstūrī LKD, vai 0 ja visi tie skaitļi ir nulles.

Piemēra sesija

Augstāk doto piemēru apraksta šāda sesija:

Funkcijas izsaukums	Rezultāts
<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

Ierobežojumi

[Ierobežojumus skatīt angļiskajā versijā!](#)

Apakšuzdevumi

[Apakšuzdevumu parametrus skatīt angļiskajā versijā!](#)

Eksperimentēšana

Piemēra testētājs uz jūsu datora lasīs ievaddatus no faila `game.in`. Šim failam jābūt šādā formātā:

- 1. rinda: `R C N`
- Nākamās `N` rindas: pa vienam gājienam katrā rindā tādā secībā kā tie tiek veikti

Katrai vienu gājienu aprakstošai rindai jābūt vienā no šādiem formātiem:

- lai aprakstītu `update(P, Q, K) : 1 P Q K`
- lai aprakstītu `calculate(P, Q, U, V) : 2 P Q U V`

Piemēram, augstāk esošais piemērs jāformatē šādi:

```
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
```

Piezīmes par valodām

C/C++ Jums jāiekļauj `#include "game.h"`.

Pascal Jums jādefinē `unit Game`. Visi masīvi tiek numurēti sākot no `0` (nevis `1`).

Tā kā veselo skaitļu vērtības var būt ļoti lielas, C/C++ lietotājiem tiek ieteikts izmantot `long long`, bet Pascal lietotājiem - `Int64` tipa mainīgos.