International Olympiad in Informatics 2013



6-13 July 2013 Brisbane, Australia Day 2 tasks

game

Bosanski — 1.1

Bazza i Shazza igraju jednu igru. Ploča je organizovana u obliku tabele sa poljima raspoređenim u R redova označenih brojevima [0, ..., R - 1], i C kolona označenih brojevima [0, ..., C - 1]. (P, Q) označava polje u redu P i koloni Q na ploči. Svako polje sadrži nenegativan cio broj. Na početku igre svi ovi brojevi jednaki su 0.

Igra se izvodi kako slijedi. U svakom trenutku, Bazza može odigrati jedan od sljedeća dva poteza:

- promijeniti cjelobrojnu vrijednost polja (P, Q);
- tražiti od Shazze da izračuna najveći zajednički djelioc (NZD) svih cijelih brojeva unutar praovougonog bloka polja čija su naspramna tjemena polja (P, Q) i (U, V) uključivo.

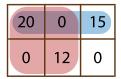
Bazza će odigrati najviše $N_U + N_Q$ poteza $(N_U izmjena vrijednosti polja i <math>N_Q$ postavljanja upita) prije nego mu to sve dosadi i ode igrati kriketa napolju.

Vaš zadatak je da napravite program koji će umjesto Shazze igrati ovu igru sa Bazzom.

Primjeri

Pretpostavimo da je R = 2 i C = 3, te da Bazza igru započinje sa sljedećim potezima:

- Promijeni vrijednost polja (0, 0) na 20;
- Promijeni vrijednost polja (0, 2) na 15;
- Promijeni vrijednost polja (1, 1) na 12.

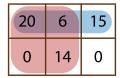


Izgled ploče nakon ovih poteza, prikazan je na slici iznad. Bazza bi mogao pitati za NZD brojeva unutar pravougaonika:

- sa naspramnim tjemenima (0, 0) i (0, 2): tri cijela broja unutar ovog pravougaonika su 20, 0 i 15, pa njihov NZD iznosi 5;
- sa naspramnim tjemenima (0, 0) i (1, 1): četiri cijela broja unutar ovog pravougaonika su 20, 0, 0 i 12, pa njihov NZD iznosi 4;

Pretpostavimo da Bazza sada odigra sljedeće poteze:

- Promijeni vrijednost polja (0, 1) na 6;
- Promijeni vrijednost polja (1, 1) na 14.



Novi izgled ploče, prikazan je na slici iznad. Bazza bi sada mogao pitati za NZD brojeva unutar istih pravougaonika kao i ranije:

- sa naspramnim tjemenima (0, 0) i (0, 2): sada su tri cijela broja unutar ovog pravougaonika 20, 6 i 15, pa njihov NZD iznosi 1;
- sa naspramnim tjemenima (0,0) i (1,1): sada su četiri cijela broja unutar ovog pravougaonika su 20, 6, 0 i 14, pa njihov NZD iznosi 2;

Ovdje je Bazza odigrao ukupno $N_U = 5$ izmjena vrijednosti polja i $N_Q = 4$ upita.

Implementacija

Treba da submitujete file sa implementiranim procedurama <code>init()</code> i <code>update()</code>, kao i funkcijom <code>calculate()</code>, kako je opisano u nastavku.

Da bismo Vam olakšali, primjeri rješenja na Vašem računaru (game.c), game.cpp i game.pas) uključuju funkciju gcd2(X, Y) koja određuje najveći zajednički djelioc (NZD/GCD) za data dva nenegativna cijela broja X and Y. Ako je X = Y = 0 onda će i gcd2(X, Y) vratiti 0.

Ova funkcija je dovoljno brza da se osvoje svi poeni; konkretno, vrijeme izvršavanja je u najgorem slučaju proporcionalno sa log(X + Y).

Vaša procedura: init()

```
C/C++ void init(int R, int C);
Pascal procedure init(R, C : LongInt);
```

Opis

Vaše rješenje treba da implementira ovu proceduru.

