Computer Programming II

In this project, you will implement the following game by using JavaFX framework. This game aims to destroy all boxes with the highest score and minimum clicks. The initial state of the game can be seen in the figure below:

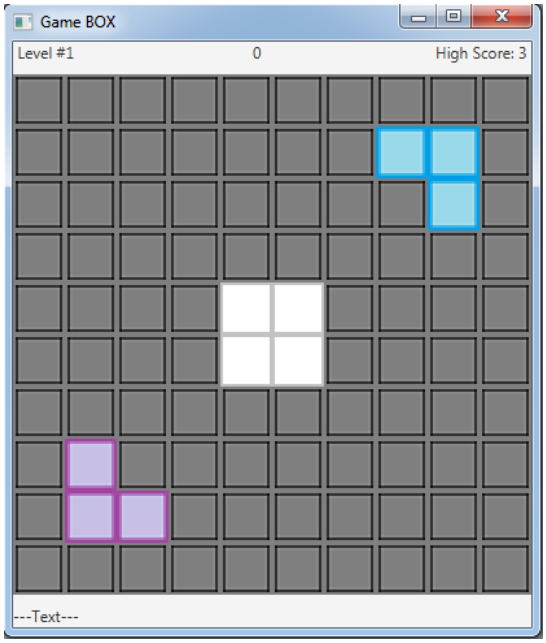


Figure 1: Game Screen

The game grid consists of 10x10 boxes. Each box is identified by using zero-based [rowIndex, columnIndex]. A sample game screen is shown in Figure 1. The user will use the mouse to destroy the boxes. When a user clicks on a box to destroy, it will also affect neighboring boxes (right, left, up, and down boxes).

For the project, you will implement this game with the following properties:

- There will be several levels; the user will not be able to play the next level without completing a previous one.
- Each level will be created based on an input file that will be provided separately.
- Each of these input files should be located in the “levels” folder in your current directory.
- There is no limit for the total number of levels (i.e., there can be 3-levels, 5-levels, 10-levels, etc.).
- The user will use the mouse to destroy the boxes.
- While you are constructing the game board, you may use images for each tile.

GAME LAYOUT

The game layout should consist of three parts as shown in Figure 2.

Top Pane: It must include current level number, current score, and highest score information.

Center Pane: The game board size should be 10x10 boxes.

Bottom Pane: It should display information about the location of the clicked box, neighbor boxes, and obtained score value.

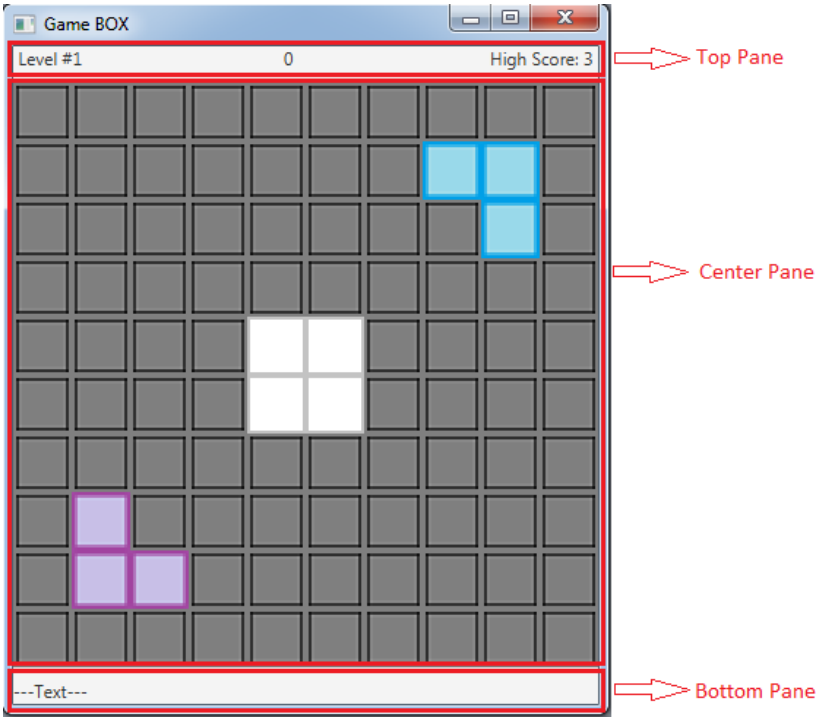


Figure 2: Game Layout

GAME BOARD CREATION

Level appearances will be prepared using input files. The appearance of the first level (Level 1) is illustrated in Figure 3. There are a total of 100 boxes (10x10) on the game board, each with different characteristics. The game board must be created according to the data in the input file.

