

#### Movement Methods:

CheckNextMove - checks the square the current piece wishes to move to and determines if the space is occupied.

IsValidMove - checks to make sure the next move is allowed at all (basic check to ensure pieces move in diagonal motions and not backwards unless "kinged")

CheckCurrentSquare - analyzes the current square's "number" to update on the piece.

#### Status Methods:

UpdateBlackPieces - called when a black piece is attacked by a red piece or when the black player kings.

UpdateRedPieces - same conditions as above method under respective conditions.

CheckKing - called when players make a kinging move for themselves, will return that piece as kinged.

GameOver - will be updated with Update methods, checks for loss of all black/red pieces or lack of valid moves for either side.

Depending on conditions, players are awarded wins, losses, or a draw.

#### Initialize Methods:

GenerateBoard - method to create/display the board in which the game is to be played.

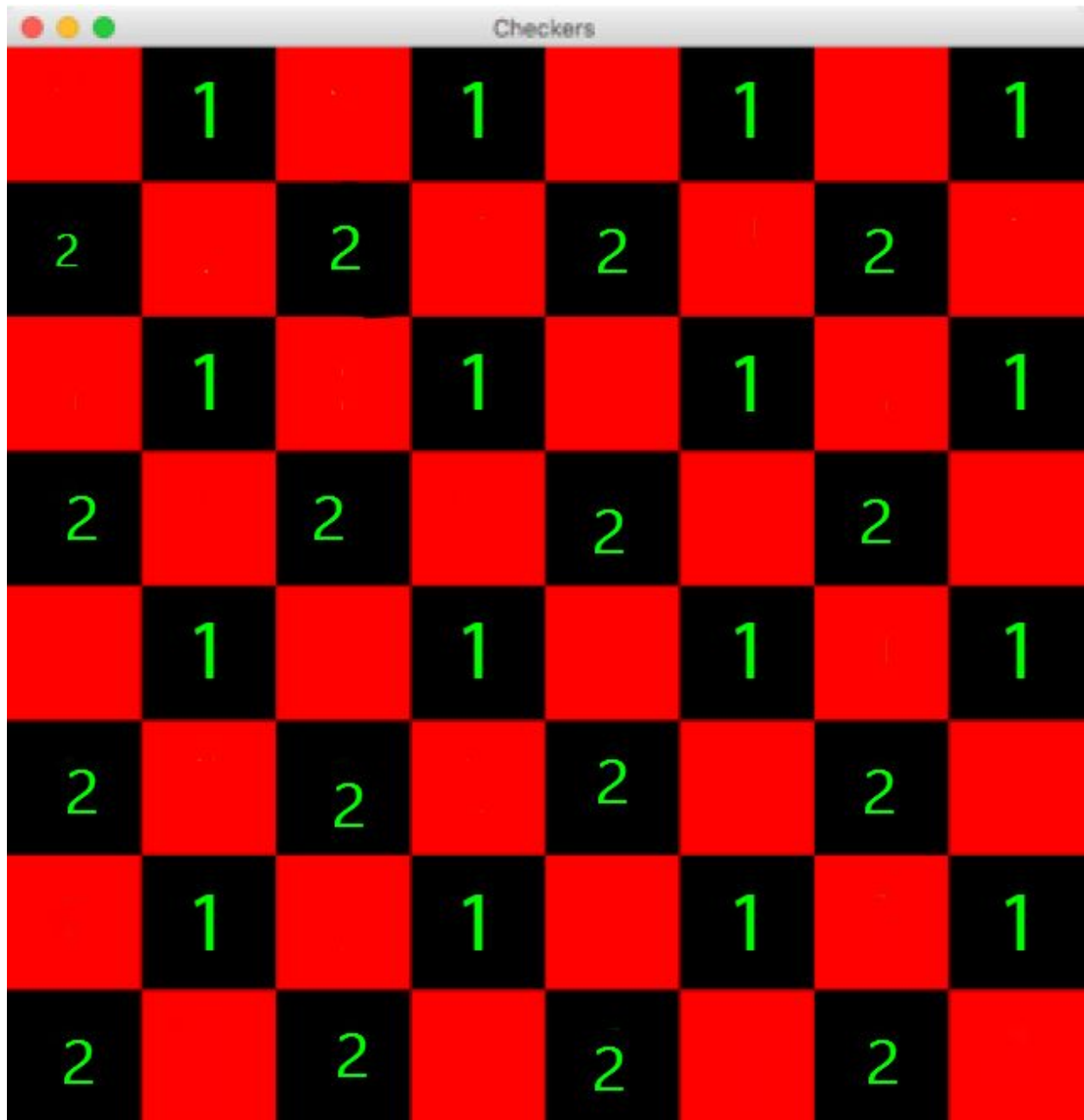
GeneratePlayers - populates the board with the player pieces with which to play the game.

GetPlayerNames - at initialization of the game, the players will be asked to input their names.

The first name input will be displayed top, second player will display bottom.

RandMoveFirst - the name of the method describes how the players are picked for who goes first; randomly. This is similar to how online chess games perform. The player picked to move first gets assigned the red pieces, the second player is assigned the black.

SelectGameMode - lets the users select if they want a two-player game, a player vs computer, or computer vs computer.



#### The Board:

The board is arranged such that only the black squares have a number assigned to them. These are the only squares which will have pieces on them. The rows are ranked by two numbers, 1 and 2. Rows starting with Red squares have Black squares marked with 1's, likewise if the row starts with a black square they are marked with 2's. Pieces on 1's can only move to 2's, but cannot move backwards unless kinged. Red players starts on top, black player on bottom.

#### Utility Methods:

CheckAdjacent - the method will check the squares adjacent to the focused square, which will be at the time this method is called, the square which the player is moving from.

CheckRow - the method analyze the row by which the player wishes to move into. If red, and if current row number is greater than the row number which the player wishes to move into, unless the player is kinged, they cannot proceed. Likewise conditions apply to the black pieces.