Cairo University
Faculty of Computers & Artificial Intelligence
Department of Computer Science
Artificial Intelligence Course

Project: Othello Board Game

- Students must work in **groups of 3-5** for their project. Students have to be from the **same lab** or from another lab taught by the same TA.

- The game should be implemented to be played in **Human vs. Computer** mode. The game that doesn't include this playing mode will not be graded.

- Students should utilize the **alpha-beta** pruning algorithm to decide on computer moves. No other algorithms allowed in this project.

- It is allowed to use **Prolog** or **Python** only.

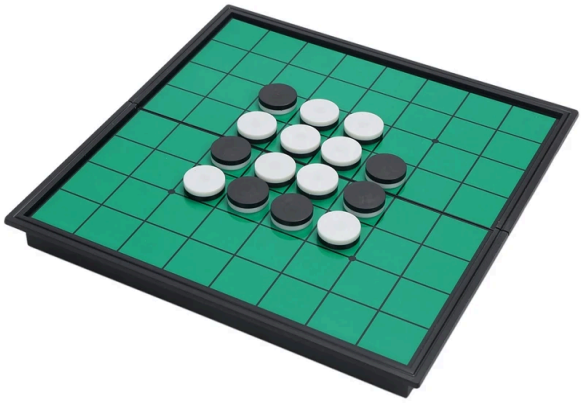
- Each project is graded based on the availability of:
    - A **game controller** that organizes the game by switching roles between the two players, receives the user's move, updates the game board, and declares the “End of Game”.
    - Suitable knowledge **representation** of the game state.
    - Adequate **utility function** that evaluates current game state with respect to a given player.
    - **Alpha-beta** pruning algorithm implementation. (You are allowed to use the draft implementation that was illustrated in your lab)
    - Support for different **difficulty levels** (Easy, Medium, Hard) each characterized through the depth of the algorithm. Eg: Easy 1, Medium 3, Hard 5.

- **Bonus:** User interface in any language (Java, C#, Python, etc.)

- Projects submitted will be checked against each other and against possible implementations on the web. Similar projects will not be discussed.
  Cheating Policy: **Negative the project grade.**

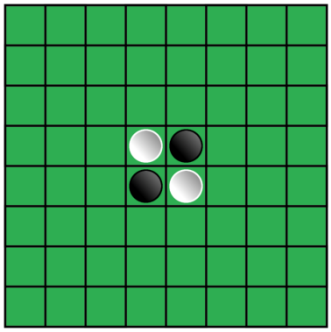
About Othello

Othello is a strategy board game for two players, played on an 8×8 uncheckered board. Two players compete, using 64 identical game pieces ("disks") that are white on one side and black on the other. Each player chooses one color to use throughout the game. Players take turns placing one disk on an empty square, with their assigned color facing up. After a play is made, any disks of the opponent's color that lie in a straight line bounded by the one just played and another one in the current player's color are turned over. When all playable empty squares are filled, the player with more disks showing in their own color wins the game.



Game Setup

- Initially, the board is set up by placing two black disks and two white disks at the center of the board exactly as shown in the opposite figure. The game always begins with this setup.
- Then, the remaining 60 disks are divided between players such that each player has 30 disks.



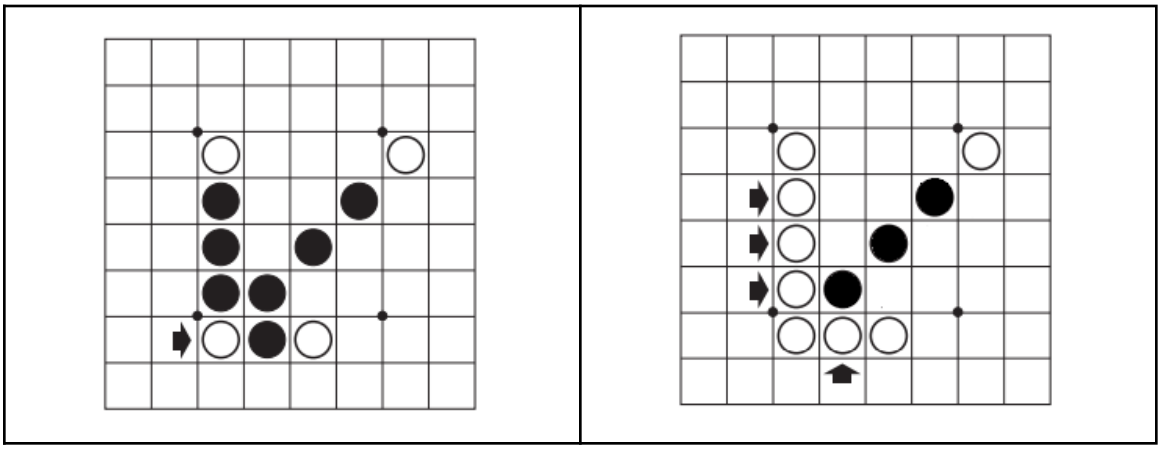
How the game goes

1. When it’s a player’s turn, he must look for an empty square on the board that is adjacent to one of the opponent’s pieces. The player can add his disk there with his color facing upwards.
2. If the newly added disk encloses a straight horizontal row or vertical column of the opponent’s pieces between two disks of the player’s color, then it’s a legal move known as “outflanking”.

