Programming Technology

Saif Ahmad Sami

W3RQNK

Assignment 2

Question:

8. Knight tournament

This two-player game is played on a board consisting of n x n fields. Initially, two white and two black knights are placed at the corners of the board (knights of the same color are placed at the opposite corners). Players step alternately, and knights can move only in an L shape, like in chess. The board is initially grey, but after each step, the visited places are colored with the color of the visitor knight (the previous color of the field doesn’t matter). A player wins, if there are 4 adjacent fields (horizontally, vertically, or diagonally) which are colored to the player's color. The game ends, when there are no more grey fields.

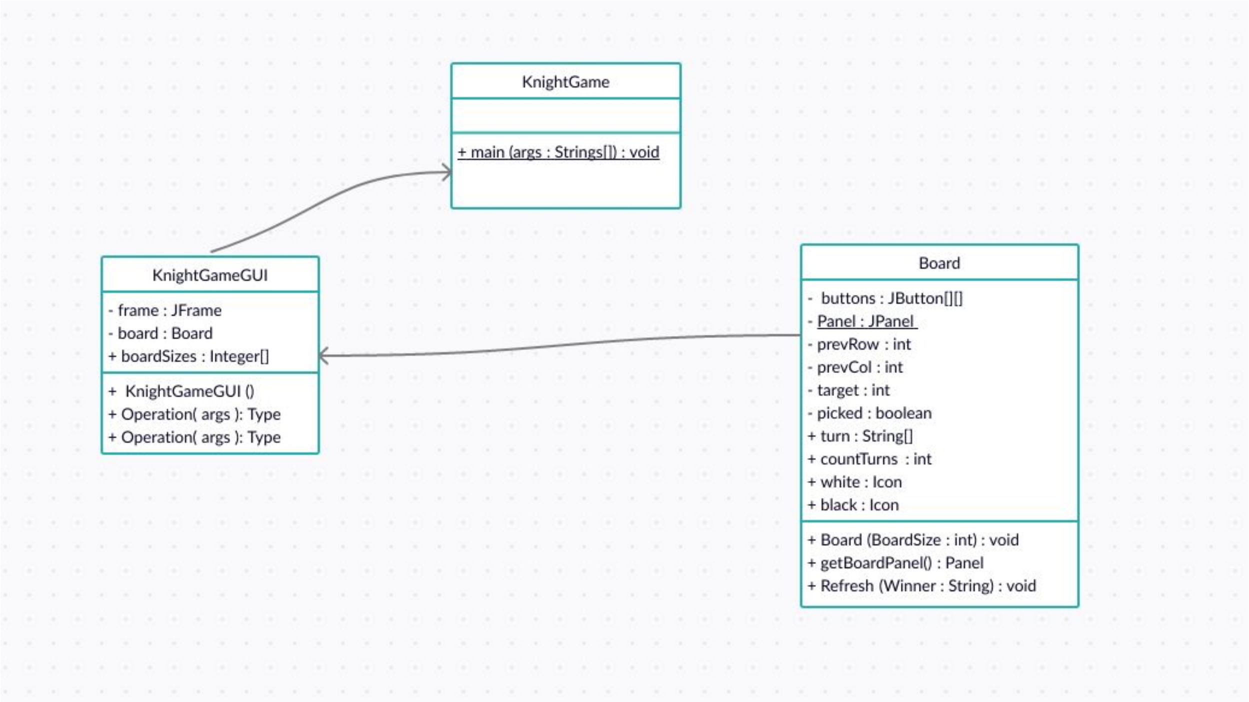
Implement this game, and let the board size be selectable (4x4, 6x6, 8x8). The game should recognize if it is ended, and it has to show in a message box which player won. After this, a new game should be started automatically.

Description:

We have to simulate a chess like board on which we will only have two Knights on each side. Their movement will be restricted by the rules of chess but they cannot attach each other meaning they are invincible. But they will leave their color on every block they have visited. So the first player who manages to have four consecutive same color fields, may it be horizontal, vertical or diagonal will win. The board can be of various sizes but it will always be a square.

For this, we will firstly create the structure of the game that is the board and all the menu options. After that we will implement the restrictions behind the movement of the Knight pieces and the check for their winning conditions.

