

Šahovski kros

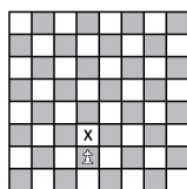
Mistični svet Šahovske Dežele je pravokotna mreža sestavljena iz samih kvadratov in ima R vrstic ter C stolpcev, kjer je R vedno večji ali enak C . Vrstice in stolpci so označeni s števili med 1 in R ter 1 in C .

Prebivalce Šahovske Dežele pogosto imenujemo kar *figure*, obstaja pa 5 različnih figur, ki lazijo po deželi: kmetje, trdnjave, lovci (tekači), kraljice in kralji. Ker so konje že davno zamenjali avtomobili in traktorji, pa žal, tudi v Šahovski Deželi, ni več konj (skakačev).

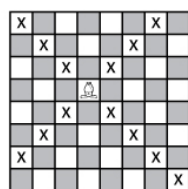
Vsaka izmed figur se med kvadrati premika na povsem samosvoj način: v enem koraku,

- se kmet lahko premakne eno vrstico naprej (torej iz vrstice r v vrstico $r + 1$), brez menjave stolpca;
- se trdnjava lahko premakne za poljubno število stolpcev levo/desno brez menjave vrstic ALI poljubno število vrstic naprej/nazaj brez menjave stolpcev;
- lovec (tekač) se lahko premakne na poljuben kvadrat na obeh diagonalah, katerih presečišče je njegov trenutni kvadrat;
- kraljica se lahko premakne na poljubno polje, na katerega bi si iz njene pozicije lahko premaknila trdnjava ali lovec (tekač);
- in kralj se lahko premakne na poljubnega izmed 8 sosednjih polj.

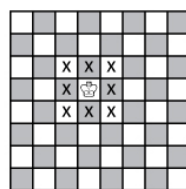
Na spodnji sliki smo z X označili kvadrate, na katere se lahko v eni potezi premakne figura (na slikah so vrstice oštevilčene od spodaj navzgor, stolpci pa od leve proti desni).



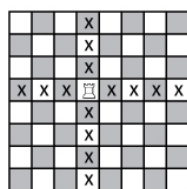
kmet



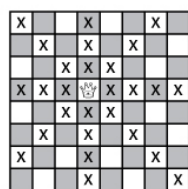
lovec (tekač)



kralj



trdnjava



kraljica

V zadnjih časih je v Šahovski Deželi postalo nevarno: med potovanje po deželi lahko neznane sile nepričakovano zajamejo figure, ki potem izginejo. Zaradi tega bi figure rade na svoj cilj prišle čim hitreje (s čim manjšim številom korakov), prav tako pa jih zanima število različnih

poti, s katerimi lahko cilj dosežejo z najmanjšim številom korakov – večje število možnih poti pomeni manjše možnosti, da jih zajamejo. Dve poti sta upoštevani kot različni, če se razlikujeta v vsaj enem obiskanem kvadratu.

Za to nalogo lahko predpostavimo, da figure v Šahovsko Deželo vstopijo v vrstici 1 in izstopijo v vrstici R . Tvoja naloga je odgovoriti na Q vprašanj: za podano vrsto figure, stolpec, v katerem vstopi v vrstico 1, stolpec, ki ga more doseči v vrstici R , da lahko izstopi, moraš izračunati najmanjše število premikov v Šahovski Deželi in število različnih možnosti, s katerimi to lahko stori.

Vhod

Prva vrstica vsebuje tri s presledkom ločena cela števila R , C in Q , število vrstic in stolpcev v Šahovski Deželi in število vprašanj. Temu sledi Q vrstic.

Vsaka vrstica je sestavljena iz

- znaka T , ki pove, za kakšne vrste figuro gre ('P' za kmeta, 'R' za trdnjavo, 'B' za lovca (tekača), 'Q' za kraljico in 'K' za kralja.
- dve števili c_1 in c_R , $1 \leq c_1, c_R \leq C$, ki označujeta, da figura začne v c_1 -tem stolpcu vrstice 1 in mora doseči c_R -ti stolpec v vrstici R .

