

Assignment 2 implementation

Dawid Skraba student no.19433692

Repository: <https://github.com/DawidSkra007/Project2.git>

The focus of this project was to make a board game called 'Domination'. This is how it works:

- Firstly, the players are initialized, the program receives both of the players' names and it will output which player is which colour. The board is also initialized under the given specifications. The board consists of displaying the invalid spaces at the corners of the board which will not be in play, it shows empty spaces where no pieces are placed at the start but can be played on and finally it will show you the colour of the top piece on the board and how many pieces are stacked on that space. The co-ordinates of each piece are also added to make it easier for users to specify which piece they want to move etc.
- In the main the function 'game_end' is constantly checked after every move to see if the game is won. This is done by the function seeing what the colour of the top piece is on every space on the board. If there are more than zero, red and green pieces, then a move is possible, and the game is played on. If the amount of one colour pieces (e.g. red) is zero and that player doesn't have any captured pieces, then the other player wins (e.g. green).
- If the game can still be played turns are implemented so each player can make a move. The updated board is printed after every turn so the player can easily see what move to make. It prints who's turn it is and it gives you a choice of what to do. Either move a piece or move a captured piece onto the board. If the player wants to move a piece, he/she will be asked to input the co-ordinates of the piece that they want to move. An error message will be printed if the piece you picked is not your colour and the question of what to do will be printed again. When the player selects a legal piece, the number of moves is printed (The number of pieces in the stack) and the question of what move they want to make will be printed that many times unless they want to stop moving. After picking what move you're going to make the co-ordinates are updated. The board is updated with the values and the number of pieces in that space is also updated. If the stack becomes bigger than five pieces as a result of a move, then the excess pieces are removed as the maximum number of pieces allowed per stack is five. If you wish to place a captured piece the program will execute a different set of instructions. If you don't have any captured pieces the question will be asked again and if you do, the co-ordinates will have to be entered, where you want to place that piece. The piece is added and the number of pieces on that space is incremented and the number of pieces the player owns is decreased.