扫雷游戏

目录

[1系统分析 1](#_Toc484771280)

[1.1需求分析 1](#_Toc484771281)

[1.2可行性分析 1](#_Toc484771282)

[（1）引言 1](#_Toc484771283)

[（2）可行性研究的前提 1](#_Toc484771284)

[1.3项目计划书 1](#_Toc484771285)

[（1）引言 1](#_Toc484771286)

[（2）概述 2](#_Toc484771287)

[（3）项目团队组织 2](#_Toc484771288)

[2系统设计 2](#_Toc484771289)

[2.1系统目标 2](#_Toc484771290)

[2.2系统功能结构 3](#_Toc484771291)

[3系统开发及运行环境 3](#_Toc484771292)

[4创建项目 3](#_Toc484771293)

[5系统文件夹组织结构 4](#_Toc484771294)

[6系统主窗体设计 4](#_Toc484771295)

# 1系统分析

## 1.1需求分析

扫雷游戏是一个小型益智游戏，它可以锻炼人们的逻辑思考能力，尤其是在开发人的智力方面，很受人们的欢迎。本组该次小游戏程序设计使用java语言开发，包含二维数组以及java图形化GUI编程等方面知识。

具体需求概要如下：

1.初始化游戏内部数据，初始化窗体级控件并将其显示在屏幕上。

2.为防止用户第一次点击就触雷，当用户第一次点击后，此时生成才包含地雷的地图，此地图对用户不可见。

3.根据用户点击判断点击的是地雷或安全区域，右键对区域标记红旗或问号作为标记。

4.当用户正确标记所有的地雷或安全点开所有的安全区域后，提示扫雷成功并给出所耗时间，否则为扫雷失败。

## 1.2可行性分析

根据《计算机软件文档编制规范》（GB/T8567-2006）中可行性分析的要求，制定可行性研究报告如下。

### （1）引言

为了更好的提高广大学生的业余生活，锻炼学生们的思维能力。本团队决定以windows xp 扫雷为原型开发一款类似的扫雷游戏。

### （2）可行性研究的前提

系统的功能符合学生的思维。

方便地对游戏进行格式化重启。

系统的功能操作要方便、易懂，不要有多余或复杂的操作。

## 1.3项目计划书

### （1）引言

1）编写目的

展示对已学知识的理解消化程度。

2）背景

扫雷游戏– 参考资料。

### （2）概述

1）项目目标

运用已学知识进行对扫雷游戏的编码。

2）应交付成果

完整代码，综合训练报告以及展示PPT。

3）项目开发环境

开发本项目所用的操作系统是Windows 10系统，开发工具为MyEclipse。

4）项目验收方式与依据

答辩进行展示，老师及同学打分。

### （3）项目团队组织

|  |  |  |
| --- | --- | --- |
| **姓名** | **工作描述** | **贡献** |
| 张博强 | 负责需求分析、扫雷游戏控制层逻辑 | 40% |
| 王妍 | 负责扫雷游戏视图层GUI界面 | 30% |
| 李昕 | 负责扫雷游戏地雷地图生成 | 30% |

# 2系统设计

## 2.1系统目标

根据要求，制定目标如下：

* 操作简单方便、界面简洁美观。
* 运用相关Java知识。
* 预置多种难度并可自定义难度
* 游戏计时功能
* 采用git作为版本控制工具
* 双击提示数字可快速开启已知安全的空白块

## 2.2系统功能结构

JavaMineSweeper扫雷游戏的结构图

# 3系统开发及运行环境

本系统的软件开发环境及运行环境具体如下。

* 操作系统：Windows 10。
* JDK环境：Java SE Development KET(JDK) version 7。
* 开发工具：MyEclipse 2014
* 开发语言：Java。
* 运行平台：Windows10。
* 分辨率：1024\*×768像素及以上

# 4创建项目

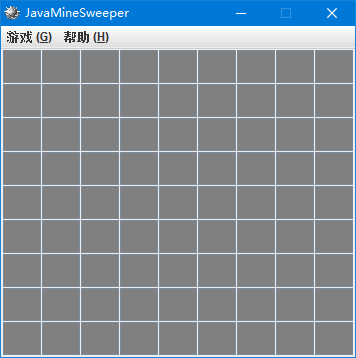
拼图游戏系统的项目名称为JavaMineSweeper，该系统是使用MyEclipse开发的一个项目。

