# Chess Validator

Chess is the oldest game, but it is still popular these days. Here you need to validate if player move his pieces right.

Board is 8x8 matrix. There are 6 kind of pieces:

* **K**ing - moves exactly one square horizontally, vertically, or diagonally.
* **R**ook- moves any number of vacant squares in a horizontal or vertical direction.
* **B**ishop - moves any number of vacant squares in any diagonal direction.
* **Q**ueen - moves any number of vacant squares in a horizontal, vertical, or diagonal direction.
* K**N**ight - moves to the nearest square not on the same [rank](https://en.wikipedia.org/wiki/Glossary_of_chess#rank), [file](https://en.wikipedia.org/wiki/Glossary_of_chess#file), or [diagonal](https://en.wikipedia.org/wiki/Glossary_of_chess#diagonal). (This can be thought of as moving two squares horizontally then one square vertically, or moving one square horizontally then two squares vertically—i.e. in an "L" pattern.)
* **P**awn – moves straight forward **one square.**

Yeah we know, that there is some more moves, but for now these are enough. You can check visual representation for moves [here.](https://en.wikipedia.org/wiki/Rules_of_chess#Basic_moves)

On first 8 rows you will receive board with all pieces in it. Empty cells will be marked with "**x**" and all pieces will be separated with coma.

On next lines you will receive moves that you need to check. There can be 3 kind of problems for which you need looking for. You need to check for these problems in order:

1. There is no such a piece on starting square.
2. Piece makes invalid move (look above).
3. Piece get out of board.

So if you find that there is no such a piece, you don't need to look for others. If you find any problem you have to cancel the move and board have to stay the same for next move.

### Input

On first 8 rows you will receive a game board rows in format:

**{x,x,Q,x,R,x,x,P}**

**{x,B,x,x,N,K,x,P}**

There are 7 symbols at all:

* **K** – King
* **R** – Rook
* **B** – Bishop
* **Q –** Queen
* **N –** Knight
* **P –** Pawn
* **x –** Emptycell

For moves you will receive lines with 6 characters in format:

**{Q01-12}**

First symbol is for piece type. Then you will read two digits for start position (row, col). Line will finish with new position (row, col)

Read moves till you find "**END**" command

### Output

You can have three different output, depending on what kind of problem you hit.

1. **There is no such a piece!**
2. **Invalid move!**
3. **Move go out of board!**

You can have to print only one message per line, so check follow the order of problems

### Constraints

* Pieces will never go to cell with another piece
* Moves count will be in range [0…1000]
* Time limit: 0.3 sec. Memory limit: 16 MB.

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| x,x,x,x,x,N,x,x  x,K,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  K33-44  K11-02  N05-26  N26-47  N11-02  N47-58  END | There is no such a piece!  There is no such a piece!  Move go out of board! |
| x,x,x,x,x,x,x,x  x,Q,x,x,x,x,x,x  x,x,x,x,x,x,x,R  x,x,x,x,x,x,x,x  x,x,x,x,x,x,x,x  x,x,x,B,x,x,x,x  x,x,x,x,x,P,x,x  x,x,x,x,x,x,x,x  Q11-15  Q15-24  R27-36  R27-28  B53-55  Q24-26  R27-07  R27-07  P65-55  P55-65  END | Invalid move!  Move go out of board!  Invalid move!  There is no such a piece!  Invalid move! |