Class Diagram:



Function Description:

knightGameGUI():

The function is the constructor of the class KnightGameGUI and will build up the board and the panel and the frame. It will also initiate an instance of the Board class.

getBoardPanel():

This function returns the Panel.

hints():

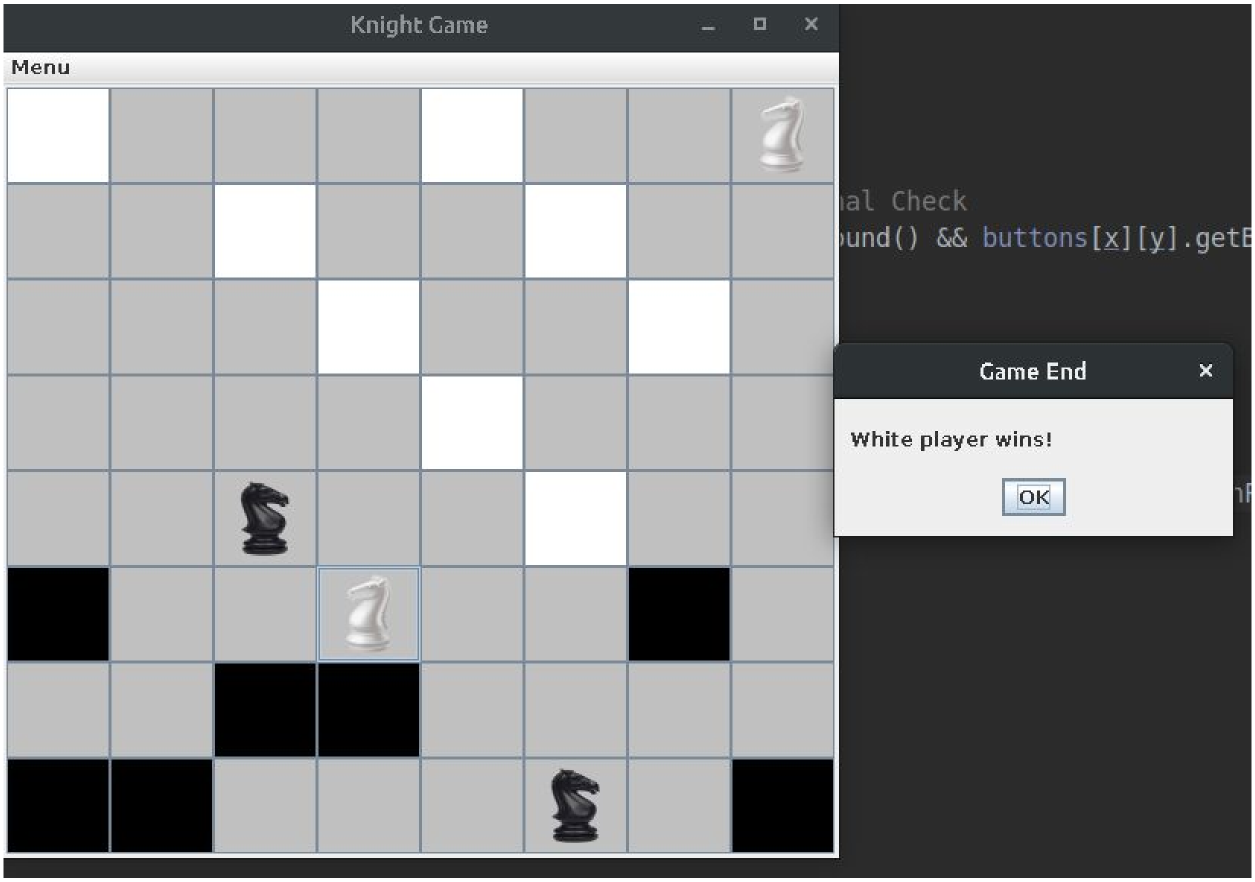
The function marks all the possible moves of the selected knight to make it easier for the player to move his piece.

actionPerformed(ActionEvent e):

The function is called whenever an action is performed on the Panel. It checks if the right player is picking the piece and if the players are taking turns. It also continuously checks if the player has achieved the target.

Testing:

