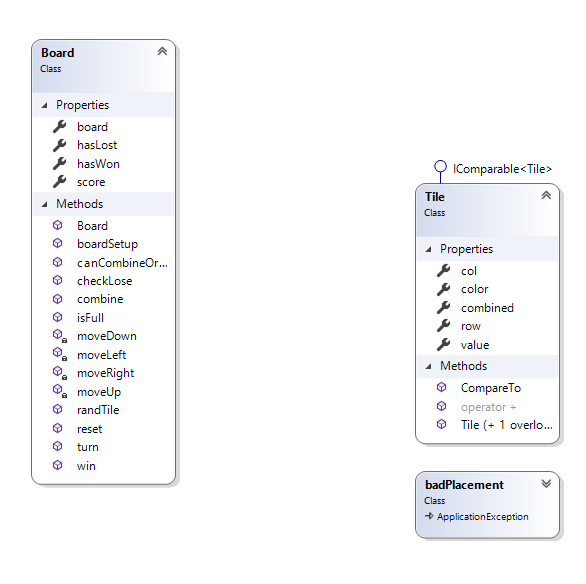
Windows App Dev Final

Caleb Grode & Aaron Borjas

Description:

Our project is a recreation of the game 2048 (<https://play2048.co/>), where the objective is to combine like tiles to reach the tile of 2048. Each tile is a power of two, and can be moved up, down, left, or right. The board is a 4x4 grid of these tiles. Every time a key action (up down left or right) is performed, tiles can combine along that path or simply move in that direction. Along with that, a new tile, either a 2 (90% chance) or a 4 (10% chance), spawns in to a random position on the board. If the board is filled with tiles and no tile combinations (in any direction) can be completed, the player loses and must restart. If the player does manage to reach 2048, they have the option of continuing until they lose or choose to reset. Resetting can be done at any point of time in the program. The score is also tracked, being added to when two tiles combine for the value that those two tiles combine to.

UML:



Pseudocode:

**Class Board:**

Public void combine(Tile one, Tile two, int row, int col) {

Add the values of tile one and two to score

Make a new tile with a value of tile one’s value plus tile two’s value

Get rid of both tile one and tile two

Set new row and col of the new tile based on one and two

Check if the new tile is 2048, aka a win

}

Public bool canCombineOrMove(Tile t, int row, int col) {

If the tile to the left is not out of bounds and the two tiles are the same or one tile is empty space

Return true

If the tile to the up is not out of bounds and the two tiles are the same or one tile is empty space

Return true

If the tile to the right is not out of bounds and the two tiles are the same or one tile is empty space

Return true

If the tile to the down is not out of bounds and the two tiles are the same or one tile is empty space

Return true

Return false

}

Public void randTile() {

If the board isn’t full

Loop until a tile has been created

If a randomly chosen location is empty

Place a new tile there (10% for 4, 90% for a 2)

Otherwise,

Decrement int used in loop one, so another tile can be made

}

Public void move\_\_ () { //“\_\_” signifies the direction of the function; moveUp, moveDown, moveLeft, moveRight

Track number of “open spaces” in the board

Loop through the board

Increment numOpen *if* the value at row,col is empty

Otherwise,

If two tiles can be combined

Combine the tiles to a position

Otherwise move the tile in the specified direction

}