Chessboard State 3: Intergalactic Chess Tournament

Note: This is a harder version of CSTATE1 and CSTATE2 - a solution to this problem will (except I/O format) pass there.

A thousand years 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!

The next year, they organized a chess tournament once more. However, since word got out of how awesome it was, contestants from all over the world joined. Slovakistan could not embarras itself - that year's tournament had to be more awesome, way more dramatic, and - most importantly - a whole lot bigger. Hence, during that year's tournament the contestants did not only play on the classic 8-by-8 chessboards, but rather on n-by-n boards.

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

Once again, the inhabitants of Slovakistan needed help.

That is the origin story of the now most popular tournament in the entire universe. These days all the galactic empires have a ceasefire in the intergalactic wars to all take part in the Intergalactic Chess Tournament. For its thousandth aniversary, the Slovakistan Milky Way Empire is preparing a truly spectacular show - the chesspieces this year will be represented by planets, and the chessboard will be an entire galaxy!

As in the legendary tale from a thousand years ago, they are facing the same problem: the programs written to decide the state of a chessboard can not handle the size of the chessboard, even when run on the most modern quantum computers. To tackle this problem, they have constructed a time machine to come to this legendary time - and look for a skilled programer who could help them with this task!

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 the known universe.

Input

The first line of the input contains the integer $1 \leq t \leq 20000$ - the number of chessboards. **t** descriptions of a chessboard follow.

The first line of each description contains two integers $8 \leq n \leq 10^{18}$, $2 \leq p \leq 2*10^5$ - 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 $1 \leq x,y \leq n$ 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 standard game of intergalactic 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 **p** within an input file does not exceed 10^6.

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. Files 1 through 7 contain testdata from CSTATE2.

After submission you can view extra information about the result of each file by clicking the result text ("accepted","wrong answer",...), such as the result of each file and the time/memory used, but no message is present like in CSTATE1 - if you are stuck, consider submitting there for a hint at what is off in your solution.

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 :)