

## Chess Rush

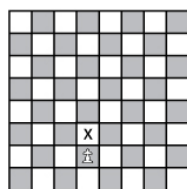
*Šahistan* je mistična pravokutna zemlja koja se sastoji od  $R \cdot S$  kvadratnih polja posloženih u  $R$  redaka i  $C$  stupaca pri čemu je  $R$  veći ili jednak  $C$ . Retci Šahistana indeksirani su od 1 do  $R$ , dok su stupci indeksirani od 1 do  $C$ .

Zanimljivo je da se stanovnici Šahistana ne identificiraju kao Šahistanci, već kao figure, a na spektru figura najčešće se identificiraju kao jedna od sljedećih pet: pješak, top, lovac, kraljica i kralj. Primijetite da su sve figure hodači, odnosno niti jedna nije skakač.

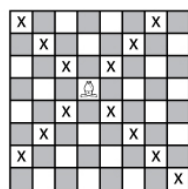
Svaka figura ima svoj način kretanja kroz prostor ovisno o njevoj identifikaciji na spektru. Tako, u jednom koraku:

- pješak se može pomaknuti za jedan redak unaprijed, odnosno s retka  $r$  u u redak  $r + 1$ , bez da promijeni stupac.
- top se može pomaknuti proizvoljan broj stupaca ulijevo ili udesno bez da promijeni redak ili se može pomaknuti proizvoljan broj redaka unaprijed ili unazad bez da promijeni stupac.
- lovac se može pomaknuti na bilo koje polje koje se nalazi na nekoj od dijagonala na kojima se nalazi i njegovo trenutno polje.
- kraljica se može pomaknuti na bilo koje polje na koje bi se mogao pomaknuti top ili lovac kada bi se nalazio na njenom mjestu.
- kralj se može pomaknuti na bilo koje od sebi susjednih polja u 8 smjerova.

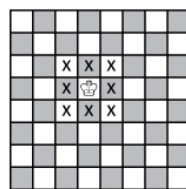
Na sljedećim smo skicama sa X označili sva polja na koja se određen tip figure može pomaknuti u jednom potezu. Retci su indeksirani odozdo prema gore, s stupci su indeksirani slijeva nadesno.



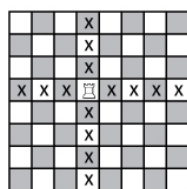
pješak



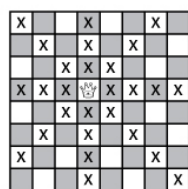
lovac



kralj



top



kraljica

Šahistanu u posljednje vrijeme prijete razne opasnosti. Shodno tome, sve se figure žele što prije pomaknuti sa svojeg polazišta na svoje odredište. Također ih zanima na koliko različitih načina to mogu napraviti (i dalje koristeći najmanji broj poteza). Dva su načina kretanja različita ako se razlikuju u barem jednom polju.

Za potrebe ovog zadatka, pretpostavljamo da se polazišna točka svake figure nalazi u prvom retku ( $r = 1$ ) te da se odredište te figure nalazi u posljednjem retku ( $r = R$ ). Vaš je zadatak odgovoriti na  $Q$  upita: koliki je najmanji broj poteza pomoću kojih figura danog tipa može doći od danog polazišta do danog odredišta te koji je broj načina da to učini?

## Ulazni podaci

U prvom se retku nalaze tri broja  $R$ ,  $C$  i  $Q$  iz teksta zadatka.

Svaki od sljedećih  $Q$  redaka sadrži:

- znak  $T$  koji odgovara tipu figure ('P' za pješaka, 'R' za topa, 'B' za lovca, 'Q' za kraljicu i 'K' za kralja);
- dva broja  $c_1$  i  $c_R$ ,  $1 \leq c_1, c_R \leq C$  koja označavaju da figura putovanje započinje sa stupca  $c_1$  u retku 1, a završava u stupcu  $c_R$  u retku  $R$ .