Izhod

Izpisati moraš Q vrstic, i -ta vrstica vsebuje dve s presledkom ločeni celi števili, odgovor na i -to vprašanje: prvo število je minimalno število potez, ki so potrebne, drugo pa število možnih poti, ki vsebujejo toliko potez. Ker so lahko odgovori zelo veliki, jih moraš računati glede na ostanek pri deljenju s številom $10^9 + 7$, pri čemer lahko uporabljaš funkcije iz knjižnice, ki je na voljo na sodniškem sistemu.

Če ni mogoče doseči ciljnega kvadrata, izpiši vrstico "0 0".

Knjižnica

Sodniški sistem za osnovno aritmetiko glede na ostanek pri deljenju s številom $10^9 + 7$ omogoča naslednje funkcije. V vseh primerih so veljavni argumenti za funkcije knjižnice poljubne vrednosti tipa `int`, vrnjen rezultat pa je v razponu $0, 1, 2, \dots, 10^9 + 6$. Vzorčna implementacija je na voljo za testiranje tvoje rešitve, za več informacij glej naslednji razdelek.

- `int Add(int a, int b)`: sešteje števili a in b , in vrne ostanek pri deljenju rezultata z $10^9 + 7$.
- `int Sub(int a, int b)`: odšteje število b od števila a , potem vrne ostanek pri deljenju rezultata z $10^9 + 7$.
- `int Mul(int a, int b)`: izračuna produkt števil a in b , potem vrne ostanek pri deljenju rezultata z $10^9 + 7$.

- `int Div(int a, int b)`: izračuna kvocient a deljeno $b \neq 0$ glede na ostanek pri deljenju z $10^9 + 7$, torej vrne vrednost $0 \leq q < 10^9 + 7$ če in samo če je rezultat $\text{Mul}(b, q) = a \bmod (10^9 + 7)$.

Lahko predpostaviš, da se vse operacije izvedejo v konstantnem času.

Za uporabo teh funkcij moraš dodati vrstico `#include "arithmetics.h"` na seznam vključitev svojega programa.

Testiranje

Datoteka v dodatku *sample.zip* vsebuje zaglavno datoteko ("header file") `arithmetics.h` z deklaracijami funkcij iz knjižnice in tudi datoteko `arithmetics.cpp`, ki vsebuje primer implementacije in jo uporablja za preverjanje svojih rešitev.

Da ju lahko uporabiš, moraš obe datoteki skopirati v isti imenik, v katerem se nahaja tudi izvorna koda tvoje rešitve (recimo `chessrush.cpp`) in dodati vrstico `#include "arithmetics.h"` na seznam vključitev svojega programa.

Potem prevedi `chessrush.cpp` skupaj z `arithmetics.cpp`, recimo z uporabo `g++ -o chessrush arithmetics.cpp chessrush.cpp` v ukazni vrstici. Če uporabljaš integrirano razvojno okolje, ki bazira na projektu, boš moral, preden prevedeš svojo rešitev, vse tri datoteke dodati v svoj projekt.

Pravilne odgovore za vzorčni vhod lahko najdeš v datotekah `output0.txt`, `output1.txt`. Nobeno izmed orodij ali funkcij, ki so na voljo, ne preverja pravilnosti tvojih odgovorov.

Ko oddajaš rešitev, moraš na sodniški sistem oddati zgolj datoteko `chessrush.cpp`.

Primeri

<i>Vhod</i>	<i>Izhod</i>
8 8 5	0 0
P 1 2	2 2
R 4 8	2 5
Q 2 3	2 2
B 3 6	7 393
K 5 5	

Omejitve

$$1 \leq Q \leq 1000$$

$$2 \leq C \leq 1000$$

$$C \leq R \leq 10^9$$

Časovna omejitev: 1.3 s

Prostorska omejitev: 64 MiB

Ocenjevanje

Podnalog	Točke	Omejitve
1	0	primer
2	8	$T \in \{ 'P', 'R', 'Q' \}$, torej vse figure so kmetje, trdnjave ali kraljice
3	15	$T = 'B'$ in $C, R \leq 100$
4	22	$T = 'B'$
5	5	$T = 'K'$ in $C, R \leq 100$ in $Q \leq 50$
6	8	$T = 'K'$ in $C, R \leq 100$
7	15	$T = 'K'$ in $C \leq 100$
8	20	$T = 'K'$
9	7	brez dodatnih omejitev