Sphere online judge
(/)

≡

CSTATE2 - Chessboard State 2: International Chess Tournament

no tags **manage tags**

Note: This is a harder version of CSTATE1 - a solution to this problem will (except I/O format) pass there. It is also an easier version of CSTATE3 - the solution to it will pass here.

A year ago the inhabitants of Slovakistan decided to organize a national chess tournament.

Since the first place prize was 1 point on SPOJ, everyone played as if their life was at stake.

However, since time was not measured in this chess tournament (unlike at most chess tournaments in the rest of the world), a problem occured - players took way too long to admit defeat.

They always just kept thinking and thinking, trying to find some move which would prevent their king from being taken. Just one more moment... And perhaps one more...

Luckily, a skilled programmer lent them a hand and got rid of this nuisance!

This year, they are organizing a chess tournament once more. However, since word got out of how awesome it was, contestants from all over the world are planning to join. Slovakistan can not embarras itself - this year's tournament has to be more awesome, way more dramatic, and - most importantly - a whole lot bigger. Hence, during this year's tournament the contestants will not only play on the classic 8-by-8 chessboards, but rather on n-by-n boards.

That gives rise to a new problem though - the programs of skilled programmers which helped them last year are not capable of deciding the state of chessboards of such size!

Once again, the inhabitants of Slovakistan need your help.

For a given description of a chessboard, decide its state - which player's king is under threat, and whether it is a check or a checkmate. Additionally, if it's a check, to help build up suspense, the contestants would like to know the number of valid moves the checked player has after which his king is no longer threatened.

For the purposes of this problem only take into account basic moves of every chesspiece; complicated moves such as Castling, En passant, moving a pawn 2 squares or Promotion are not accepted in Slovakistan.

Input

The first line of the input contains the integer **1 ≤ t ≤ 12000** - the number of chessboards. **t** descriptions of a chessboard follow.

The first line of each description contains two integers **$8 \le n \le 10^5$, $2 \le p \le 10^5$** - the length of side of the chessboard and the number of pieces currently on it.

p lines follow, each in the form '**x y c**' , where **$1 \le x,y \le n$** indicate the coordinates of the chesspiece; the upper-left square has coordinates **(1,1)** while the bottom-right square is at **(n,n)**.

c represents the type of the chesspiece - this character is from the set **{KQRBHPkqrbhp}**, representing in this order the king, queen, rook, bishop, knight (horse), and pawn.

The pieces belonging to the White player are marked by lowercase characters; the upper side of the chessboard (lower **y** coordinates) belongs to him, hence white pawns move in the positive **y** direction.

The pieces belonging to the Black player are marked by uppercase characters; the lower side of the chessboard (greater **y** coordinates) belongs to him, hence black pawns move in the negative **y** direction.

A blank line follows after every chessboard description.

The number or placement of the pieces may by all means be impossible to reach in a game of chess, however you may assume that on every chessboard there is exactly one white king ('**k**') and one black king ('**K**'). No two pieces are on the same coordinates.

Input files are 'reasonable' - that is, if a file contains a large amount of testcases, they are reasonably small. Specifically, the sum of **n+p** within an input file does not exceed **450,000**.

Output

For each chessboard output one line with one of the following messages:

- "**Safe**", if neither players' kings are being threatened
- "**Impossible**", if both players' kings are being threatened
- "**{colour} Check - m Plausible Moves**", where **{colour}** is either "**Black**" or "**White**", if the respective player's king is being threatened, and there exists **m** valid moves by his pieces after which his king is no longer threatened
- "**{colour} Checkmate**", where {colour} is either "**Black**" or "**White**", if the respective player's king is being threatened, and there exists no valid move by any of his pieces after which his king is no longer threatened

Additional note

There are 16 input files "0" through "15". File 0 is the example given below. File 1 contains roughly a half of the testdata from CSTATE1; file 2 contains the rest of testdata from CSTATE1, along with some chessboards of size 100. Time limit is ##5 seconds for each file - four times my worst time on any input file. Despite that, if you believe your solution runs just slightly longer, ping me and I might increase it.

Example

```
Input:
7
8 4
5 1 k
4 3 H
4 5 h
5 7 K

8 4
5 2 h
3 3 k
6 4 H
4 5 K

8 6
1 1 k
4 1 H
4 2 H
2 3 H
3 3 H
7 6 K

8 7
1 1 k
4 1 H
4 2 H
2 3 H
3 3 H
7 6 K
2 7 r

8 9
1 1 K
7 1 R
8 1 r
1 2 R
3 2 h
8 2 r
1 5 r
2 5 r
6 6 k

8 6
1 1 k
8 2 R
4 4 B
2 5 R
7 6 K
3 7 q

8 9
3 2 P
4 2 P
5 2 P
3 3 P
4 3 k
5 3 P
4 4 P
5 4 p
4 5 K
```

```
Output:
Impossible
Safe
White Checkmate
White Check - 1 Plausible Moves
Black Checkmate
White Check - 1 Plausible Moves
Black Check - 5 Plausible Moves
```

If you have any questions, problems or suggestions, do let me know :)

Added by:	Hodobox (/users/hodobox)
Date:	2017-07-04
Time limit:	~~2.5s~~ 5s
Source limit:	50000B
Memory limit:	1536MB
Languages:	All
Resource:	own problem