COMP 10050
Software Engineering Project 1
Assignment 2
Othello board-game
Dé Máirt, 30 Márta 2020

The goal of this assignment is to create a program that manages a game of Othello for two players.

This assignment must be done individually.

Introduction

Othello is a board game that is played on a board with 64 squares in an 8×8 arrangement using 64 black and white discs, black on one side, white on the other. Players are assigned black or white and take turns to place a disc with their assigned colour facing up. After each play, any discs with the opponent's colour displayed that are in a straight line bounded by the disc just placed and another disc displaying the current player's colour are turned over to display the current player's colour. The winner of the game is the player who has the majority of discs turned to display their colour when the last move is made.

Othello is based on an older board game, **Reversi**, that was invented in 1883. The Othello game was patented in Japan in 1973. It differs from Reversi in two main ways. Firstly, in Othello, the first four pieces are placed in the centre of the board in a standard pattern, whereas Reversi starts with an empty board. Secondly, in Reversi, the game ends as soon as one player cannot make a move, whereas in Othello a player without a move just passes and the other player moves again.

A description of the game can be found at `https://en.wikipedia.org/wiki/Reversi`. The rules of Othello are described there and at
`https://www.ultraboardgames.com/othello/game-rules.php` and
`https://www.mastersofgames.com/rules/reversi-othello-rules.htm`.

You can play Othello against a computer at `https://www.eothello.com`,
`https://www.mathsisfun.com/games/reversi.html`,
`https://www.othelloonline.org` and `https://www.webgamesonline.com/reversi`.

Playing Othello

The game begins with four discs placed in a square in the middle of the grid, two with their black side facing up, two with their white side facing up, with the two discs with the same colour on a diagonal with each other. The convention is that the discs with the black side facing up are to the north-east and south-west as both players look at the board. Figure 1 shows the starting position of a game of Othello[1].

[1] Diagrams and positions are taken from or based on those on the Wikipedia page, `https://en.wikipedia.org/wiki/Reversi`.

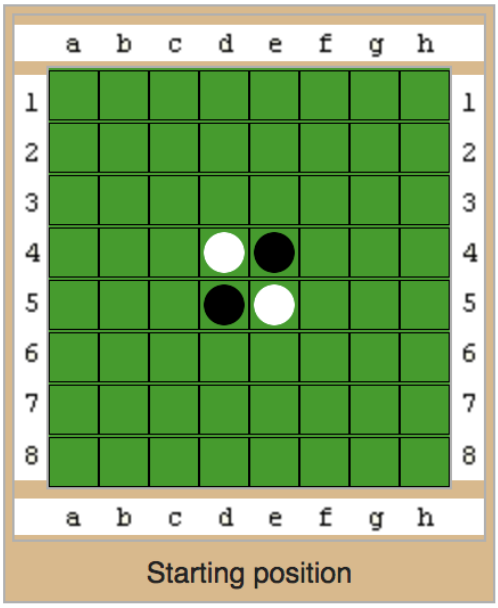
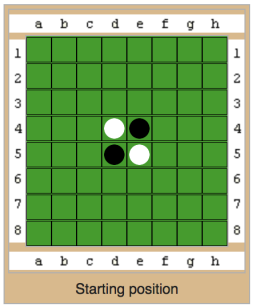
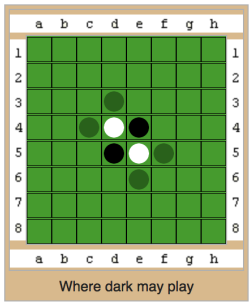


Figure 1: Starting position of game of Othello.

Each player must place a disc in such a position that there exists at least one straight (horizontal, vertical or diagonal) occupied line between the new disc and another disc of that colour, with one or more contiguous discs of the other colour between them. The player assigned black moves first. For Black's first move, the possible moves are indicated in Figure 2b.



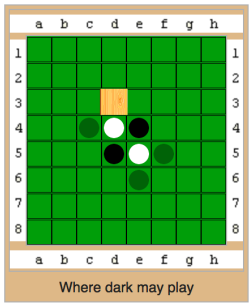
(a) Starting position.



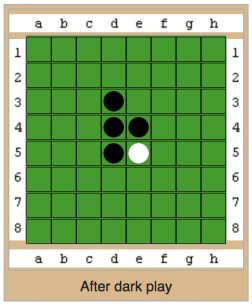
(b) Possible moves for Black.

Figure 2: Black's first move.

After placing their disc, Black turns over (flips to black, "captures") all white discs lying on a straight line between the new piece and any "anchoring" black discs. Assuming that Black places a disc in square **d3** (Figure 3a), they will capture the white piece in square **d4** and the board will be as presented in Figure 3b.