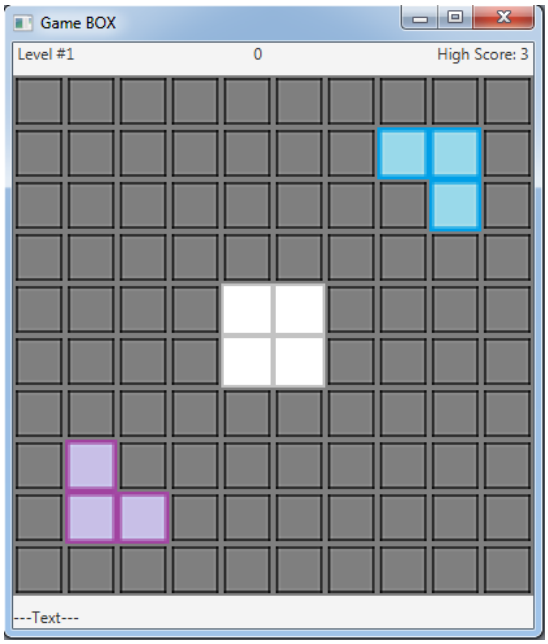


Figure 3: Game Screen for Level 1

Level 1 Input File:

Box type, rowIndex, columnIndex
Empty,4,4
Empty,5,5
Empty,4,5
Empty,5,4
Mirror,1,7
Mirror,1,8
Mirror,2,8
Wood,7,1
Wood,8,1
Wood,8,2

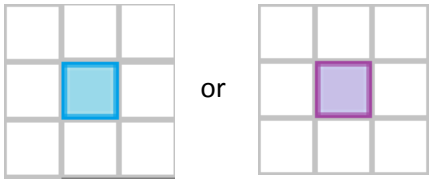
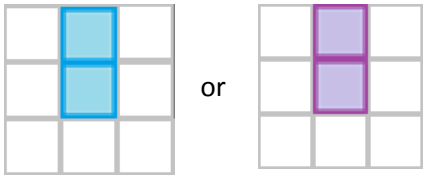
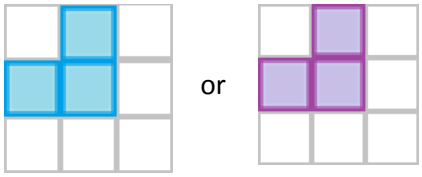
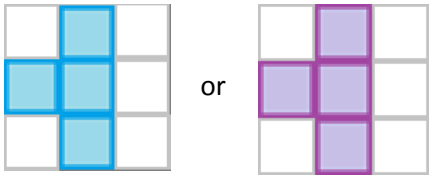
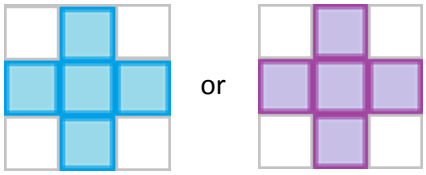
Level Input File

The input file contains the box type and box row-column index on the game board, each separated by commas. The default box type is *Wall* so missing location box types should be *Wall* type.

