

Szachowa gorączka

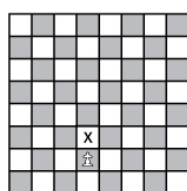
Mityczny świat Szachlandii jest prostokątną planszą złożoną z kwadratów ułożonych w R wierszy i C kolumn ($C \leq R$). Wiersze i kolumn numerujemy odpowiednio od 1 do R i od 1 do C .

Mieszkańcy Szachlandii zwyczajowo nazywani są *pionami*, istnieje 5 różnych typów pionów: pionki, wieże, gońce, hetmany i króle. W przeciwieństwie do popularnego przekonania, kawaleria już dawno wymarła w Szachlandii, dlatego nie istnieją tam konie.

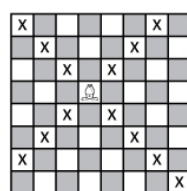
Każdy z pionów ma unikalny sposób poruszania się po planszy: w jednym kroku,

- pionek może wykonać ruch o wiersz do przodu (z wiersza r do $r + 1$), bez zmiany kolumny;
- wieża może poruszyć się o dowolną liczbę kolumn w lewo/prawo bez zmiany wiersza lub poruszyć się o dowolną liczbę wierszy w przód/tył bez zmiany kolumny;
- gońiec może poruszyć się do dowolnego pola znajdującego się na przekątnej przecinającej aktualnie zajmowane pole;
- hetman może poruszyć się do dowolnego pola, do którego może poruszyć się gońiec lub wieża z jego pozycji;
- król może poruszyć się do dowolnego z 8 sąsiadujących pól.

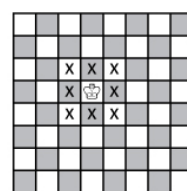
Na poniższych rysunkach jako X oznaczone pola, do których dany pion może poruszyć się w jednym kroku (wiersze ponumerowano od dołu do góry, a kolumny od lewej do prawej).



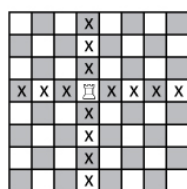
pionek



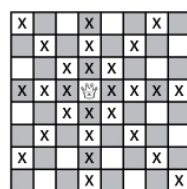
gońiec



król



wieża



hetman

Obecnie Szachlandia stała się niebezpiecznym miejscem, pionki które przechodzą przez nią mogą zostać niespodziewanie porwane przez nieczyste siły. Z tego powodu pionki chcą dotrzeć do celu tak szybko (w najmniejszej możliwej liczbie ruchów) jak to możliwe, są również zainteresowane liczbą różnych sposobów na jakie są w stanie to zrobić używając minimalnej liczby

kroków. Dwie ścieżki uznajemy za różne jeśli różnią się co najmniej jednym odwiedzionym polem.

Zakładamy, że piony zaczynają swoją podróż w danej kolumnie wiersza 1 i kończą w danej kolumnie wiersza R . Twoim zadaniem jest odpowiedzenia na Q zapytań: mając dany typ pionu, początkowy i końcowy numer kolumny, należy obliczyć minimalną liczbę ruchów potrzebnych do dotarcia do celu oraz liczbę sposobów w jaki można to zrobić.

Wejście

Pierwsza linia wejścia zawiera trzy liczby całkowite R , C i Q , liczbę wierszy i kolumn Szachlandii oraz liczbę zapytań.

Każda z następnych Q linii zawiera:

- znak T oznaczający typ pionu w zapytaniu ('P' dla pionka, 'R' dla wieży, 'B' dla gońca, 'Q' dla hetmana i 'K' dla króla);
- dwie liczby całkowite c_1 i c_R , $1 \leq c_1, c_R \leq C$, oznaczające, że pion startuje z kolumny c_1 wiersza 1 i ma dotrzeć do kolumny c_R wiersza R .

