

# 基于Java的连连看游戏的设计与实现

陈珊<sup>1,2</sup>, 张引琼<sup>1</sup>

(1.湖南农业大学 信息科学与技术学院, 湖南 长沙 410128 ; 2. 中南大学 信息科学与工程学院, 湖南 长沙 410128 )

**摘要:**采用Eclipse开发平台,用Photoshop技术和Java语言实现了具有刷新、提示、炸弹、演示、计时、计分、设置、刷新排行榜等功能的连连看游戏系统的开发。该游戏为单机游戏,无需下载安装,系统界面友好、操作简单,是一款适合大众的小游戏。

**关键词:**连连看;Java;主界面;地图;对话框

**中图分类号:**TP311 **文献标识码:**A **文章编号:**1009-3044(2011)35-9107-04

## Design and Implementation of LianLianKan Game Based on Java

CHEN Shan<sup>1,2</sup>, ZHANG Yin-qiong<sup>1</sup>

(1.College of Information Science and Technology, Hunan Agricultural University, Changsha 2.College of Information Science and Engineering, Central South University, Changsha 410128)

**Abstract:** The design used Eclipse development platform, PhotoShop technology and Java language to implement the LianLianKan game. Functions include refresh, hint, bomb, demo, timing, scoring, setting, refresh list of best-gamer and so on. This is a single-player game with friendly interface and easily operation, it's a very suitable game for population.

**Key words:** LianLianKan; Java; main interface; map; dialog

随着社会的飞速发展,人们的压力也在逐渐增大,玩游戏成为了人们放松的首选方式,为了满足人们的需求,游戏事业也在火爆发展<sup>[1]</sup>,开发的游戏种类也越来越多,越来越新颖。而“连连看”游戏是一个经典的游戏,“连连看”顾名思义就是找出相关联的东西,这种“连连看”在网上基本是用在小游戏中,就是找出两张相同的图片,如果可以用不超过三根的折线相连,将可以消掉这一对图片<sup>[2]</sup>。本文将阐述如何用Java语言设计和实现一个连连看游戏。

## 1 系统架构

本系统主要分成三个部分,如图1所示。

1)游戏系统功能的实现部分:主要实现搭建程序的主框架、显示总体图形用户界面、实施事件监听、控制连连看背景音乐和音效、处理游戏的时间与分数、更新排行榜内容并显示排行榜等功能。

2)游戏系统的地图部分:主要包括构建连连看方块对象、根据游戏难度随机布局方块、判断路径的算法、显示容器中方块的图形界面、消除方块时动画的连线显示以及实现游戏各项功能对应的算法等等。

3)游戏系统的上层对话框部分:主要是实现三个对话框的图形用户界面和实施事件监听,并可以通过设置对话框对游戏属性进行设置、通过帮助对话框、关于对话框让用户了解游戏的各项规则、计分规则。

## 2 游戏的设计与实现

### 2.1 游戏的设计

本系统设计采用单机模式,系统总流程图如图2所示。考虑到本游戏是单机小游戏,充分考虑其娱乐性及受众人群,并没有很复杂的操作和功能。

### 2.2 关键技术的实现

1)游戏初始化:初始化系统所需要的资源(系统中所用到的各种图片以及连连看方块);初始化用户界面;获取系统的计分板、系统的计时板、界面上的排名按钮、消除方块的动画的速度;设置运行时主界面窗口显示的位置、主界面窗口的大小、标题、程序图标、动画光标;设