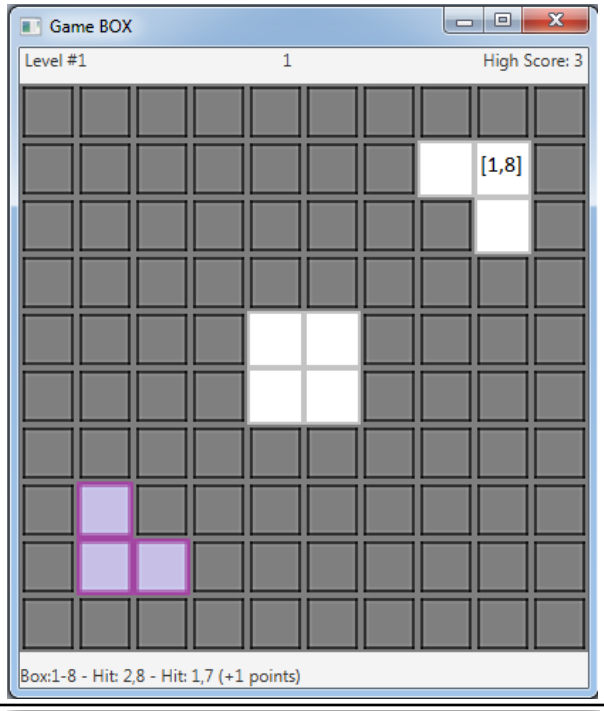
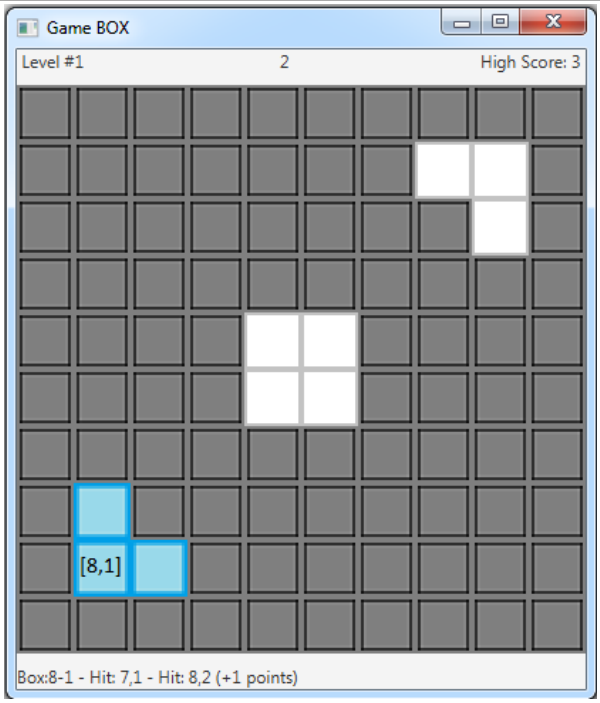
BOX TYPES

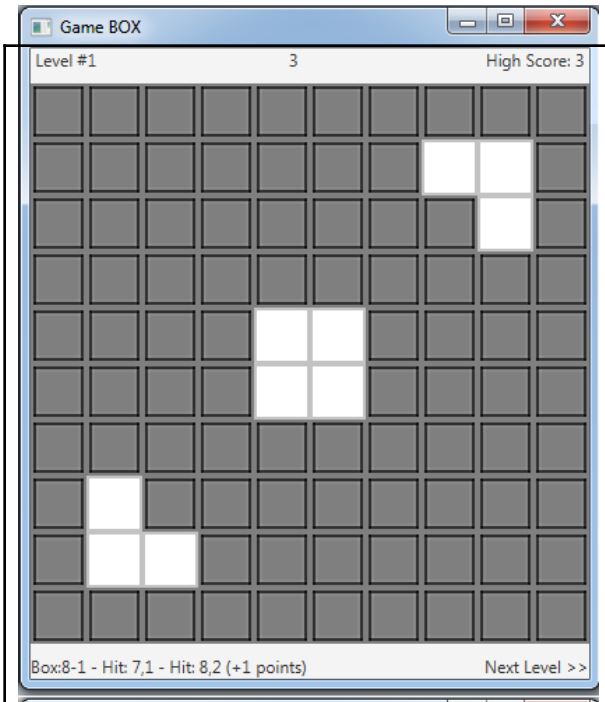
	Empty Type Box - It cannot be destroyed
	Wall Type Box: - It cannot be destroyed
	Mirror Type Box: - Durability (or life) value is 1. It returns to the empty box type if the click or hit occurs.
	Wood Type Box: - Durability (or life) value is 2. It returns to the mirror box type and durability becomes 1 if the click or hit occurs.

GAME POINTS

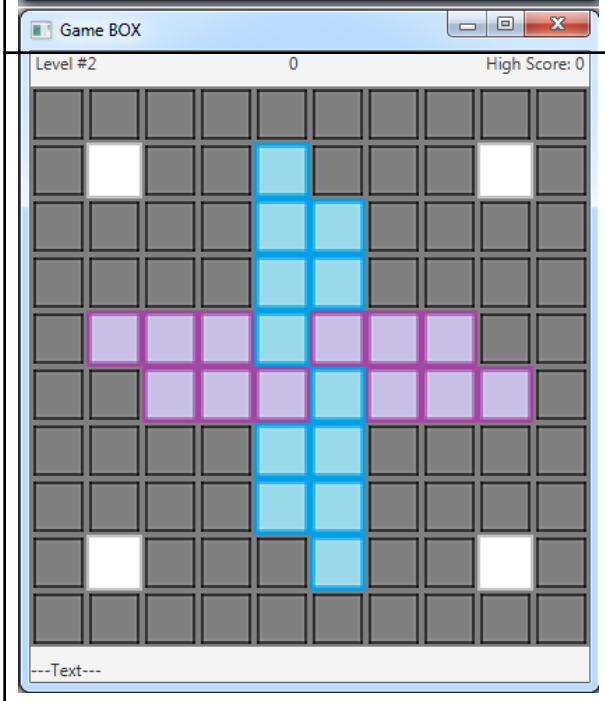
	1 box destroy: -3 points - Click 1 box
	2 box destroy: -1 points - Click 1 box and hit 1 neighbor box
	3 box destroy: +1 points - Click 1 box and hit 2 neighbors box
	4 box destroy: +2 points - Click 1 box and hit 3 neighbors box
	5 box destroy: +4 points - Click 1 box and hit 4 neighbors box

