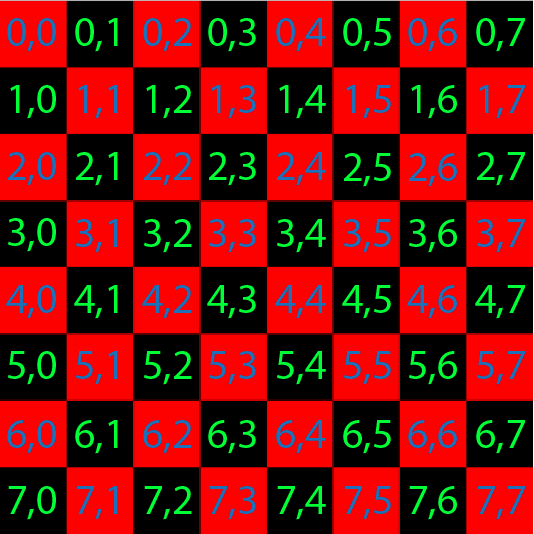
Number board with (x,y) coordinates.

Class player{

playerUsername;

pieceColor; -assign either black or White using PieceType enum

int numPieces;

getUsername();

setUsername();

setColor(); -determine if the player has the black or white pieces

setNumPieces()- each time a player loses a piece, decrease numPieces by 1, when either player reaches 0 the game is over

}

Enum PieceType{

RED, WHITE;

}

Class piece{

String piece\_Id- name of the piece

PieceType type- color of the piece, either WHITE or BLACK

int currentX;

int currentY;

int newX;

int newY;

boolean isKing- will become true when the piece is a king

movePiece(int oldx, int oldy, int newx, int newY){

* use current and new position to make sure move is legal. When the x coordinate is odd, the y coordinate has to be even and vice versa.
* if the piece is a king, it can move in any direction, otherwise it can only move forward
* Depending which side of the board the piece is on, make sure it is only moving forward unless it’s a king.
* check for a jump. In a jump the row changes by 2

}

makeKing(){

* change piece to king title

}

}

Class Checkers(){

gameType;- either human-human, human-computer, or computer-computer

player\_Started; -contains a reference to the player who made the first move and is black

currentPlayer- keeps track of which player is making the current move

turnNumber; Static variable that increases by one each time a move is made

result; - keeps track of the result of the game

checkGameStatus(){

* check each players number of pieces, if one player has 0 pieces the game is over and the other player wins
* check if there are possible moves still, if not game is a draw

}

}