Ex3:

Before beginning our joint project we sat and discussed the pros and cons of each individual project.

For the following assignment we decided to combine both our projects to create a better one.

First, we built our turn logic almost from scratch using a very simple pattern which includes getting the possible moves list, getting the move from player and playing the move on the board.

Using a twist on Shani's implementation for the function the gets the moves list, we first look for tiles of same type on the board, and then check for an empty spot in a direct direction of the spot.This implementation was different than Liora's, which looks for all the empty tiles and then checks for the spots that lead to a tile of same type.

We decided to create a move class to hold the location of the move on the board and a variable to determine the number of tiles that will flip given the specific move was chosen (it mainly helps us calculate the algorithm for the AI player but can be useful for the HumanPlayer to order the printed moves by the score he will get from each move).

The player's move is taken from a method which the player implements in his/her class. Last, the turn sends the player's chosen move and game board to a function to flip the tiles, which was taken from Liora's code.

The turn function returns the current game status, which is an enum representing the game's status (full board, game over, in progress, etc), which was taken from Liora's code.

In addition we added a GameUI interface, so the game can run on console or on a GUI in the future- with minimal code changes, this was taken from Shani's code.

We added an enum Tile to represent the tiles which are placed on the board, and preferred that implementation over a tile class or just straight forward chars.

Liora Zaidner 323742775

Shani Herskowitz 321659387