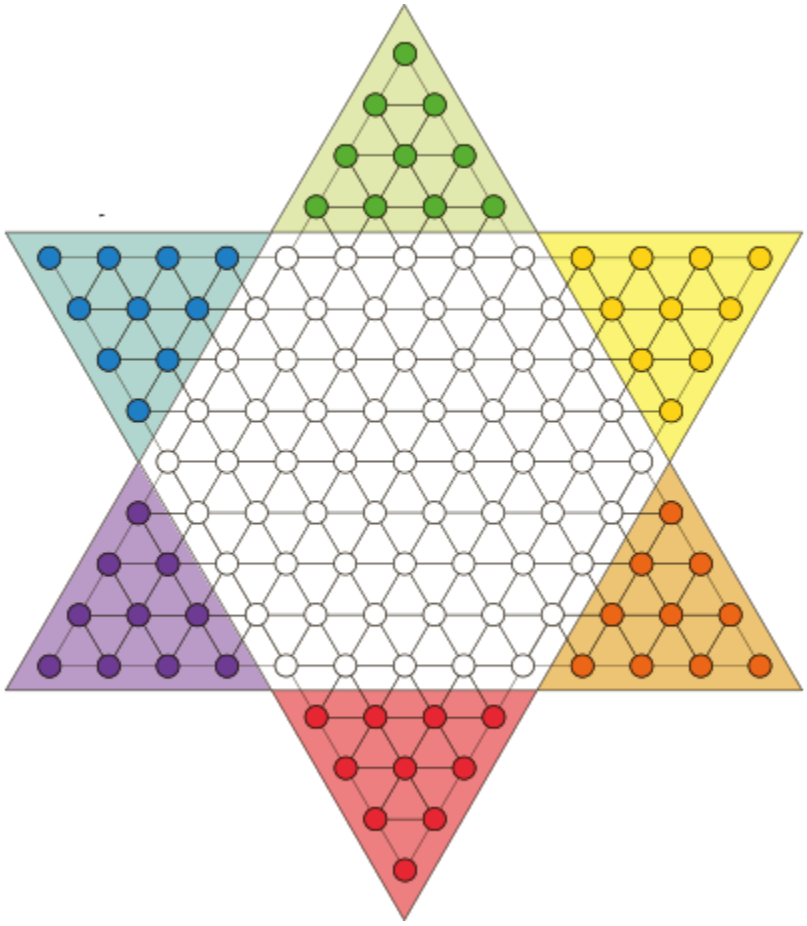
Cairo University
Faculty of Computers & Artificial Intelligence
Course: Artificial Intelligence
Semester: Spring 2022

Term Project on Games

- Students must work in groups of 3-4 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 don't include this playing mode will not be graded.
- Students should utilize the min-max algorithm to decide on computer moves. Alpha-Beta Pruning is a plus. 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:
  o A game controller that organizes the game by switching roles between the two players, receives the user's play, changes the game board, and declares End of Game.
  o Suitable knowledge representation for the game state.
  o Adequate utility function that evaluates current game state with respect to a given player. A positive utility denotes a good state while a negative one denotes a bad state. The larger the utility, the better the state. The utility function for player A typically equals negative utility function for player B.
  o Basic Min-Max implementation (You are allowed to use the draft implementation that was illustrated in your lab).
  o Support for different difficulty levels (Easy, Medium, Hard) each characterized through the depth of the min-max algorithm. Ex: Easy 1, Medium 3, Hard 5.
- Project Bonus:
  o User interface in any language (Java, C#, Python...etc)
  o Using Alpha-Beta Pruning
- Cheating Policy: Negative the project grade.
- Projects submitted will be checked against each other and against possible implementations on the web. Similar projects will not be discussed (Both original and copied projects).

Cairo University
Faculty of Computers & Artificial Intelligence
Course: Artificial Intelligence
Semester: Spring 2022

Project
Chinese Checker



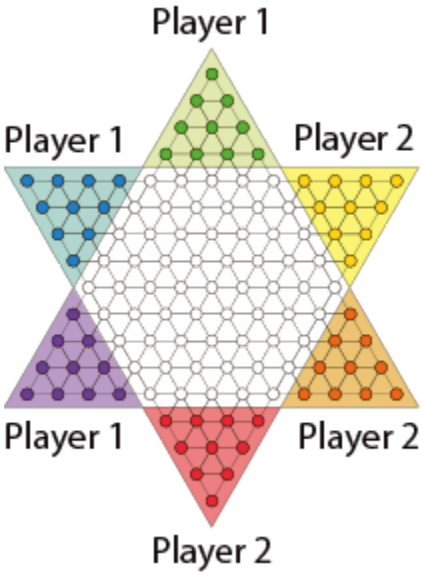
Description

Chinese checkers can be played by up to 6 players. However, we will implement only a 2 players game version for this project. The game board is basically a star with six edges. Each edge contains a different colored set of 10 marbles as seen in the above picture. The first player will own three colored sets and the second player will own the remaining three colored sets.

For example:

- Player 1 will own **Green**, **Blue** and **Purple** sets
- Player 2 will own **Red**, **Orange** and **Yellow** sets

Cairo University
Faculty of Computers & Artificial Intelligence
Course: Artificial Intelligence
Semester: Spring 2022



Object

The object of the Chinese checkers is to **get all of your marbles to the opposite point of the star**. The first player to do this wins.

