# Project 1

< Chess Game >

CSC-5-45113

Name: Shi Liu

Date: 07/17/2018

# Introduction

Title: Chess Game

# This Chess Game called Gobang Chess. Gobang is one of the traditional black and white chess originated in ancient China. Modern Chinese chess is called "Lian Zhu" in Japanese. English is translated into "Renju". English is called "Gobang" or "FIR" (the abbreviation of Five in a Row). There are many names of "Lian five Zi", "five Zi Lian", "string ball", "five eyes", "five eyes touch", and “five grid".    Gobang cannot only enhance thinking ability, improve intelligence, but also enrich philosophy and help cultivate self-cultivation. It has the obvious characteristics of modern leisure, "short, flat, fast", and the profound knowledge of classical philosophy, "Yin and Yang Yi Li". It has the characteristics of simple and easy learning, the people are pleased to see, and have profound skills and high level international competition; its chess culture is long and mysterious and west of the East. The intuition of the party is not only the concept of "field", but also the connection of "point". It is a communication point between Chinese and Western cultures, and a crystallization of ancient and modern philosophies.   This game is a java language as a development tool, using the constructor to draw the chess, as well as the falling function and judgment function, to realize the functions of double players.

# Design Idea

The first step: draw the chess score.  
The second step: the two sides of the war.  
The third step: to judge whether there are five beads.  
The fourth step: repeat first, second, third steps until there are five sons.

**Summery**

# Through the constructor to achieve the chessboard, write the drop function to achieve the player drop, and determine whether there is a result of five sub beads by judging the function.



1. Add menu

JMenuBar bar = new JMenuBar();

this.setJMenuBar(bar);

// 添加菜单栏目录

JMenu menu1 = new JMenu("游戏菜单"); // 实例化菜单栏目录

JMenu menu2 = new JMenu("设置");

JMenu menu3 = new JMenu("帮助");

bar.add(menu1); // 将目录添加到菜单栏

bar.add(menu2);

bar.add(menu3);

JMenu menu4 = new JMenu("博弈模式"); // 将“模式”菜单添加到“设置”里面

menu2.add(menu4);

// 设置“”目录下面的子目录

JRadioButtonMenuItem item1 = new JRadioButtonMenuItem("人人博弈");

JRadioButtonMenuItem item2 = new JRadioButtonMenuItem("人机博弈");

2．Add a mouse click event to each menu as follows:

For example, item 1 is the game setting menu defined above, and his mouse click event is added as follows, and other menus are similar.

item1.addMouseListener(new MouseListener() {

@Override

public void mouseReleased(MouseEvent e) {

// TODO Auto-generated method stub

}

@Override

public void mousePressed(MouseEvent e) {

// TODO Auto-generated method stub

Icon icon = new Icon() {

@Override

public void paintIcon(Component c, Graphics g, int x, int y) {

// TODO Auto-generated method stub

}

@Override

public int getIconWidth() {

// TODO Auto-generated method stub

return 0;

}

@Override

public int getIconHeight() {

// TODO Auto-generated method stub

return 0;

}

};

Object[] options = { "保存并重新开始游戏", "不，谢谢" };

int n = JOptionPane.showOptionDialog(null, "是否保存设置并重新开始", "人机博弈设置", 0, 1, icon, options, "保存并重新开始游戏");

if (n == 0) {

panel.setIsManAgainst(true);

panel.Start();

item1.setSelected(true);

}

}

@Override

public void mouseExited(MouseEvent e) {

// TODO Auto-generated method stub

}

@Override

public void mouseEntered(MouseEvent e) {

// TODO Auto-generated method stub

}

@Override

public void mouseClicked(MouseEvent e) {

// TODO Auto-generated method stub

}

});

3. Variable description

public Image boardImg; // 定义背景图片

static int[][] allChess = new int[15][15]; // 棋盘数组

static int[][] temporaryChess = new int[15][15];

int x;// 保存棋子的横坐标

int y;// 保存棋子的纵坐标

Boolean canPlay = false; // 游戏是否继续，默认为继续

Boolean isBlack = true;// 是否是黑子，默认为黑子

Boolean isManAgainst = false; // 判断是否是人人对战

int blackTime = 120;//倒计时

int whiteTime = 120;//倒计时

int gameDifficulty；//游戏模式

4. Judging win or lose

private boolean checkWin(int x, int y) {

// TODO Auto-generated method stub

boolean flag = false;

// 保存共有多少相同颜色棋子相连

int count = 1;

// 判断横向 特点：allChess[x][y]中y值相同

int color = allChess[x][y];

// 判断横向

count = this.checkCount(x, y, 1, 0, color);

if (count >= 5) {

flag = true;

} else {

// 判断纵向

count = this.checkCount(x, y, 0, 1, color);

if (count >= 5) {

flag = true;

} else {

// 判断右上左下

count = this.checkCount(x, y, 1, -1, color);

if (count >= 5) {

flag = true;

} else {

// 判断左下右上

count = this.checkCount(x, y, 1, 1, color);

if (count >= 5) {

flag = true;

}

}

}

}

return flag;

}

Is to judge the horizontal, vertical, oblique 4 in the direction of whether there are 5 consecutive pieces of the same color

5.Game mode

If it is a man-machine mode, the name will be randomly listed. If it is in the mode of everyone, the name will switch to the next game when the mouse is clicked.