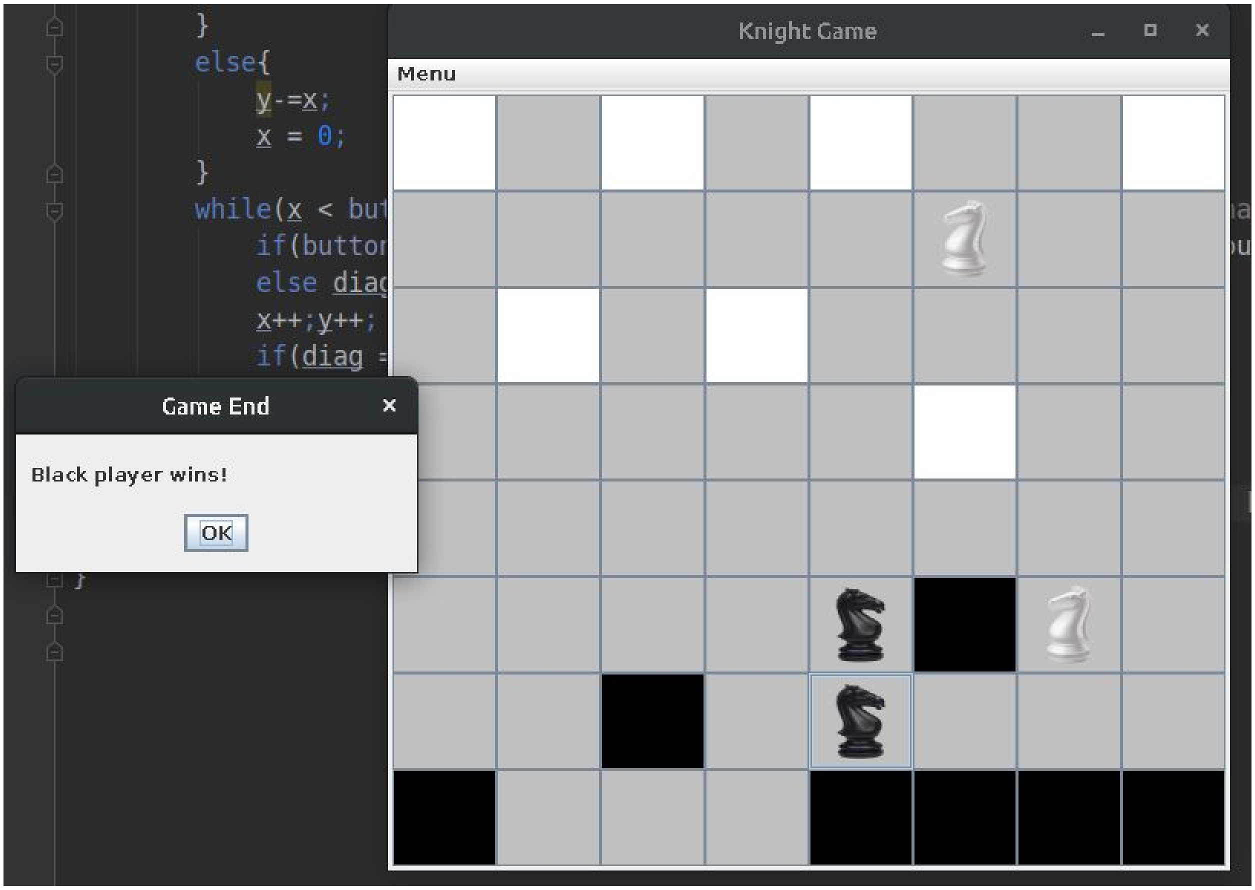
Diagonal Check:



White player wins!

```
al Check
und() && buttons[x][y].getE
```

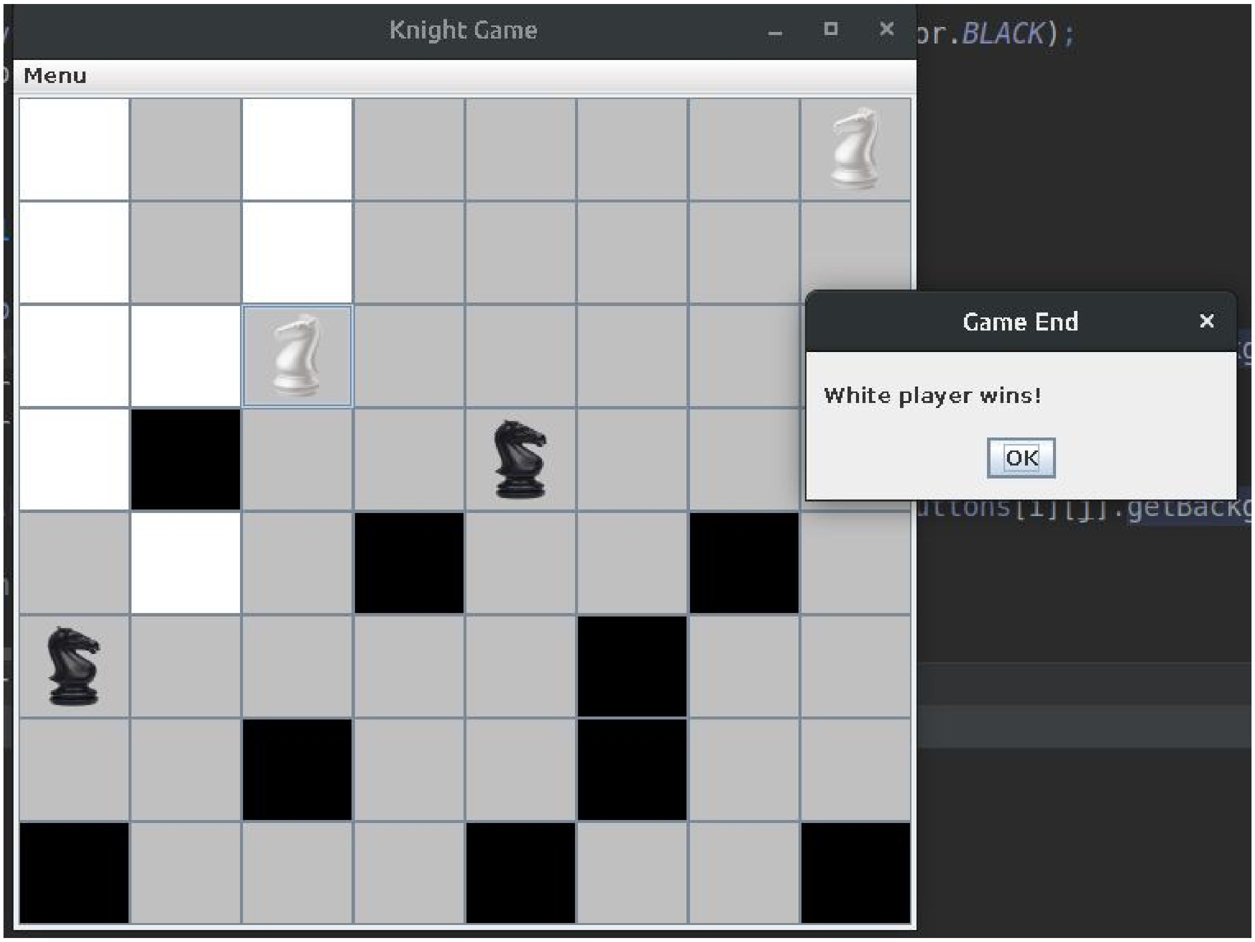
Horizontal Check:



```
        }
        else{
            y-=x;
            x = 0;
        }
        while(x < but
            if(button
            else diag
            x++;y++;
            if(diag =
```

