# Introduction

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  | | --- | | problem **0** | | **Reversi Moves** | | y points | |  |
|  |  |

Reversi is a game played between two players on an 8x8 grid. The opening position is shown below. White moves first. On White’s turn, White must place a white piece on an empty square, such that there is at least one black piece between the new piece and another white piece, horizontally, vertically, or diagonally. After placement, all such black pieces (linearly between the new piece and the next white piece) are changed to white. The pieces must be adjacent in a single line, without empty squares. Black then plays similarly, converting one or more lines of white pieces. Play continues until either the board is filled or one player cannot make a legal move. An example of a game’s first few moves is shown below.

Opening Position Move 1: f4 Move 2: f5 Move 3: f6 Move 4: g5 Move 5: h6

     

Move 6: f7 Move 7: e6

 

Your program should accept a sequence of coordinates representing the placement of pieces for each turn (white places first.) Before each input, your program will print an ASCII representation of the board at the current state of the game. When “END” is input, the program terminates. You do not need to check the validity of each move.

# Input

f4

f5

f6

g5

h6

f7

e6

END

# Output

Your program should output the state of the board to the screen after all moves are completed. Use ‘B’ for Black, ‘W’ for White, and ‘ . ’ (period) for empty squares. *Hint: printing the state of the board after each move may help you debug your program.*

........

........

........

...WWW..

...BWBW.

....WB.W

.....B..

........

Black move 8: END