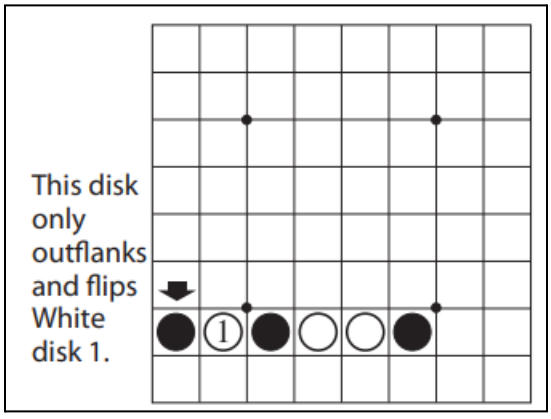
3. Once the player outflanks the opponent's disks, they are flipped over to the player's color (i.e. the player captured them). The disks now count as the player's pieces on the board even if he didn't originally play them.
4. After that the player passes the turn to his opponent to continue playing.
5. When it is no longer possible for either player to move, the game is over. Disks are counted and the player with the majority of their color showing is the winner.

Game Rules

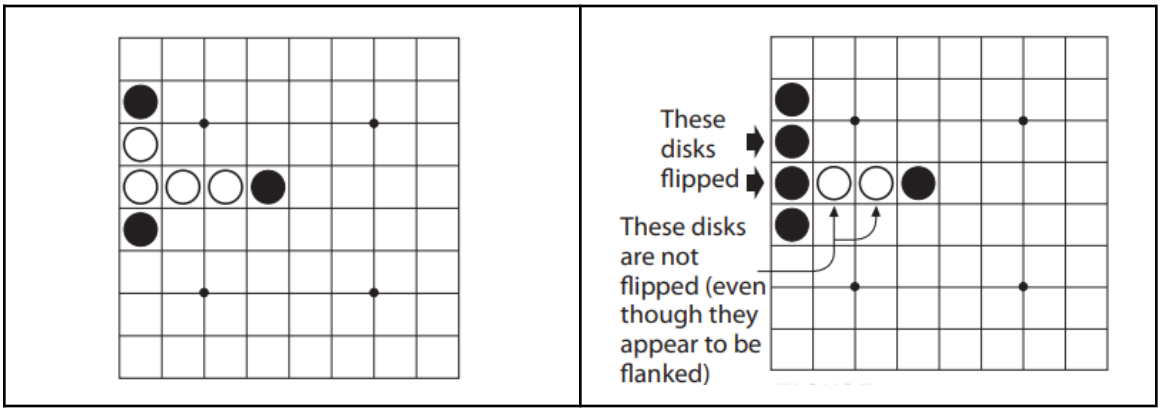
- Black always moves first.
- If a player cannot outflank and flip at least one opposing disk, they miss their turn and their opponent moves again.
- A disk may outflank any number of disks in one or more rows in any direction (either horizontally or vertically) at the same time.



- Players may not skip over their own color disks to outflank an opposing disk.



- Disks may only be outflanked as a direct result of a move and must fall in the direct line of the disk placed down.



- The game may end if a player runs out of pieces.

Note: *You can play the game online to get familiar with how it works:*

https://www.eothello.com/

Required Deliverables

You are required to submit **one zip file** containing the following:

- Your **code** (either **.pl** or **.py** file(s))
- A **text file** containing the team members' **names, IDs and groups** and a **link to a video** uploaded on your drive.

The zip file should follow the following naming convention: **ID1_ID2_ID3_Group**

In the video, each team member should speak (stating his/her ID at the start) and take one of the main parts in the code (representation, controller, alpha-beta, moves and utility) and explain it. The video should also include running the game as a demo showing how players take turns and how the computer automatically plays. The overall duration of the video should not exceed 10 minutes.

Grading Criteria

Board, disks and state representation	**1.5 marks**
Updating and printing board after each move	**1 mark**
Player switching (human vs. computer)	**0.5 marks**
Representing possible moves	**1.5 marks**
Utility function	**1.5 marks**
Difficulty levels (easy, medium, hard)	**1 mark**
Alpha-beta pruning algorithm	**1 mark**
GUI	**1 mark**

Possible Penalties

Failing to implement a two-player mode	**-1 mark**
Not including the group and IDs in the file's naming convention	**The assignment won't be marked**
Not submitting the video required in deliverables section	**-2 marks**
Team members are not from the same lab or not with the same TA	**-1 mark**
Team members are less than 3 or more than 5	**-1 mark**