**Java程序设计**

**课程设计报告**

**\*\*\*\***

|  |  |
| --- | --- |
| **专 业** |  |
| **班 级** |  |
| **学 号** | **\*\*\*\*\*\*\*\*\*\*** |
| **姓 名** |  |

**二 ○ 年 月**

**目录**

1. ···················································设计目的
2. ·············································设计思路与分析

··················································2.1时序图

····················································2.2类图

1. ···················································具体代码
2. ·····················································评分表

## 设计目的

为了体现所学知识，计划使用java做一个类似贪吃蛇的小游戏

## 二、设计思路与分析

使用java的GUI的图形界面，运用动作响应函数，按钮，面板，等组件。

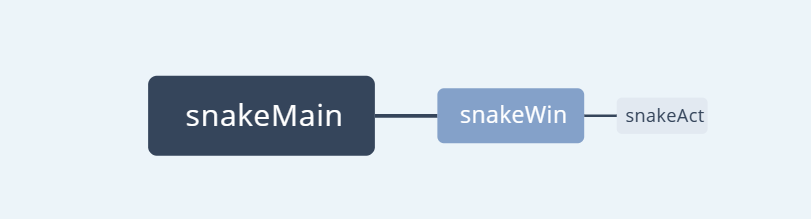
主要使用swing，awt中的event，Jpanel，ActionListener，KeyListener，Runnable库