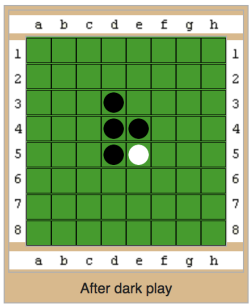
(a) Black chooses to play in square **d3**.



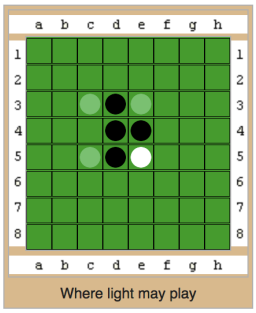
(b) Board after Black plays.

Figure 3: After Black's first move.

Now White plays. White operates under the same rules, with the rôles reversed: White places a white disc, causing at least one black disc to be flipped. For White's first move after the move just made by Black, the possible moves are indicated in Figure 4b. Assuming that White places a disc in square **c5** (Figure 5a), they will capture the white piece in square **d5** and the board will be as presented in Figure 5b.

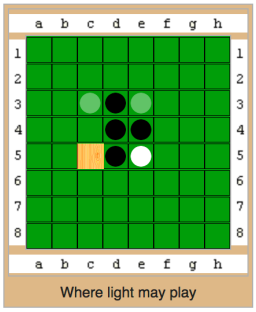


(a) Board after Black's first move.

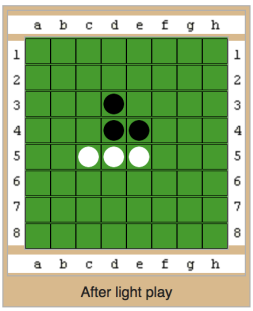


(b) Possible moves for White.

Figure 4: White's first move.



(a) White chooses to play in square **c5**.



(b) Board after White plays.

Figure 5: After White's first move.

Players take alternate turns. If one player cannot make a valid move (ie they cannot place a disc to capture at least one disc of the other colour), they must "pass" and play passes back to the other player. When neither player can move, the game ends. This occurs when the grid has filled up (all

64 discs have been placed) or when neither player can legally place a disc in any of the remaining squares. In the latter case, it means the game may end before the board is completely filled. This may occur because one player has no discs remaining on the board in their colour or because neither player can place a disc to capture at least one disc of the other colour. A player cannot pass if they have a valid move, even if that move disadvantages them.

The winner of the game is the player who has more discs on the board when the game ends. A draw or tie is possible.

Program functionality

Your program should have the following functionality:

- Ask each player to provide their name as input from the keyboard and assign this name to the player;
- Assign players a disc colour;
- Keep track of each player's score;
- Initialise the components of the game, eg initialise the board, initialise the score, set Black to be the first player to move;
- Request a move from each player in turn. A move should be entered at the keyboard. A move can either be **p** (for "Pass") or in the format `ld`, where `l` is a letter in the range **a** to **h**, and `d` is a digit in the range **1** to **8**;
- Check that the entered move is valid and, if it is, effect the move. If the move is not valid, request another move;
- A player should only be allowed to Pass if they have no valid move;
- Update and print the board after each move;
- When the game is over, establish which player has won, print out the winner, the score and the final board. The date and time of the game and the result of the game (ie the winner, the score) should also be appended to a file `othello-results.txt`.

Program code

You should use structures to represent elements of the game, eg the board, the players, etc. For example, the struct for the board would include a 2-Dimensional array, as well as the score and information on the next player to play.

Code requirements

- Use functions where possible;
- Use data structures to represent the board, the players, etc;
- Break your code into independent modules;
- Comment your code;
- Use a Git repository;
- In your repository include a text file describing:
  - How you decided to implement the board, the players, the disks...
  - How you decided to implement the game logic
  - ...

Evaluation Criteria

Your solution will be graded according to the following criteria:

- Your code should be well commented and appropriately divided into modules;
- Your work should be placed in the Git repository;
- The submitted text file should describe your design choices;
- Implementation of the players;
- Implementation of the board;
- Implementation of the game logic;
- Initialisation of the game elements;
- Correct printing of the board;
- Correct determination of the winner;
- Writing of result of game to file on disk.

Example

The following is an example of how the game should start:

```
$ othello
        ***  Welcome to Othello!  ***

Enter name of Player 1 (Black): John

Enter name of Player 2 (White): Tigger

        Score: John (Black) 2:2 Tigger (White)
            --- --- --- --- --- --- --- ---
        1  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
        2  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
        3  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
        4  |   |   |   | W | B |   |   |   |
            --- --- --- --- --- --- --- ---
        5  |   |   |   | B | W |   |   |   |
            --- --- --- --- --- --- --- ---
        6  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
        7  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
        8  |   |   |   |   |   |   |   |   |
            --- --- --- --- --- --- --- ---
             a   b   c   d   e   f   g   h
```

You must upload your submission by 23:59 on Friday, 7 May.