Focus(Domination) Software Engineering

By: Joseph Ogunbiyi

Student Number: 19367106

https://github.com/TLobaOriginal/Labs1/tree/master/Project2Focus

Focus is a game where the objective is for you to make sure your opponent cannot make a move. To do this we must follow the rules of the game. The rules go as such.

- Each player takes a turn (2 players in this case). Red is player 1 and Green is player 2.
- A player can move their piece(stack) by x or less amount of squares (min 1) x being the number of pieces in stack. A stack is limited to 5 pieces and the top piece means that the stack belongs the respective player. (The player with its colour).
- Should a stack exceed the limit the bottom piece must be removed. If the bottom piece is your colour then you keep it in your reserves and bring it back into play later. If the bottom piece is your opponent's then it is captured and is taken out of play for the remainder of the game.

The is designed in a way that although it is a square it has 12 pieces that are invalid or non – existent. These would be row 1: column 1, 2, 7, 8. Row 2: column 1 and 8. Row 7: Columns 1 and 8. Row 8: Columns 1, 2, 7 ,8. The rest are useable. We also have 16 empty squares on the board. We initialised and printed this using 'initialize_board' and 'print_board_update' respectively. 'CoinToss' picks at random whether player one or player two goes first.

Initialize player deals with initialising the total pieces for player 1 and 2 and their reserves and captured pieces. Their names are also labelled.

'WinnerChecker' is a function that checks to see if a winner is found and if not then the loop for the turns will continue.

ThisTurn: We used this to basically deal with the turn system, handling the reserved pieces logic and the board pieces logic. 'Available' serves as a function to check if there is a piece on the board.

We get a player to pick the coordinates for the piece they want to move. But to move we use the famous game movement keys W, A, S, D. This was all incorporated in logic.c

We also have a separate function for reserved pieces being entered into the game. It just requires coordinates.

We incorporated Stack.c in order for pieces to be stacked on top of each other. We used linked list for us to deal with this. We made 2 separate ones. One for pieces on the board being moved and one for reserved pieces being added into the game. In each function we have the functionality for removing the bottom piece added.