Sample Game Scenario

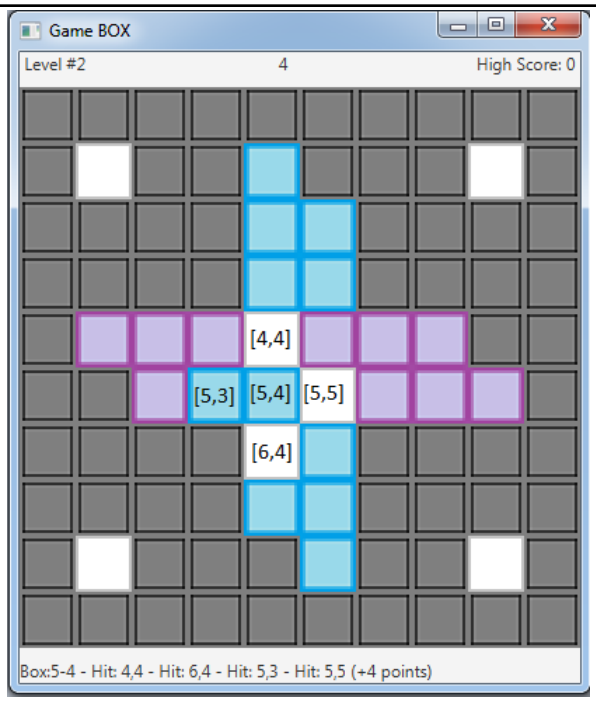
	- Box at [1,8] and neighboring boxes (at [2, 8] and [1, 7]) are Mirror type originally and durability (life) value is 1. - Suppose that the user clicked on the box at position [1, 8] and hit also two neighboring boxes (at [2,8] and [1,7]). - After the box click at [1, 8], durability value of boxes at position [1, 8], [2, 8] and [1, 7] decreased by 1. Thus, now the durability value is 0 and the box type changed to the *Empty type* (as it is seen in the figure). - And with the click action, 3 blocks are affected (clicking the box at [1, 8] and hitting the boxes at [2, 8] and [1, 7]), the user gets +1 point.
	- Then, the user clicked the box at position [8, 1] and hit the boxes at [7,1] and [8,2]. - Box at [8, 1] and neighboring boxes (at [7, 1] and [8, 2]) were *Wood type* originally and durability (life) value was 2. - After the box click at [8, 1], the durability value of boxes at position [8, 1], [7, 1], and [8, 2] decreased by 1. Thus, now the durability value is 1 and the box type changed to the *Mirror type (as it is seen in the figure)*. - And with the click action, 3 blocks are affected (clicking the box at [8, 1] and hitting the boxes at [7, 1] and [8, 2]) and the user gets +1 point.



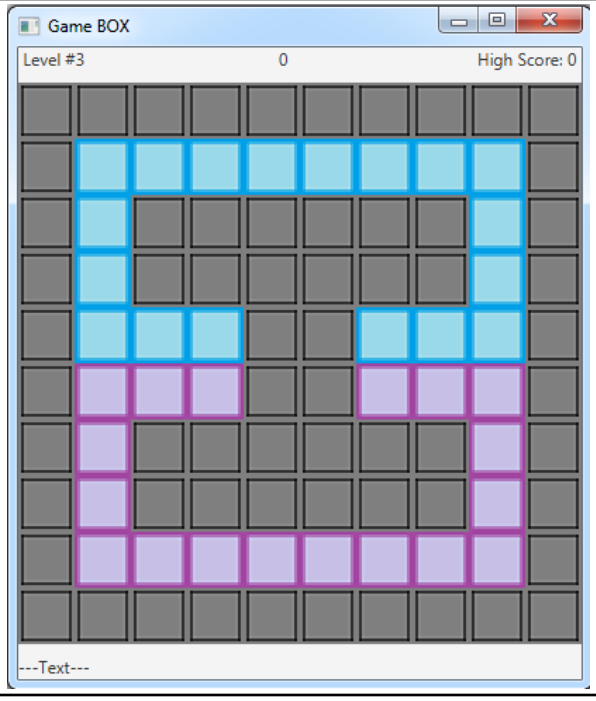
- Finally, the user clicked the box at position [8, 1] again and destroy all boxes.
- Thus, Level 1 is completed and the text “Next Level” is displayed.
- Now the user can continue next level with click the "Next Level" text.

