Project Name: Alpha Merge

Introduction:

The project is to build a version of the popular 2048 game, that can be played on the console. Due to the limitations in the display of windows based terminals, the numbers will be replaced with letters, in this case 11 letters from A to K.

Game rules:

The game is played on a square 4X4 board. Each square can host a number, or in our case an alphabet. The player can make up/down/right/left moves and all the alphabet-blocks will move accordingly.

At the beginning the board starts with two A blocks and the rest of 14 blocks empty. When a player makes a move - say he/she presses right, all the blocks will move to the right most possible position within their row. The same goes for the other three directions.

If two blocks of the same alphabet collide during this movement (are next to each other and moving in the same direction), then merge to form the next alphabet. I.e. two A blocks will merge into one B block.

Every move of the player adds a new block in one of the empty blocks on the board.

The aim is to move the blocks so that they continue to merge into higher alphabets until we reach K.

The game ends when the player reaches K or when no more moves are possible.

Features:
   - Board size can be 4X4 or 6X6 or 8X8; Default is 4X4
   - Different alphabets will have different colors
   - Score, time and number of moves captured
   - Score stored for high score comparison

Python code:

Blocks class:
   - Instantiates a blocks with default value being A
   - Has attributes - value (A-K), position (row and column)
   - Color

Board class:
- Instantiates a board of nXn size on users choice
- And prints the board
- Has attributes
    - number of blocks- incremented every time a new block is introduces and decremented when blocks merge
    - Number of empty spaces
    - score

Moves class:
Takes in the current board and based on input moves blocks
- Loops over each block based on the direction of the move- if it is move right then loops over each block in each row from rightmost to left most, performing functions -
    - Merge if the blocks are the same: update the right block value
    - Delete the left block that got merged
    - Update blocks positions
- Once all moves are finished initiates a new block on a random square
- Attributes:
    - number of moves
    - Number of merges

User class:
Initiates a user as he/she starts the game. Takes in inputs - size of the board. Stores score of previous games

Game Class:
Keeps details of the game played in one sitting. By a given user.
Attributes: Game number, user, score etc.
Game class would initiate Board, Blocks and Moves classes.