|  |
| --- |
| 28组 |
| 俄罗斯方块 |
| 林永泽，马钰杰，潘晨峰 |

|  |
| --- |
| Josephstalin  2013/11/4 |

## 1.课程设计的目的

Java语言是当今流行的网络编程语言，它具有面向对象、跨平台、分布应用等特点。俄罗斯方块游戏是一个经典的小游戏，由于它简单有趣，因而得到了广泛的流行，男女老幼都适合。而俄罗斯方块游戏的设计工作复杂且富有挑战性，它包含的内容多，涉及的知识广泛，与图形界面联系较大，包括界面的显示与更新、数据收集等，在设计的过程中，必将运用到各方面的知识，这对于设计者而言，是个很好的锻炼机会。

因此为了进一步巩固课堂上所学到的知识，深刻掌握Java语言的重要概念及其面向对象的特性，锻炼熟练的应用面向对象的思想和设计方法解决实际问题的能力，通过此次课程设计，巩固所学Java语言基本知识，增进Java语言编辑基本功，拓宽常用类库的应用。使学生通过该教学环节与手段，把所学课程及相关知识加以融会贯通，全面掌握Java语言的编程思想及面向对象程序设计的方法，为今后从事实际工作打下坚实的基础。

## 2．俄罗斯方块的基本规则

1． 一个用于摆放小型正方形的平面虚拟场地，其标准大小：

行宽为10，列高为20，以每个小正方形为单位。

Cell类为小正方形，其中包括row、col、image等成员变量

getRow()、getCol()、getImage()分别获取其所在行、列和图片

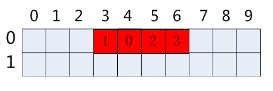
setRow()、setCol()、setImage()分别设置其值

2． 一组由4个小型正方形组成的规则图形，

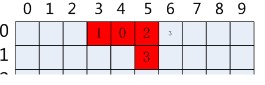
英文称为Tetromino，中文通称为方块共有7种，

分别以S、Z、L、J、I、O、T这7个字母的形状来命名。

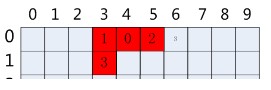
I：一次最多消除四层（橘色）



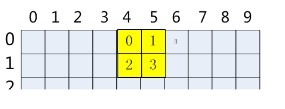
J（左右）：最多消除三层，或消除二层（紫色）



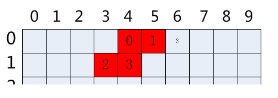
L：最多消除三层，或消除二层（蓝色）



O：消除一至二层（红色）



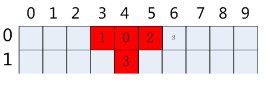
S（左右）：最多二层，容易造成孔洞（绿色）



Z （左右）：最多二层，容易造成孔洞（浅蓝色）

![C:\Users\pingguo\AppData\Roaming\Tencent\Users\1091209676\QQ\WinTemp\RichOle\UDX38N](DB{6%9BO748EJWC.jpg](data:image/jpeg;base64,)

T：最多二层（黄色）



3. Tetris类

(1)玩家键盘操作：

Q：退出

S：重新开始

P：终止运行

C：继续运行

Z：向左旋转

SPACE：直接下落

UP：向右旋转

DOWN：加速下落

LEFT：整体左移

RIGHT：整体右移

（2）方块移到区域最下方或是着地到其他方块上无法移动时，

就会固定在该处，而新的方块出现在区域上方开始落下。

（3）当区域中某一列横向格子全部由方块填满，则该列会消失

并成为玩家的得分。同时删除的列数越多，得分指数上升。

（4）当固定的方块堆到区域最上方而无法消除层数时，则游戏结束。

（6）一般来说，游戏还会提示下一个要落下的方块，熟练的玩家会计算到下一个方块，评估要如何进行。由于游戏能不断进行下去对商业用游戏不太理想，所以一般还会随着游戏的进行而加速提高难度。

（7）预先设置的随机发生器不断地输出单个方块到场地顶部

俄罗斯方块游戏

游戏显示区

暂停游戏

降低级数

提高级数

开始游戏

显示操作结果

显示玩家相应操做

功能键区

退出游戏

## 3. 类图



Cell 类

|-- int row 行号

|-- int col 列号

Tetromino 类

|-- Cell[] cells 4个格子

T 型方块 继承 Tetromino 类

I 型方块 继承 Tetromino 类

...