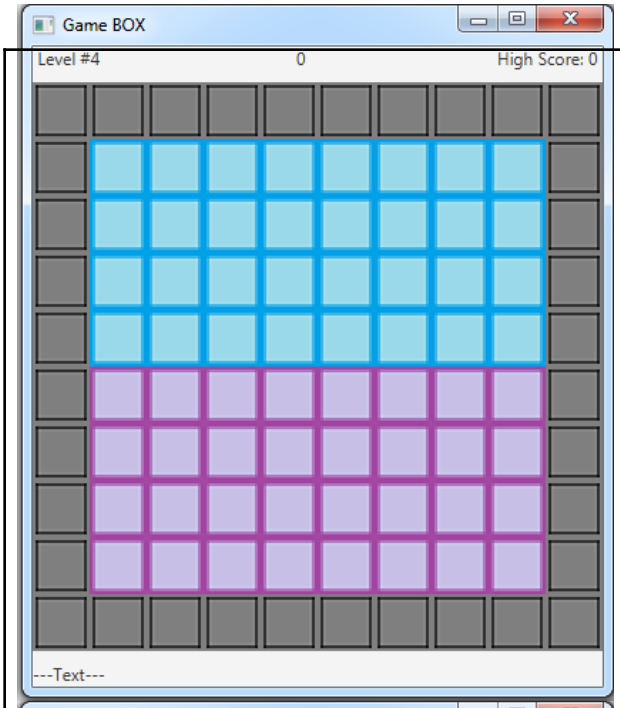
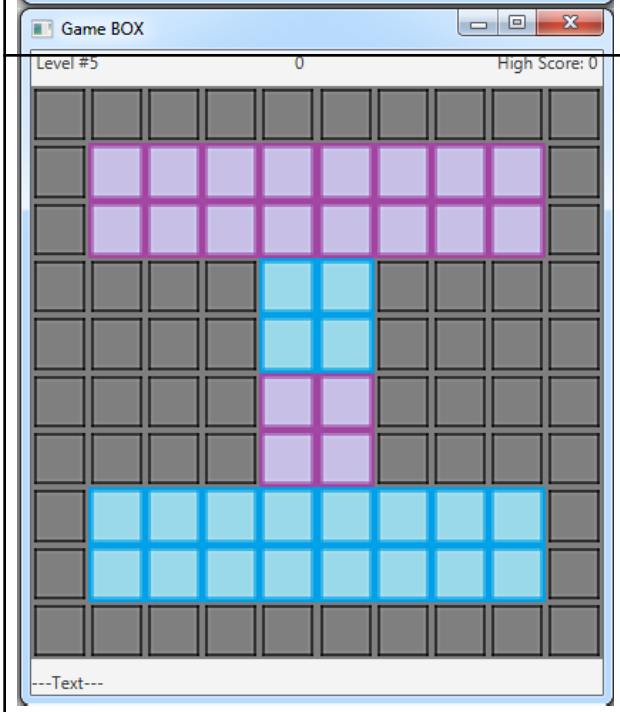


A sample game screen for Level 2

<table>
<tr>
<td> 
</td>
<td>

Boxes at [5, 4] and [5, 3] are Wood type originally and durability (life) value is 2. Boxes at [4, 4], [5, 5], and [6, 4] are Mirror type originally and durability (life) value is 1.
Suppose that the user clicked on the box at position [5, 4] and hit also four neighboring boxes (at [5, 3], [4, 4], [5, 5], and [6, 4]).
After the box click at [5, 4], durability value of boxes at position [5, 4], [5, 3], [4, 4], [5, 5] and [6, 4] decreased by 1. Thus, now the durability value for boxes at positions [5, 4] and [5, 3] is 1 and the box type changed to the Mirror type. And the durability value for other boxes at positions [4, 4], [5, 5] and [6, 4] is 0 and the box type changed to the Empty type (as it seen from the figure).
And with the click action, 5 blocks are affected (clicking the box at [5, 4] and hitting the boxes at [4, 4], [6, 4], [5, 3], and [5, 5]) and the user gets +4 points.

</td>
</tr>
<tr>
<td> 
</td>
<td>A sample game screen for Level 3</td>
</tr>
</table>

	A sample game screen for Level 4
	A sample game screen for Level 5