运行环境：

jdk1.8

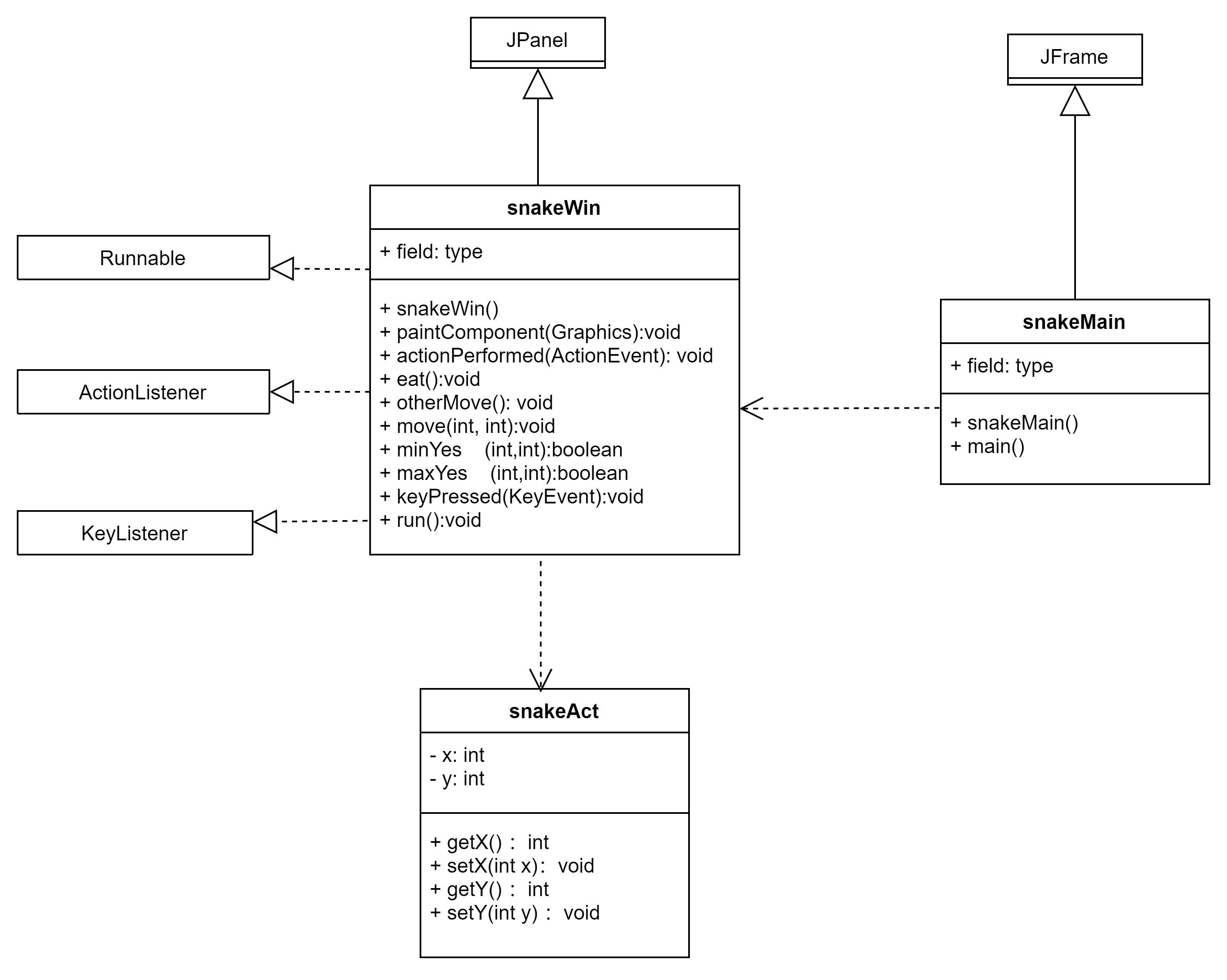
IDEA

## .1时序图



## 2.2 类图

三、系统设计与实现

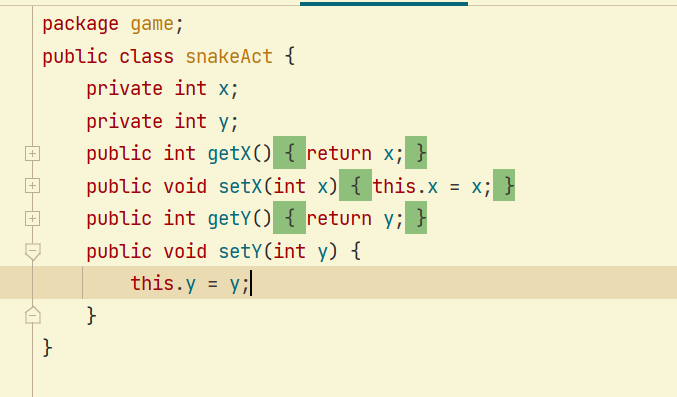


## 3.1部分功能截图

1、创建窗体，将游戏加入窗体



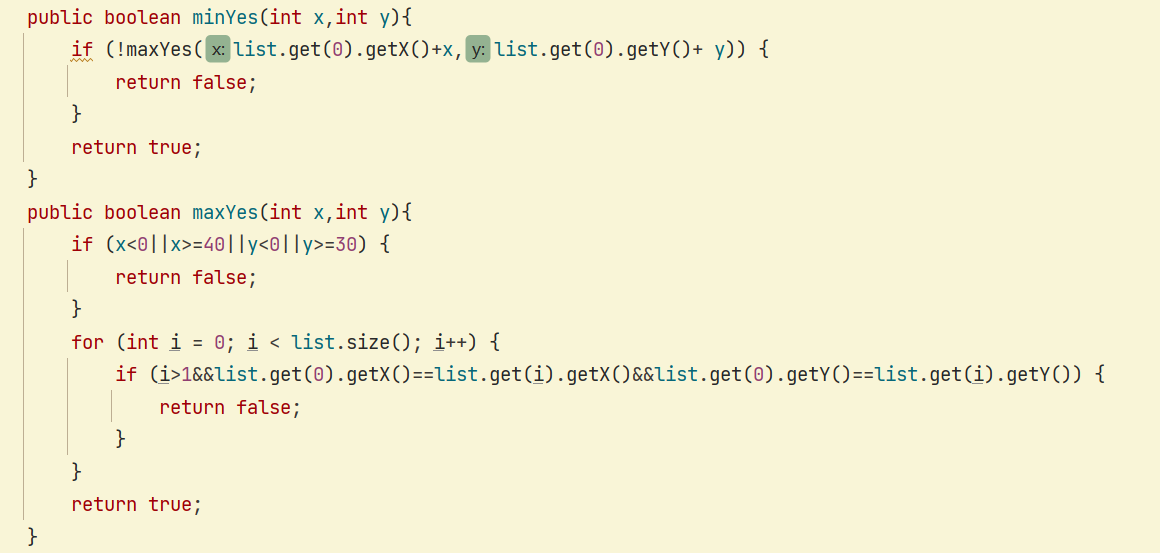
2、定义小蛇的位置



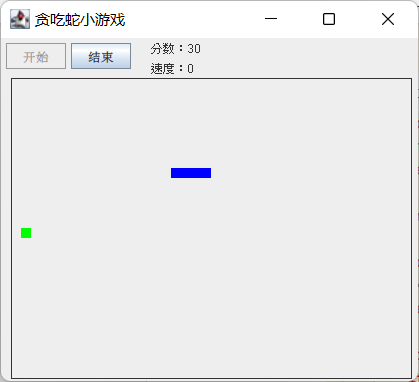
3、小蛇吃东西，长按加速



4、判断是否撞墙



5、运行效果截图



## 3.2核心代码

snakeMain,java

package game;

import java.awt.Graphics;

import javax.swing.\*;

public class snakeMain extends JFrame {

public snakeMain() {

snakeWin win = new snakeWin();

add(win);

setTitle("̰贪吃蛇v1.0------By FireRagin");

setSize(435,390);

setLocation(200, 200);

setVisible(true);

}

public static void main(String[] args) {

new snakeMain();

}

}

snakeAct.java

public class snakeAct {

private int x;

private int y;

public int getX() {

return x;

}

public void setX(int x) {

this.x = x;

}

public int getY() {

return y;

}

public void setY(int y) {

this.y = y;

}

}

snakeWin.java

package game;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.util.\*;

import java.util.List;

public class snakeWin extends JPanel implements ActionListener,KeyListener,Runnable{

int fenShu=0,Speed=0;

boolean start = false;

int rx=0,ry=0;

int eat1=0,eat2=0;

JDialog dialog = new JDialog();

JLabel label = new JLabel("你挂了！你的分数是"+fenShu+"。");

JButton ok = new JButton("T\_T");

Random r = new Random();