Parametri ove procedure daju Vam početnu veličinu ploče, a procedura Vam omogućava da inicijalizirate svoje globalne varijable i strukture podataka. Procedura će biti pozvana tačno jednom, prije bilo kojeg poziva update() ili calculate().

Parametri

- R: Ukupan broj redova.
- C: Ukupan broj kolona.

Vaša procedura: update()

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

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

Opis

Vaše rješenje treba da implementira ovu proceduru.

Ova procedura će biti pozvana kada Bazza postavi novu vrijednost u nekom polju tabele.

Parametri

- P: Red na ploči u kojem se nalazi polje ($0 \le P \le R 1$).
- Q: Kolona na ploči u kojoj se nalazi polje (0 ≤ Q ≤ C 1).
- K: Cio broj koji predstavlja novu vrijednost polja (0 ≤ K ≤ 10¹⁸). Ova vrijednost može
 biti ista kao i ona koja je već ranije bila upisana na polje.

Vaša funkcija: calculate()

```
C/C++ long long calculate(int P, int Q, int U, int V);
Pascal function calculate(P, Q, U, V : LongInt) : Int64;
```

Opis

Vaše rješenje treba da implementira ovu proceduru.

Ova funkcija treba da izračuna najveći zajednički djelioc svih cijelih brojeva unutar pravougaonika sa naspramnim tjemenima (P, Q) i (U, V). Ovaj raspon polja je uključiv, tj., i polja (P, Q) i (U, V) su uključena u pravougaonik.

Ukoliko svi cijeli brojevi u pravougaoniku imaju vrijednost nula, onda funkcija vraća vrijednost nula.

Parametri

- P: Red koji odgovara gornjem-lijevom polju u pravougaoniku ($0 \le P \le R 1$).
- \mathbb{Q} : Kolona koja odgovara gornjem-lijevom polju u pravougaoniku ($0 \le \mathbb{Q} \le \mathbb{C} 1$).
- U: Red koji odgovara donjem-desnom polju u pravougaoniku ($P \le U \le R 1$).
- V : Kolona koja odgovara donjem-desnom polju u pravougaoniku (Q ≤ V ≤ C 1).
- Vraća: NZD (GCD) cijelih brojeva u pravougaoniku, cijelih brojeva u pravougaoniku, ili
 ako svi ovi cijeli brojevi imaju vrijednost nula.

Primjer izvršavanja

Slijedi primjer izvršavanja za ranije opisani primjer :

Poziv fun	Vraća	
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

Ograničenja

- Vremensko ograničenje: pogledati podzadatke
- Memorijsko ograničenje: pogledati podzadatke
- $1 \le R, C \le 10^9$
- $0 \le N \le 10,000$
- $0 \le K \le 10^{18}$, gdje je K neki cijeli broj koji će Bazza staviti u tabelu.

Podzadaci

Pogledajte tekst zadatka na engleskom jeziku za detalje o podzadacima.

Podzadatak	Bodovi	R	С	Nu	N _Q	Vremensko ograničenje	Memorijsko ograničenje

Lokalno testiranje

Grader na Vašem računaru čita ulazne podatke iz file-a game.in. Ovaj file treba biti u sljedećem formatu:

- linija 1: R C N
- sljedećih N linija: po jedan potez na svakoj liniji, onim redoslijedom kojim se ovi potezi izvršavaju

Linija za svaku operaciju treba da bude u jednom od sljedećih formata:

- da označimo (update(P, Q, K)): 1 P Q K
- da označimo calculate(P, Q, U, V): 2 P Q U V

Ilustracije radi, primjer iznad bi trebao biti zapisan u sljedećem formatu:

```
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
```

Napomene vezane za programske jezike

```
C/C++ Morate uraditi #include "robots.h".

Pascal Morate definisati unit Robots. Svi nizovi su indeksirani počevši od 0 (a ne od 1).
```

Kako cijeli brojevi u tabeli mogu biti jako veliki, programeri C/C++ bi trebali koristiti long long tip, a Pascal korisnici bi trebali koristiti long long tip.