## Izlazni podaci

Potrebno je ispisati  $Q$  redaka pri čemu  $i$ -ti sadrži dva broja koja predstavljaju odgovore na  $i$ -to pitanje. Prvi broj u retku predstavlja najmanji broj poteza potreban da odgovarajuća figura prođe dano putovanje, a drugi broj predstavlja broj načina na koje je to moguće ostvariti. Budući da broj načina može biti velik, potrebno je samo ispisati njegov ostatak pri dijeljenju s  $10^9 + 7$ .

Ako neka figura ne može doći do odredišta, potrebno je u tom retku ispisati "0 0" (bez navodnika).

## Biblioteka

Dostupna vam je biblioteka koja sadrži osnovne aritmetičke operacije modulo  $10^9 + 7$ . U svim slučajevima, ulazni parametri mogu biti bilo koje `int` vrijednosti, a izlazna vrijednost može biti  $0, 1, 2, \dots, 10^9 + 6$ . Osnovna implementacija dana vam je za potrebe testiranja, a implementirane funkcije su:

- `int Add(int a, int b)`: zbraja brojeve  $a$  i  $b$  modulo  $10^9 + 7$ .
- `int Sub(int a, int b)`: oduzima broj  $b$  od broja  $a$  modulo  $10^9 + 7$ .
- `int Mul(int a, int b)`: množi brojeve  $a$  i  $b$  modulo  $10^9 + 7$ .
- `int Div(int a, int b)`: dijeli broj  $a$  sa  $b \neq 0$  modulo  $10^9 + 7$ . Odnosno, vraća vrijednost  $q$  ako i samo ako je `Mul(b, q)` jednako  $a$  modulo  $10^9 + 7$ .

Možete pretpostaviti da se sve operacije izvršavaju u konstantnom vremenu.

Da biste koristili ove funkcije, potrebno je uključiti odgovarajuću biblioteku. Odnosno, potrebno je u rješenje dodati liniju `#include "arithmetics.h"`.

## Testiranje

Sa sustava za evaluaciju možete preuzeti arhivu *sample.zip* koja sadži zaglavlje `arithmetics.h` sa deklaracijama spomenutih funkcija, kao i implementacijsku datoteku `arithmetics.cpp` koja sadrži oglednu implementaciju tih funkcija. Obje su datoteke potrebne kako biste testirali vaše rješenje.

Da biste to uspješno napravili, potrebno je kopirati obje datoteke u direktorij u kojem se nalazi izvorni kod vašeg rješenja (npr. `chessrush.cpp`) koje koristi biblioteku `arithmetics.h`.

Da biste preveli (kompajlirali) vaše rješenje, dovoljno je izvršiti naredbu `g++ -o chessrush arithmetics.cpp chessrush.cpp` u terminalu. Ako pak koristite IDE, najvjerojatnije je ručno potrebno dodati sve tri datoteke u vaš projekt prije no što ga izgradite (buildate).

Točne izlaze probnih primjera možete pronaći u datotekama `output0.txt`, `output1.txt`. Niti jedan od alata iz arhive ne provjerava ispravnost vašeg rješenja.

Prilikom slanja rješenja na sustav za evaluaciju potrebno je samo učitati (uploadati) datoteku `chessrush.cpp`.

## Probni primjeri

*Ulazni podaci*

8 8 5  
P 1 2  
R 4 8  
Q 2 3  
B 3 6  
K 5 5

*Izlazni podaci*

0 0  
2 2  
2 5  
2 2  
7 393

## Ograničenja

$1 \leq Q \leq 1000$   
 $2 \leq C \leq 1000$   
 $C \leq R \leq 10^9$

**Vremensko ograničenje:** 1.3 s

**Memorijsko ograničenje:** 64 MiB

## Bodovanje

Podzadatak	Bodovi	Ograničenja
1	0	probni primjeri
2	8	$T \in \{'P', 'R', 'Q'\}$ , tj. sve su figure pješaci, topovi ili kraljice
3	15	$T = 'B'$ i $C, R \leq 100$
4	22	$T = 'B'$
5	5	$T = 'K'$ i $C, R \leq 100$ and $Q \leq 50$
6	8	$T = 'K'$ i $C, R \leq 100$
7	15	$T = 'K'$ i $C \leq 100$
8	20	$T = 'K'$
9	7	nema dodatnih ograničenja