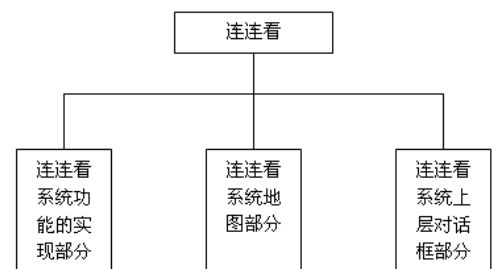


图1 系统结构图

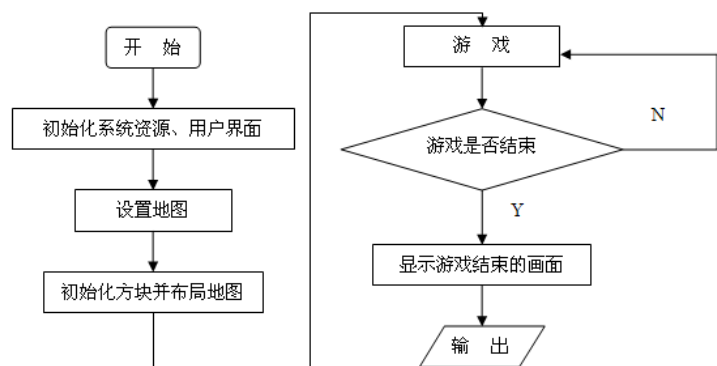


图2 系统流程图

收稿日期:2011-09-15

基金项目:湖南农业大学教改项目(B2010071)

本栏目责任编辑:谢媛媛

..... 软件设计开发 .....

9107

置背景音乐、音效。关键代码如下:

```
public LLK() {
    initResource(); //初始化系统所需要的资源
    map = new Map(); //地图
    ui = new MapUI(map, dots); //用户界面
    initUI(); //初始化用户界面
    ui.setScore(scoring); //实现计分
    ui.setClock(clocking); //实现计时
    ui.setTop10Button(goTop10);
    eliminating.setSpeed(setting.Deletespeed); //消除速度
    this.setSize(faceSize);
    Toolkit toolkit = Toolkit.getDefaultToolkit(); //得到默认的Toolkit对象
    Dimension screenSize = toolkit.getScreenSize(); //得到默认对象的屏幕大小
    this.setLocation((int) (screenSize.width - faceSize.getWidth()) / 2,
        (int) (screenSize.height - faceSize.getHeight()) / 2); //居中显示
    this.setTitle("连连看"); //设置标题
    this.setIconImage(icon); //设置程序图标
    URLClassLoader urlLoader = (URLClassLoader) this.getClass().getClassLoader(); //返回该类的类加载器
    URL url = urlLoader.findResource("images/cursor.gif"); //加载光标图标
    Image animateImage = new ImageIcon(url).getImage(); //获取光标图像
    Toolkit toolKit = Toolkit.getDefaultToolkit(); //得到默认的Toolkit对象
    Cursor cursor = toolKit.createCustomCursor(animateImage, new Point(0, 0), "cursor");
    this.setCursor(cursor);
}
```

其中 initResource()、initUI()是实现系统资源以及用户界面的初始化,调用此两个函数之后将会出现系统最初的界面,如图3所示。

2) 背景音乐:背景音乐的实现是本系统的一大特点,为该系统添加了更多的生气。在该系统的设置界面,可以选择播放音乐或不播放音乐,同时还可以选择播放的曲目。关键代码如下:

```
public void play() { //播放音乐
    if(isPlaying){return;}
    try {
        sequencer = MidiSystem.getSequencer(); //创建一个序列器实例
        sequencer.setSequence(sequence); //告诉序列器有一个可用的MIDI
        sequencer.open();
        sequencer.addMetaEventListener(this); //添加一个元事件监听器
        catch (InvalidMidiDataException ex) {}
        catch (MidiUnavailableException e) {}
        thread = new Thread(this); //创建一个新的线程
        thread.start(); //启动线程
    }
}
```

3) 计时、计分器:系统实现了对游戏进行的同时计时与计分的功能,游戏总的时间是方块数 $\times 4$ ,从游戏开始起开始计时,直至用户将界面上的方块全部消除或者时间用完为止,游戏结束;计分也是从用户开始游戏时开始计分,每消除一对方块加10分,当用户点击的两个方块不能消除时,扣1分,同时在游戏中使用了相应的道具扣相应的分数,最后若用户在规定的时间内提前将界面上的方块消除了,总分将会加上剩余的时间 $\times 2$ 作为最后的分数。计时器的关键代码如下:

```
public void paint(Graphics g) { //绘制显示的时间
    Graphics2D g2 = (Graphics2D) g;
    Dimension d = getSize();
    g2.setBackground(LLK.PinkColor);
    g2.clearRect(0, 0, d.width, d.height); //删除画布的矩形区域
    g2.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
        RenderingHints.VALUE_ANTIALIAS_ON); //使用抗锯齿模式完成呈现
    g2.setColor(color);
    g2.setFont(font48);
    g2.drawString("时间:" + getTime(), 16, 40); //输出时间
}
```