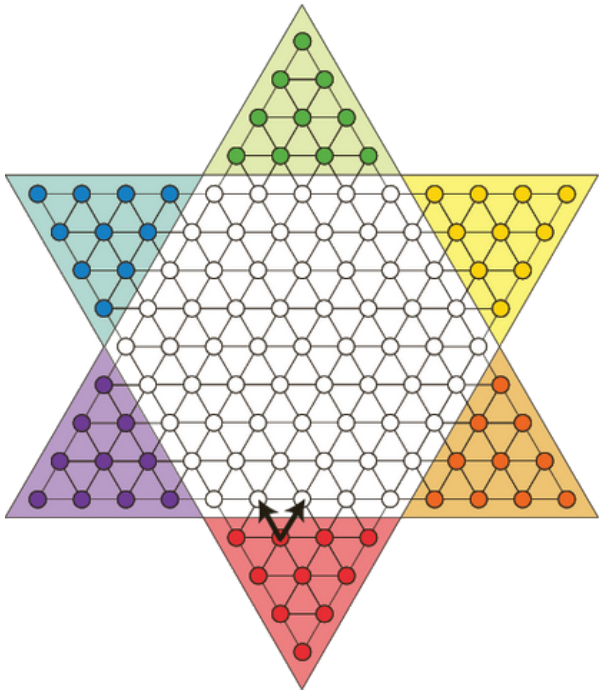
For example, For player 1 to win:

1. He must move all green marbles to the other opposite side of the star (Replacing red marbles).
2. He must move all blue marbles to the other opposite side of the star(Replacing orange marbles)
3. He must move all purble marbles to the other opposite side of the star(Replacing yellow purple marbles)

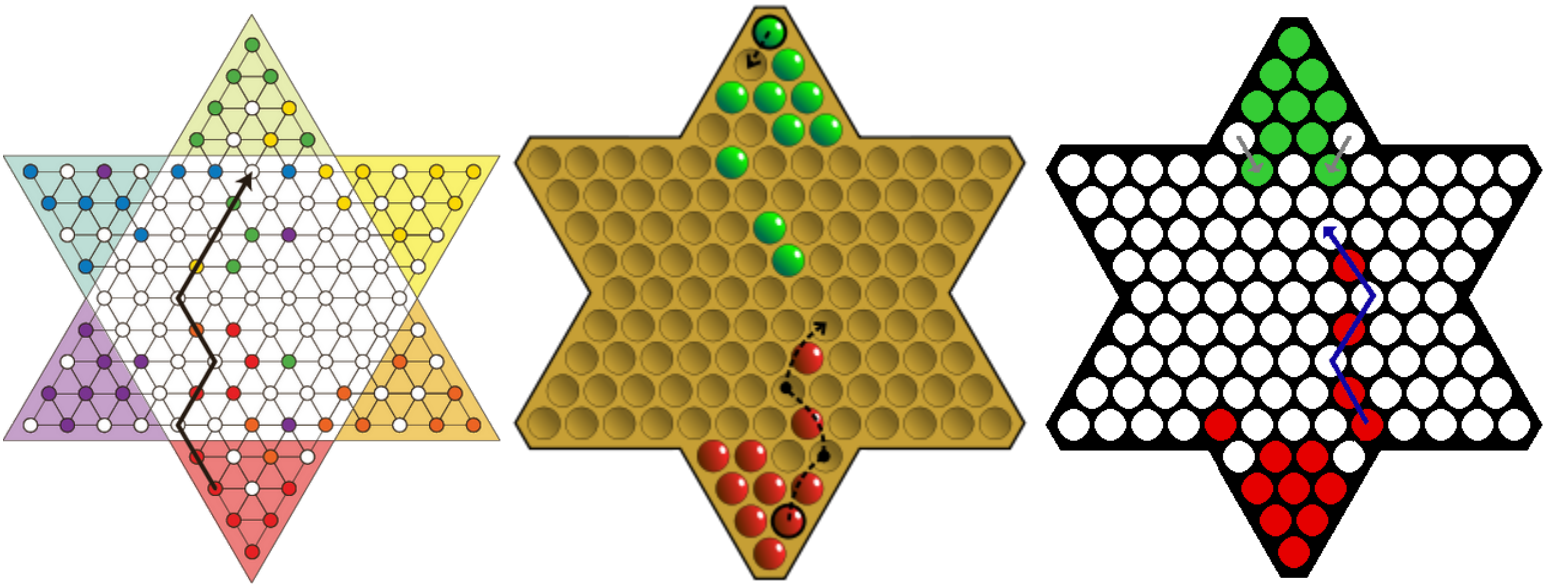
How the game goes on

Players take turns moving their pieces across the board. During a player's turn:

1. They can move their piece to any **adjacent empty hole**.



2. A player's piece can also **hop over an adjacent piece, into an empty hole**. That piece can be their own colored piece or the opponent piece. The player can **continually hop over subsequent pieces** during that turn, and in any direction, as long as there are empty holes on the other side of those pieces.



In the above picture, it's acceptable to drop the marble while hoping **at any empty hole on the black arrows**.

For more details about the movement rules, please review the below videos
https://www.youtube.com/watch?v=PM-VuNLYI0s
https://www.youtube.com/watch?v=E0vSvWdNiUg

You are required to implement the checker game using these aspects:

1. Star gameboard as seen in the above figures
2. 6 colored sets. Each set contains 10 marbles
3. 2 Players, Human vs PC. Each player will own 3 colored sets and human plays first
4. After each move, print the updated game in well organized format
5. Support 3 difficulty levels (Easy, Medium and Hard). The difficulty selection is done upon launching the game and before making the first move
6. Use Python or Prolog
7. Use Minimax or (Alpha-Beta is bonus)
8. Implement GUI (Bonus)

Delivery data and time will be decided on Blackboard. No late submissions are accept