

220 Othello

Othello is a game played by two people on an 8 x 8 board, using disks that are white on one side and black on the other. One player places disks with the white side up and the other player places disks with the black side up. The players alternate placing one disk on an unoccupied space on the board. In placing a disk, the player **must** bracket at least one of the other color disks. Disks are bracketed if they are in a straight line horizontally, vertically, or diagonally, with a disk of the current player's color at each end of the line. When a move is made, **all** the disks that were bracketed are changed to the color of the player making the move. (It is possible that disks will be bracketed across more than one line in a single move.)

	1	2	3	4	5	6	7	8
1								
2								
3				●				○
4				●	●	●		
5			○	●	○	○		
6			●	●	●	○		
7								
8								

	1	2	3	4	5	6	7	8
1								
2								
3				●				○
4				●	●	●		
5			○	●	○	○		
6			○	○	●	○		
7			○					
8								

On the left

Legal Moves for White

(2,3),(3,3),(3,5),(3,6)

(6,2),(7,3),(7,4),(7,5)

On the right

Board Configuration after

White Moves to (7,3)

Write a program to read a series of Othello games. The first line of the input is the number of games to be processed. Each game consists of a board configuration followed by a list of commands. The board configuration consists of 9 lines. The first 8 specify the current state of the board. Each of these 8 lines contains 8 characters, and each of these characters will be one of the following:

‘-’ indicating an unoccupied square

‘B’ indicating a square occupied by a black disk

‘W’ indicating a square occupied by a white disk

The ninth line is either a ‘B’ or a ‘W’ to indicate which is the current player. You may assume that the data is legally formatted.

The commands are to list all possible moves for the current player, make a move, or quit the current game. There is one command per line with no blanks in the input. Commands are formatted as follows:

List all possible moves for the current player. The command is an ‘L’ in the first column of the line. The program should go through the board and print all legal moves for the current player in the format (x, y) where x represents the row of the legal move and y represents its column. These moves should be printed in row major order which means:

- 1) all legal moves in row number i will be printed before any legal move in row number j if j is greater than i
- and 2) if there is more than one legal move in row number i , the moves will be printed in ascending order based on column number.

All legal moves should be put on one line. If there is no legal move because it is impossible for the current player to bracket any pieces, the program should print the message “No legal move.”

Make a move. The command is an ‘M’ in the first column of the line, followed by 2 digits in the second and third column of the line. The digits are the row and the column of the space to place the piece of the current player’s color, *unless the current player has no legal move*. If the current player has no legal move, the current player is first changed to the other player and the move will be the move of the new current player. You may assume that the move is then legal. You should record the changes to the board, including adding the new piece and changing the color of all bracketed pieces. At the end of the move, print the number of pieces of each color on the board in the format “Black - **xx** White - **yy**” where **xx** is the number of black pieces on the board and **yy** is the number of white pieces on the board. After a move, the current player will be changed to the player that did not move.

Quit the current game. The command will be a ‘Q’ in the first column of the line. At this point, print the final board configuration using the same format as was used in the input. This terminates input for the current game.

You may assume that the commands will be syntactically correct. Put one blank line between output from separate games and no blank lines anywhere else in the output.

Sample Input

```

2
-----
-----
-----
---WB---
---BW---
-----
-----
-----
W
L
M35
L
Q
WWWWB---
WWWB----
WWB-----
WB-----
-----
-----
-----
B
L
M25
L
Q

```

Sample Output

(3,5) (4,6) (5,3) (6,4)

Black - 1 White - 4

(3,4) (3,6) (5,6)

----W---

---WW---

---BW---

No legal move.

Black - 3 White - 12

(3,5)

WWWWB---

WWWWW---

WWB-----

WB-----