这段代码即是获取当前的所用的时间并在计时板上以抗锯齿模式显示出来。

4) 生成地图:系统需要根据用户所选择的游戏的难度随机布局一副地图,关键代码如下所示:

```
public Map(int level){
    this.level = 28;
```

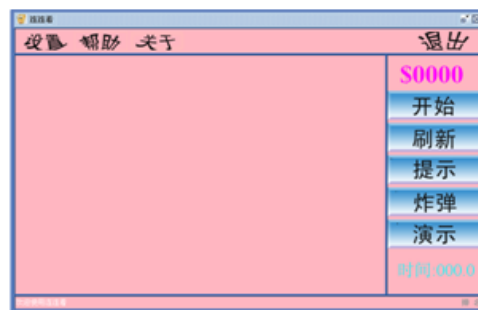


图3 系统用户界面

```

map = new int[10][17];
array = new int[170];
restBlock = this.level * 4;
this.level = level;
restBlock = level * 4;
initMap();
private void initMap(){
    for(int i = 0; i < level; i++){ //方块个数为 level*4
        array[i * 4] = i + 1;
        array[i * 4 + 1] = i + 1;
        array[i * 4 + 2] = i + 1;
        array[i * 4 + 3] = i + 1;
    }
    random(array);
    for(int i = 0; i < 10; i++){
        for(int j = 0; j < 17; j++){
            map[i][j] = array[i * 17 + j];
        }
    }
}

```

其中, initMap 是根据游戏难度生成连连看方块的图片, 并且每张图片出现四次, 然后在调用随机函数 random 布局好连连看的图片; 而 Map 函数是实现获取该次游戏中应该生成多少个连连看方块, 并调用 initMap 函数初始化地图。

5) 消除图片: 在本游戏中的规则是, 用户点击的两张相同的图片能用不超过三根直线相连时, 则可以消除这两张图片。因此在消除方块时, 有三种情况, 第一, 直接可以通过一根直线相连的方块; 第二, 两个方块的连线有一个拐角; 第三, 相连的直线有两个拐角。当判断两个方块能消除时, 将这两个方块所对应的位置置零, 及可将方块的图片消除。两个拐角情况的关键代码如下所示:

```

private boolean twoCorner(Point a, Point b){ //两个拐点
    vector = scan(a, b);
    if(vector.isEmpty())
        return false;
    for(int index = 0; index < vector.size(); index++){
        Line line = (Line)vector.elementAt(index); //在存储变量中找到一个元素
        if(line.direct == 1){ //找到的直线为横线
            if(vertical(a, line.a, false) && vertical(b, line.b, false)){
                if(!test)
                    animate = new Eliminating(0, a, line.a, line.b, b);
                return true; //先直线再横线再直线
            }
        } else if(horizon(a, line.a, false) && horizon(b, line.b, false)){
            if(!test)
                animate = new Eliminating(1, a, line.a, line.b, b);
            return true; //先横线再竖线再横线
        }
    }
    return false;
}