tetris (俄罗斯方块)

|-- Tetromino tetromino 一个正在下落的方块

| |-- cells 4个格子

|-- Tetromino nextOne 下一个 准备下落的方块

| |-- cells 4个格子

|-- Cell[][] wall 墙, 是方块下落到底部 打散着陆到墙上

|-- rows 20行

|-- cols 10列 个格子

三、功能分析

１、绘制功能：

paint() ：

1. 绘制背景图片
2. 平移绘图坐标系
3. 绘制正在下落的方块
4. 画墙
5. 绘制下一个要下落的方块
6. 绘制分数

paintWall() ：绘制墙

paintScore() ：绘制下落行数和分数

paintNextOne() ：绘制下一个下落的

paintCurrentOne(): 绘制正在下落的

２、开始功能： 开始自动下落 startAction()

1) 游戏开始时候执行 action() 方法中执行这个方法！

2) 开始流程

1.清空墙

2.生成下落的方块

3.清空分数

4.清理暂停 和 结束状态

5.启动定时器 定时执行 softDropAction

３、动作事件：ａｃｔｉｏｎ（）

引发键盘事件；Q为退出，S为重新开始 ，P为终止运行，C为继续运行，Z为向左旋转，SPACE为直接下落，UP为向右旋转，DOWN为加速下落，LEFT为整体左移，ＲＩＧＨＴ为整体右移

４、下落功能：（softDropAction()）

如果（检查）能够下落就下落，否则就着陆到墙上，如果有满行则消除，并检查游戏是否结束。新的方块出现在区域上方开始落下。

（检查）能够下落: tetrominoCanDrop()

着陆到墙上: tetrominoLandToWall()

破坏满的行：destroyLines();

检查游戏是否结束：checkGameOver();

５、直接下落功能：ｈａｒｄＤｒｏｐＡｃｔｉｏｎ（）

当按ＳＰＡＣＥ键时，方块直接下落

６、消除行：destroyLines()

销毁已经满的行，并且计分（１）迭代每一行（２）如果检查某行满是格子就销毁这一行

７、结束游戏功能 checkGameOver()

1) 流程

1. 检查wall[0][4] 是否有方块，如果有，游戏游戏结束

2. 如果游戏结束 gameOver = true

3. 停止timer

2) 事件处理

如果游戏结束，只能处理 [S] 重新开始游戏

3）在清理满行以后检查游戏结束！

修改 softDropAction hardDropAction

８、暂停、继续功能 pauseAction() continueAction()

1) 暂停流程

1. 停止定时器

2. 暂停状态设置为 true

2）暂停事件绑定到 [P] 按键。一定要拦截其它按键的处理

暂停期间只能处理 [C] 按键事件

3）继续流程

1. 重新创建并且启动定时器！

2. 暂停状态设置为 false

９、移动(左右)功能 moveRightAction（）、moveLeftAction（）

可以移动方块，移动边界就不能再移动了，如果碰到墙上的方块

也不能再移动了

变通解决方案：先移动方块，移动以后立即检查移动结果，如果

发现出界，就立即移动回来。也就是原地不动。

如果移动以后，检查发现与墙上的方块重合，也移动回来。

功能分析：

1）检查是否出界 outOfBound()

2）检查是否重合 coincide()

tetromino.moveLeft();

if(outOfBound() || coincide()){

tetromino.moveRight();

}

１０、 旋转功能

rotateRightAction()　、　rotateLeftAction()

tetromino.rotateRight();

if(outOfBound() || coincide()){

tetromino.rotateLeft();

}

## 3. 新增加的功能

1. 增加了等级制度

随着得分的增加，系统会加快下落的速度，同时面板上的level会增加，物理上会调用setLevel方法，其level属性封装在Cell类之中。

2. 增加了菜单

利用相应的方法，构建出帮助菜单和游戏菜单。

3. 增加了关于我们和游戏帮助弹出框

4. 增加了音乐功能

5. 增加了新游戏功能

6. 增加了游戏退出提示功能

7. 增加了游戏一键换肤功能

8. 增加了游戏成绩保存功能

9．增加了游戏成绩天梯排行榜功能

10.增加游戏等级提示功能

11.增加了游戏结束提示功能