Black player wins!

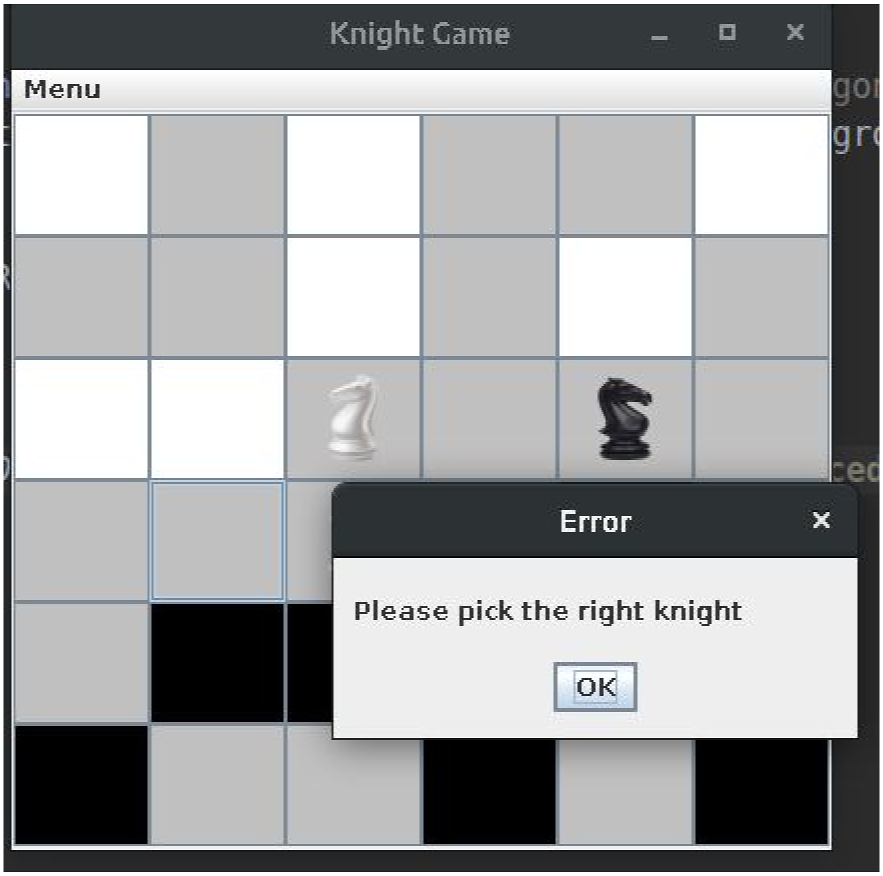
Vertical Check:



White player wins!

```
or.BLACK);
uttons[i][j].getBackg
```

Error Check:



Please pick the right knight