Wyjście

Na wyjściu powinno znaleźć się Q linii, i -ta z nich zawiera dwie liczby całkowite będące odpowiedziami na i -te zapytanie: pierwsza z nich to minimalna liczba kroków potrzebna do osiągnięcia celu, druga z nich to liczba różnych ścieżek do celu używająca minimalnej liczby kroków. Jako że odpowiedź może być bardzo duża należy podać ją modulo $10^9 + 7$, w tym celu można użyć funkcji bibliotecznych dostarczonych przez sprawdzarkę.

Jeśli nie da się dotrzeć do celu należy wypisać linię "0 0".

Biblioteka

Sprawdzarka dostarcza następujących funkcji bibliotecznych służących do wykonywania podstawowych operacji w arytmetyce modulo $10^9 + 7$. Każda z funkcji może być wywołana z parametrami będącymi dowolnymi wartościami typu `int`, a jej wynik znajduje się w zakresie $0, 1, 2, \dots, 10^9 + 6$. Przykładowa implementacja jest dostępna do testowania rozwiązania, instrukcja testowania znajduje się w następnej sekcji.

- `int Add(int a, int b)`: dodaje liczby a i b , zwraca wynik modulo $10^9 + 7$.
- `int Sub(int a, int b)`: odejmuje b od a , zwraca wynik modulo $10^9 + 7$.
- `int Mul(int a, int b)`: oblicza iloczyn a i b , zwraca wynik modulo $10^9 + 7$.
- `int Div(int a, int b)`: oblicza iloraz a podzielonego przez $b \neq 0$ modulo $10^9 + 7$, to znaczy zwraca takie $0 \leq q < 10^9 + 7$, że `Mul(b,q)` wynosi $(a \bmod 10^9 + 7)$.

Możesz założyć, że wszystkie powyższe operacje wykonywane są w czasie stałym.

Aby uzyskać dostęp do tych funkcji musisz dodać `#include "arithmetics.h"` do listy nagłówków swojego rozwiązania.

Testowanie

Dostarczony plik *sample.zip* zawiera nagłówek `arithmetics.h` z deklaracjami funkcji bibliotecznych, a także plik `arithmetics.cpp`, z ich przykładową implementacją, której możesz użyć do testowania swojego rozwiązania.

Aby móc ich użyć powinieneś skopiować oba pliki do folderu, w którym znajduje się twój kod źródłowy (`chessrush.cpp`) i dodać linię `#include "arithmetics.h"` do listy jego nagłówków.

Następnie powinieneś skompilować `chessrush.cpp` razem z `arithmetics.cpp`, np. używając `g++ -o chessrush arithmetics.cpp chessrush.cpp` w linii komend, lub używając IDE ręcznie dodać wszystkie trzy pliki do plików projektu przed kompilacją rozwiązania.

Poprawne odpowiedzi do testów przykładowych znajdują się w plikach `output0.txt`, `output1.txt`. Żadne z dostarczonych narzędzi czy funkcji nie sprawdza poprawności twoich odpowiedzi.

Do sprawdzarki należy wysłać tylko plik `chessrush.cpp`.

Przykłady

<i>Wejście</i>	<i>Wyjście</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	

Ograniczenia

$$1 \leq Q \leq 1000$$

$$2 \leq C \leq 1000$$

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

Limit czasu: 1.3 s

Limit pamięci: 64 MiB

Ocenianie

Podzadanie	Punkty	Ograniczenia
1	0	przykład
2	8	$T \in \{'P', 'R', 'Q'\}$, każdy z pionów jest pionkiem, wieżą lub hetmanem
3	15	$T = 'B'$ i $C, R \leq 100$
4	22	$T = 'B'$
5	5	$T = 'K'$ i $C, R \leq 100$ i $Q \leq 50$
6	8	$T = 'K'$ i $C, R \leq 100$
7	15	$T = 'K'$ i $C \leq 100$
8	20	$T = 'K'$
9	7	brak dodatkowych ograniczeń