JButton newGame,stopGame;

List<snakeAct> list = new ArrayList<snakeAct>();

int temp=0;

Thread nThread;

public snakeWin() {

newGame = new JButton("开始");

stopGame = new JButton("结束");

newGame.addActionListener(this);

stopGame.addActionListener(this);

this.addKeyListener(this);

this.setLayout(new FlowLayout(FlowLayout.LEFT));

this.add(newGame);

this.add(stopGame);

dialog.setLayout(new GridLayout(2, 1));

dialog.add(label);

dialog.add(ok);

dialog.setSize(200, 200);

dialog.setLocation(200, 200);

dialog.setVisible(false);

ok.addActionListener(this);

}

public void paintComponent(Graphics g){

super.paintComponent(g);

g.drawRect(10, 40, 400, 300);

g.drawString("分数："+fenShu, 150, 15);

g.drawString("速度："+Speed, 150, 35);

g.setColor(new Color(0, 255, 0));

if(start){

g.fillRect(10+rx\*10, 40+ry\*10, 10, 10);

for (int i = 0; i < list.size(); i++) {

g.setColor(new Color(0, 0, 255));

g.fillRect(10+list.get(i).getX()\*10, 40+list.get(i).getY()\*10, 10, 10);

}

}

}

public void actionPerformed(ActionEvent e) {

if(e.getSource()==newGame){

newGame.setEnabled(false);

start = true;

rx=r.nextInt(40);ry=r.nextInt(30);

snakeAct tempAct = new snakeAct();

tempAct.setX(20);

tempAct.setY(15);

list.add(tempAct);

this.requestFocus();

nThread = new Thread(this);

nThread.start();

repaint();

}

if(e.getSource()==stopGame){

System.exit(0);

}

if(e.getSource()==ok){

list.clear();

start=false;

newGame.setEnabled(true);

dialog.setVisible(false);

fenShu=0;

Speed=0;

repaint();

}

}

private void eat() {

if (rx==list.get(0).getX()&&ry==list.get(0).getY()) {

rx = r.nextInt(40);ry = r.nextInt(30);

snakeAct tempAct = new snakeAct();

tempAct.setX(list.get(list.size()-1).getX());

tempAct.setY(list.get(list.size()-1).getY());

list.add(tempAct);

fenShu = fenShu+100\*Speed+10;

eat1++;

if(eat1-eat2>=4){

eat2=eat1;

Speed++;

}

}

}

public void otherMove(){

snakeAct tempAct = new snakeAct();

for (int i = 0; i < list.size(); i++) {

if (i==1) {

list.get(i).setX(list.get(0).getX());

list.get(i).setY(list.get(0).getY());

}else if(i>1){

tempAct=list.get(i-1);

list.set(i-1, list.get(i));

list.set(i, tempAct);

}

}

}

public void move(int x,int y){

if (minYes(x, y)) {

otherMove();

list.get(0).setX(list.get(0).getX()+x);

list.get(0).setY(list.get(0).getY()+y);

eat();

repaint();

}else {

nThread = null;

label.setText("你挂了！你的分数是"+fenShu+"。");

dialog.setVisible(true);

}

}

public boolean minYes(int x,int y){

if (!maxYes(list.get(0).getX()+x,list.get(0).getY()+ y)) {

return false;

}

return true;

}

public boolean maxYes(int x,int y){

if (x<0||x>=40||y<0||y>=30) {

return false;

}

for (int i = 0; i < list.size(); i++) {

if (i>1&&list.get(0).getX()==list.get(i).getX()&&list.get(0).getY()==list.get(i).getY()) {

return false;

}

}

return true;

}

public void keyPressed(KeyEvent e) {

if(start){

switch (e.getKeyCode()) {

case KeyEvent.VK\_UP:

move(0, -1);

temp=1;

break;

case KeyEvent.VK\_DOWN:

move(0, 1);

temp=2;

break;

case KeyEvent.VK\_LEFT:

move(-1, 0);

temp=3;

break;

case KeyEvent.VK\_RIGHT:

move(1, 0);

temp=4;

break;

default:

break;

}

}

}

public void keyReleased(KeyEvent e) {}

public void keyTyped(KeyEvent e) {}

public void run() {

while (start) {

switch (temp) {

case 1:

move(0, -1);

break;

case 2:

move(0, 1);

break;

case 3:

move(-1, 0);

break;

case 4:

move(1, 0);

break;

default:

break;

}

repaint();

try {

Thread.sleep(300-30\*Speed);

} catch (InterruptedException e) {

// TODO 自动生成的 catch 块

e.printStackTrace();

}

}

}

}

**《java程序设计》课程设计评分表**

**学 号：**

**姓 名：**

|  |  |  |  |
| --- | --- | --- | --- |
| **项目** | **内容** | **所占分值** | **成绩** |
| **功能模块** | **功能实现** | **20分** |  |
| **可扩展性** | **10分** |  |
| **程序界面** | **10分** |  |
| **代码部分** | **面向对象** | **10分** |  |
| **算法** | **10分** |  |
| **答辩部分** | **问题** | **30分** |  |
| **报告** |  | **10分** |  |
| **总分** |  | | |