```

6) 刷新、显示排行榜: 在玩家玩完游戏之后, 状态栏中的排名按钮将会显示成可以使用的状态, 用户若想了解自己的游戏水平, 则可以查看排行榜, 通过点击状态栏中的排名按钮, 然后将会出现一个界面, 提示玩家输入姓名, 当玩家输入的为有效信息并点击确定之后将会弹出一个英雄榜的对话框, 此对话框中包含了游戏前十名的玩家的信息, 若该玩家所得的分数超过了上次排行榜上的任何一个分数, 排行榜将会被刷新, 此时英雄榜界面上显示的为最新的排名信息; 若玩家的得分没有超过之前排行榜上最后一名的分数, 则玩家不能在排行榜上看到自己的相关信息。关键代码如下所示:

```

public void Update(String Player_Name, String Score) throws IOException,
FileNotFoundException {
    System.out.println("Player_Name="+Player_Name+"Score="+Score);
    if (file.exists()) {
        System.out.println("File=====" + file.getName());
        List<String[]> writerList = new ArrayList<String[]>(); //写
        List<String[]> contentList = getFileContent(file);
        int k=0;
        String[] content2=new String[contentList.size()];
        boolean tag=true; //用于标记是否做了修改
        for(int i=0;i<contentList.size();i++){
            content2=contentList.get(i);
            for(int j=0;j<content2.length;j++){
                if(i>=9&&i<=18&&tag){ //html 中从9到18行是排名的代码
                    if(Integer.parseInt(content2[5])<Integer.parseInt(Score)){

```

```
String[] insert = new String[] { "<p><font", "color=\"#374D76\">第", "玩家名次", "名", "玩家名字", "玩家分数", "分", "</font></p>";
insert[2]=(i-8)+""; //修改名次
insert[4]=Player_Name; //修改玩家名字
insert[5]=Score; //修改分数
writerList.add(insert);
tag=false;}}}
if(i>=9&&i<=18&&!tag) { //将排名往下压
int position=Integer.parseInt(content2[2]);
position++;
System.out.println("po="+position);
content2[2]=position+""; }
writerList.add(content2);
writerList.remove(19); //删掉第19条记录(被刷下去的排名)
System.out.println("开始\n");
String[] strTemp=new String[writerList.size()];
for(int i=0;i<writerList.size();i++){
strTemp=writerList.get(i);
for(int j=0;j<strTemp.length;j++){
System.out.print(strTemp[j]+" ");
System.out.println();
System.out.println("结束\n");
contentInputToHTM(writerList);}
```

这段代码主要是获取当前用户的资料以及用户所得分数,然后将用户所得分数与排行榜中的分数进行比较,若大于排行榜中的某一个分数,则将此用户的信息与分数插入到排行榜中,然后在调用显示排行榜的函数将其在排行榜界面显示出来。

### 3 总结

随着Java的广泛应用,以及Java开发系统的可移植性大大提高,Java开发的游戏广受欢迎。Java开发的游戏还是有一定的限制,画面表现、场景效果没有其他语言开发的那么好,但是Java开发的游戏反应比较快,其优势还是已初露端倪。

### 参考文献:

- [1] 周凤英,文惺. Java游戏开发基础[J]. 电脑爱好者, 2003(12):122.
- [2] 袁伟. VC++实现连连看游戏[J]. 电脑编程技巧与维护, 2010(1):79-84.
- [3] Eckel B. Thinking in Java[M]. 4th ed. Prentice Hall PTR, 2006.

---

(上接第9101页)

据备份还应当进行定期检查,定期进行复制,保证备份数据的完整性、使用性和时效性。信息安全技术是一门综合的学科,它涉及信息论、计算机科学和密码学等多方面的知识,它研究计算机系统和通信网络内信息的保护方法,以实现系统内信息的安全、保密、真实、完整和可用。

物流信息系统企业按照现代管理的思想、理念,以信息技术为支撑所开发的信息系统。物流信息系统充分利用数据、信息、知识、计算机网络等资源,实施物流业务、控制物流业务、支持物流业务、实现物流信息共享,以提高物流业务的效率,提高决策的科学性,其最终目的是提高企业的核心竞争力。因此信息系统的安全与保密是一项极其重要的信息系统管理工作。

### 参考文献:

- [1] 李东. 管理信息系统的理论与应用[M]. 北京:北京大学出版社, 1998:81-88.
- [2] 董丕明. 计算机信息管理基础[M]. 北京:中国铁道出版社, 2000:126-132.
- [3] 王立坤. 物流管理信息系统[M]. 北京:化学工业出版社, 2000:186-189.
- [4] 方勇, 刘嘉勇. 信息系统安全导论[M]. 北京:电子工业出版社, 2003:212-216.
- [5] 黄均勇, 蒋云. 物流信息管理[M]. 北京:北京理工大学出版社, 2010:240-245.