2048 Pt. 2

In the previous section you started to create the game 2048. Now you should finish coding the game, implementing a two-dimensional array for the full game board. Also keep track of the score, and support movement in all four directions.

To review the rules of 2048, when the game board shifts in a certain direction all of the numbered tiles move toward that edge of the board, and like-numbered tiles combine, adding their sum to the score. The game is over when there are no more blank tiles, and additional moves cannot be made because the state of the board will not change (i.e., there are no two adjacent tiles with the same number, vertically or horizontally.) The goal of 2048 is to get the highest score possible before the game ends.

Model-View-Controller

Recall that we are organizing the classes that make up this project following the MVC design pattern.

The Model handles all the data in the game: it updates the board, figures out if the game has been won, and keeps track of the score.

The View handles all input from the user and output to the screen. It draws the board and displays messages.

The Controller is the main method that can talk to both objects. The Model cannot talk to the View, nor can the View talk to the Model. The Controller can call methods from both classes, and pass references to objects as appropriate.

Example output

```
Current score: 8960
   32  128  256  512
   16   2   32   512
   2    2   16   128
   4   16   32   -
Move: a
```

Notice that when we move left, the two 2s combine to form a 4 and the score is incremented by that amount.

```
Current score: 8964
   32  128  256  512
   16   2   32   512
   4   16  128   -
   4   16   32   -
Move: w
```

Notice that when we move up, the two 4s and the two 16s combine to form a 8 and a 32, and the two 512s combine to form a 1024. The score is incremented by the sum of all of those values.

```
Current score: 10028
   32  128  256 1024
   16   2   32    -
   8   32  128    -
   4    -   32    -
Move: q
Your final score is 10028
Goodbye!
```

When we quit, the game immediately ends and the final score is displayed.

Finally, be sure to implement a `checkWin()` method that determines whether or not the game is over. The game is over when there are no more empty spaces left on the board, and no two tiles are next to each other with the same value that can be combined. You should check all adjacent tiles vertically and horizontally.