For this exercise, you will be given 3 hours to implement the game of Othello (a.k.a. Reversi). You may use any language you want, in any programming environment. You have full access to the internet, except for things directly related to existing implementations of the game. Email your solution to dev@everlaw.com.

What we're looking for

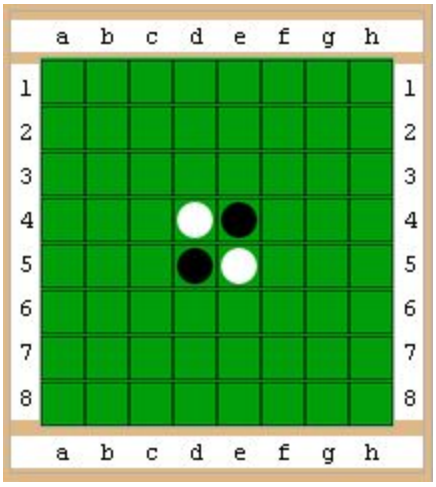
- Clean, readable, well-planned-out code for the game logic
- Some system of input-output (command line is fine, anything else is also fine) that allows the game to be played
- A game that functions as specified in the rules below
- If you're running low on time, a plan for implementing the rest of your solution

What we're NOT looking for

- We don't expect a pretty user interface. The bare minimum of playability is fine, such as a command line interface.
- We don't expect optimal performance. Clean, well-organized code is more important than speed.
- You don't need to use a complex development environment or complex libraries. Only use things that you're comfortable with and that are necessary.
- We don't expect excessive documentation. You should provide as much or as little documentation as you're comfortable with, but clean code should be fairly self-explanatory, and spending too much time documenting instead of coding is a liability.
- You don't have to use a particular programming language. We prefer for you to use any language that you're familiar with.

Overview and setup

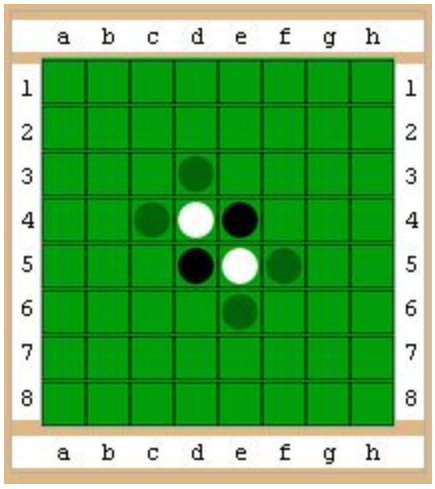
- Othello is played by two players (white and black) on an 8 x 8 board.
- Each player starts with two disks in the center of the board, placed diagonally across from each other like so:



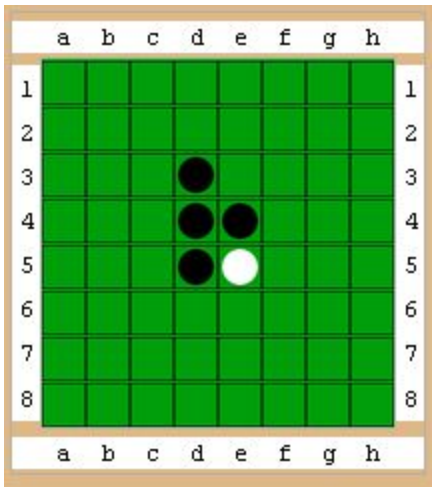
- Play alternates, with black going first.
- Each turn consists of placing a new disk of your color on the board, and then “flipping” all of your opponent’s disks that are “outflanked” (more on this below) to your own color.
- The goal is to finish the game with more disks on the board than your opponent.

Play

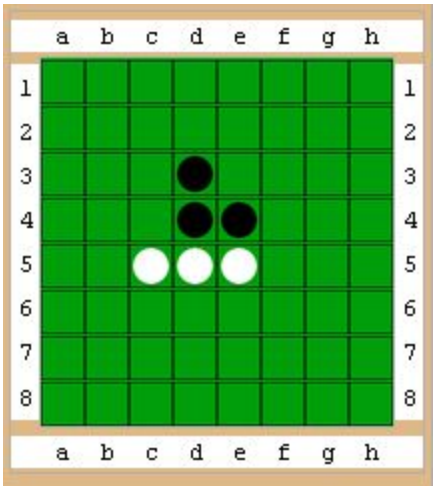
- You must place your disk in a location that “outflanks” one or more of your opponent’s disks. “Outflanking” means surrounding a line of your opponent’s disks (horizontal, vertical, OR diagonal) with two of your own discs.
- For example, here are the four locations that black can place a disk on the first turn:



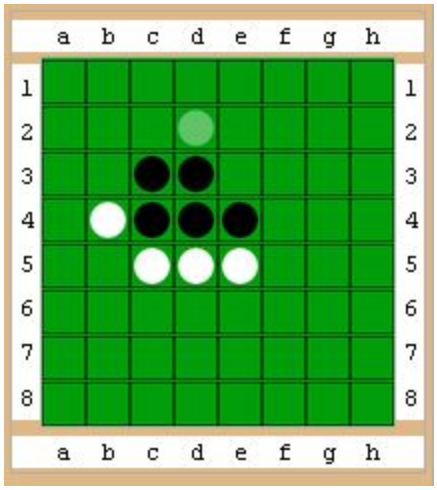
- If black plays at location d3, then the white disk at d4 is outflanked by d3 and d5. Hence, that disk would be flipped to black:

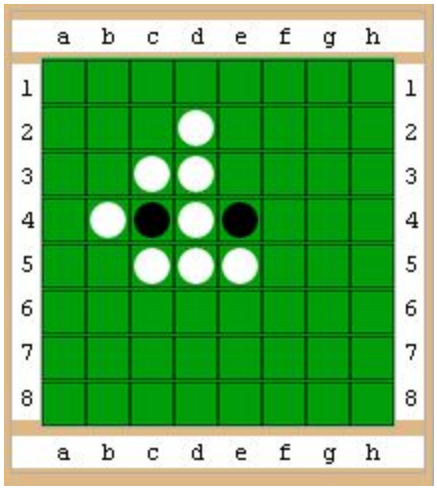


- If white subsequently plays at location c5, then the black disk at d5 becomes outflanked and flipped to white as follows:



- If a play outflanks multiple rows of your opponents disks, ALL such rows are flipped. For example, below is a before-and-after of a game board where white plays at location d2. This play outflanks c3 diagonally because of the white disk at b4, and also outflanks d3 and d4 vertically because of the white disk at d5. All three black disks are flipped to white. Notice that the black disk at c4 is **not** outflanked by the new white disk at d4. **Only the originally-played disk can outflank your opponent's disks.**





- If a player has no available move that will outflank an opponent's disk, the player's turn is forfeited and the opponent gets to go again.
- If a player has a valid move, it must be played (i.e., the player cannot pass).
- When all spaces on the board are filled, or neither player has a valid move, the game ends. The player with the most disks on the board is the winner.