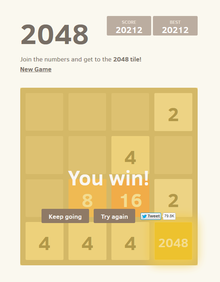
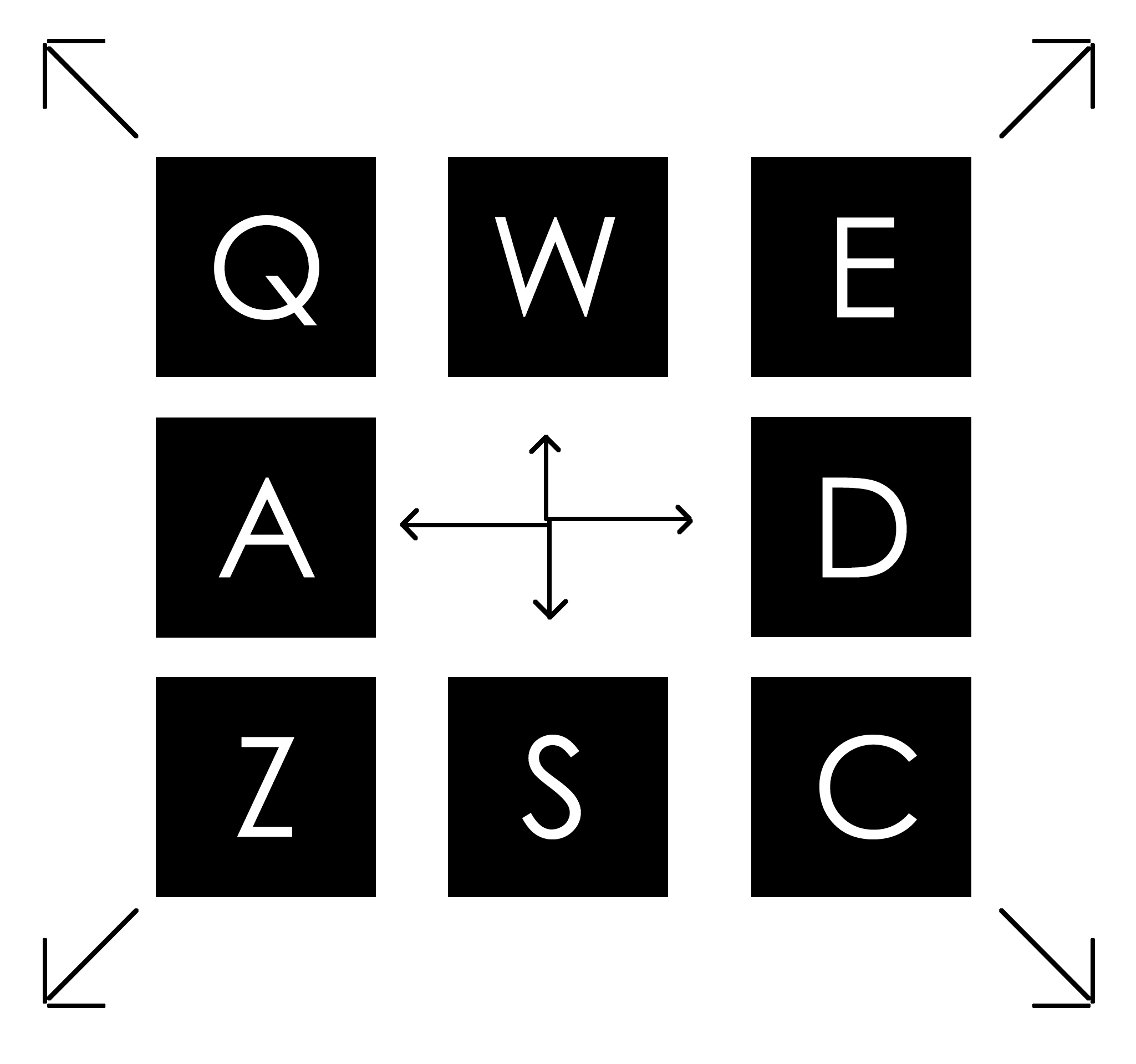
1. Program description, description of program/application

The program Diagonal 2048, is a program modified based on original 2048 game. The **game's** objective is to slide numbered tiles on a grid to combine them to create a tile with the number **2048**; however, you can keep playing the **game**, creating tiles with larger numbers. The keys that will be used to control the game is ASDW (for straight merging) and QEZC (for diagonal merging). The target audience for this game is everyone, the game was designed for spare time, which means it does not force the player to complete the game.



|  |  |  |  |
| --- | --- | --- | --- |
| C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeTopRight.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeTopLeft.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeBotRight.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeBotLeft.jpg |
| C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeUp.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeDown.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeLeft.jpg | C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\mergeRight.jpg |

1. Analysis and design (IPO, variable required)

|  |  |
| --- | --- |
| Input | -Key Event (A, S, D, W, Q, E, Z, C)  -Mouse Event, click on the button to change game state. |
| Process | -Merge the 2D array in different directions according to the key event  -Update score record |
| Output | The graphics output of grid updates accordingly to the change to a 2D array, and update score output. |
| Variables | -Grid: a 2D array for game grid  -screen size: control panel size  -Row: #rows of the 2D array  -Column: #column for 2D array  -State: control game state (home, instruction, change dimension, game state)  -buttonSize: standard button size in the game  -scrW, scrH: changed screenSize by user |

1. Concept planning to use, how u use, major methods (logic)

The concept that will be applied is firstly, a superclass that extends JPanel so that the graphics could update within that container. Secondly, I will implement a 2D array to make the grid of the 2048 game panel. Thirdly, the KeyListener and MouseListener to read the input from the user. Lastly, the nested loop to access and manipulate the data in the 2D array.

I will organize code mainly with classes, the class for reading key and mouse action will be in handler class. The paintComponent will also be in one class, and there will be a runner method to add everything into the main JFrame.

C:\Users\deyun\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Untitled-1.jpg

The logic of this game is that if the current index is not zero, then if it is equal to next Non-zero index, then add them together at current index. If the current index is zero, then see if the last index is equal to next Non-zero index, sum them if condition is true, if not, then simply replace the current index with the next Non-zero index.

1. What I’ve researched, graphics, general logic of the game

I’ve searched up for the use of mouse and key action listener to respond to button clicking, and game playing with key pressed.

I learned the graphics, such as paintComponent and fontMatrix in the textbook, because I found it easier to understand with their examples. These will be used to center the text, and update graphics.

I’ve seen a Youtube video hardcoding 2048. Based on that, I seek for the pattern behind that hard coding and made an algorithm that can work with any size of grid.

https://www.youtube.com/watch?v=71r1YZ\_\_IM8&list=PLYjvpPnQVE2awC2juEerxSdnLeRJnD1Go