Name : Luo Siwei　　Neptun Code : CDXMQ3　　Assignment Number : 6

Index:

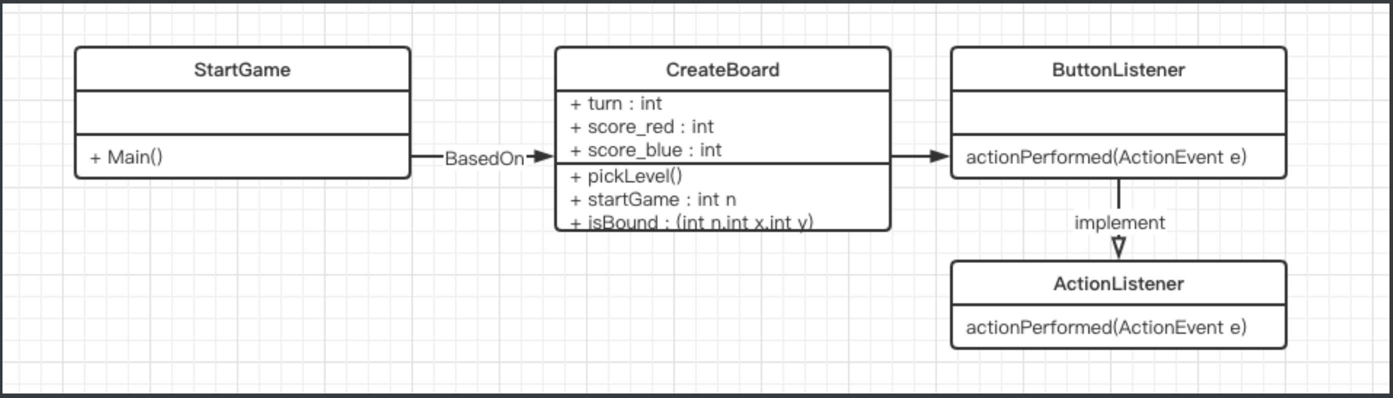
1.the description of the exercises

```
4. Tricky five-in-a-row
Create a game, which is a variant of the well-known five-in-a-row game.
The two players can play on a board consists of n x n fields. Players
put their signs alternately (X and O) on the board. A sign can be put
only onto a free field. The game ends, when the board is full, or a
player won by having five adjacent signs in a row, column or diagonal.
The program should show during the game who turns.

The trick in this variant is that if a player makes 3 adjacent signs
(in a row, column or diagonal), then one of his signs is removed
randomly (not necessary from this 3 signs). Similar happens, when the
player makes 4 adjacent signs, but in this case two of his signs are
removed.

Implement this game, and let the board size be selectable (6x6, 10x10,
14x14). The game should recognize if it is ended, and it has to show in
a message box which player won (if the game is not ended with draw),
and automatically begin a new game.
```

2.the class diagram



3.the short description of each methods

Main Method : The only purpose of the `main` method is to call the `picklevel` method in class createboard.

PickLevel Method : The `picklevel` method creates three different JButtons. After player clicks any Button, the component listener will get the mouse click event, then call the startGame method and pass in the parameter `n` . The parameter `n` is determined by the button.

StartGame Method : The `startgame` method is the main method of the game. It will create an `n*n` two-dimensional array of type JButton. Here, the responsibilities of each JButton are the same, because each JButton has the same listener. After clicking the button, if the value of field is less than 4, it will increase by 1.A method named `isBound` will be called to determine whether the field is within the boundary.

isBound : To determine whether the field is within the boundary.

4.the connections between the events and event handlers.

```
1.Click the button to start the Game
JButton jButton = new JButton("3x3");
jButton.addActionListener(e -> {
    startGame(3);
    frame.setVisible(false);
});
```

```
2.Click the button to increase the number
jButton[i][j].addActionListener(e -> {
    int[][] edges = {{0,1},{1,0},{0,-1},{-1,0},{1,1},{-1,-1},{-1,1},
{1,-1},{0,0}};
    for (int k = 0; k < 9; k++) {
        int newx = finalI + edges[k][0];
        int newy = finalJ + edges[k][1];
        if( isBound(n,newx,newy)){
        if(Integer.parseInt(jButton[newx][newy].getText())<4) {
            jButton[newx]
[newy].setText(Integer.toString((Integer.parseInt(jButton[newx]
[newy].getText()) + 1)));
        }
    }
}
```

5.Test Cases

```
test case 1.
click close the window
==> successful exit the program
```

```
test case 2.
we picked a 3x3 board,
in red turn, click (0,0),
in blue turn, click (0,0),
in red turn, click (0,0),
in blue turn,click (2,2),
in red turn,click (2,2),
in blue turn,click (2,2),
in red turn,click (2,2),
in blue turn,click (0,0)
in red turn,click (2,0)
in blue turn,click (2,0)
in red turn,click (2,0)
in blue turn,click (2,0)
in red turn,click (0,2)
in blue turn,click (0,2)
in red turn,click (0,2)
in blue turn,click (0,2)
==> Red win the game,score is 5(Correct!)
```

```
test case 3.
we picked a 3x3 board,
in red turn, click (1,1),
in blue turn, click (1,1),
in red turn, click (1,1),
in blue turn, click (1,1),
==> Blue win the game,score is 9(Correct!)
```

```
test case 4.
we picked a 5x5 board,
in red turn, click (1,1),
in blue turn, click (1,1),
in red turn, click (1,1),
in blue turn, click (1,1),
in red turn, click (3,3),
in blue turn, click (3,3),
in red turn, click (3,3),
in blue turn, click (3,3),
in red turn, click (4,1),
in blue turn, click (4,1),
in red turn, click (4,1),
in blue turn, click (4,1),
in red turn, click (1,4),
in blue turn, click (1,4),
in red turn, click (1,4),
in blue turn, click (1,4),
==> Blue win the game,score is 25(Correct!)
```

```
test case 5.
we picked a 7x7 board,
in red turn, click (3,1),
in blue turn, click (4,1),
in red turn, click (5,1),
in blue turn, click (6,1),
in red turn, click (1,3),
in blue turn, click (2,3),
in red turn, click (3,3),
in blue turn, click (4,3),
in red turn, click (3,1),
in blue turn, click (4,1),
in red turn, click (5,1),
in blue turn, click (6,1),
in red turn, click (1,3),
in blue turn, click (2,3),
in red turn, click (3,3),
in blue turn, click (4,3),
in red turn, click (3,1),
in blue turn, click (4,1),
in red turn, click (5,1),
in blue turn, click (6,1),
in red turn, click (1,3),
in blue turn, click (2,3),
in red turn, click (3,3),
in blue turn, click (4,3),
    ==> Red win the game,score is 31(Correct!)
```