# 5系统文件夹组织结构



# 6系统主窗体设计

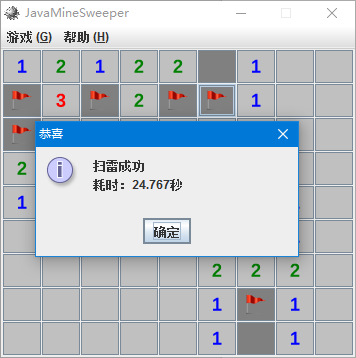
主窗体是程序操作过程中必不可少的，它是与用户交互的重要环节。通过主窗体，用户可以调用系统相关的各子模块，快速掌握本系统中所实现的各个功能。扫雷游戏中，用户将进入主窗体，主窗体中以菜单方式显示各功能按钮，单击这些菜单按钮的时候，打开相应功能。



扫雷游戏主界面（一）



扫雷游戏主界面（二）



扫雷游戏主界面（三）



扫雷游戏主界面（四）



扫雷游戏功能菜单



扫雷游戏自定义功能

# 7总结

扫雷游戏是一个小型益智游戏，它可以锻炼人们的逻辑思考能力，尤其是在开发人的智力方面，很受人们的欢迎。本组该次小游戏程序设计使用java语言开发，包含二维数组以及java图形化GUI编程等方面知识。

本游戏设计思路为：首先初始化游戏内部数据，初始化窗体级控件并将其显示在屏幕上。为防止用户第一次点击就触雷，当用户第一次点击后，此时生成才包含地雷的地图，此地图对用户不可见。然后根据用户点击判断点击的是地雷或安全区域，右键对区域标记红旗或问号作为标记。当用户正确标记所有的地雷或安全点开所有的安全区域后，提示扫雷成功并给出所耗时间，否则为扫雷失败。

游戏程序设计仿照windows xp经典的扫雷游戏制作，鼠标左键单击即可开始游戏，当点击空白块时将向用户展示提示数字或开启连续的空白块，双击数字可以根据已知情况快速开启安全的区域，当标记全部地雷或点开了全部都安全区域后，游戏将提示用户扫雷成功，并给出扫雷花费的时间。当一局游戏未结束时可随时通过菜单按钮重新开局。同时菜单中还可以设置游戏难度，默认提供了初级、中级、高级的难度，分别是9\*9、16\*16、22\*22，雷数分别是10、40、99，除此之外游戏还允许通过手动输入宽度和雷数自定义游戏难度。而当用户踩中地雷后，游戏结束并给出用户失败的提示，确认点击后将重新开始新的一局游戏。

程序采用分层设计思路，包括视图层和控制层，其中视图层负责对整个界面的绘制。而控制场，对整个界面的操作进行控制。在最顶层，由java鼠标监听器事件驱动地地图地图，人为规定零为空白块，-1为地雷，大于零的数字为提示数字。接着当用户第一次点击后随机放置地雷随机生成的来的坐标，已付一作为标志标记在地图上。接着根据地图上坐标的指示。分别计算出提示数据必应覆盖到地图上今天来有是吐槽创建用户可视窗口以及空间，游戏按组组成，根据游戏宽度循环创建按钮，同时为每个按钮添加鼠标的监听器。对点击的操作进行处理，分别处理左键双击左键单击以及右键的事件。当正确插旗后地雷计数器+1，错误插旗后地雷计数器-1，点开空白块空白块计数器+1,当地雷计数器==地雷数或总块数-空白块计数器==总雷数调用victoryMsg方法，弹出胜利消息并计算耗时。

Java是完全面向对象的语言，面向对象是一种程序设计思想，计算机程序的设计实质上就是将现实中的一些事物的特征抽离出来描述成一些计算机事件的过程，这种抽象的过程中，我们把具体的事物封装成一个一个的整体进行描述，使被描述的对象具备一些系统性、整体性的的行为和特征，我们把这种设计思想称为面向对象。

面向对象这种程序设计模式它将现实世界中的一切事物都看作是对象，例如，一个人是一个对象，汽车、飞机、小鸟等等，都是对象；它强调从对象出发，以对象为中心用人类的思维方式来认识和思考问题。每个对象都具有各自的状态特征（也可以称为属性）及行为特征（方法），java就是通过对象之间行为的交互来解决问题的。

类是面向对象中一个重要的概念。类是具有相同属性和行为特征的对象的抽象，类是对象的概念模型，对象是类的一个实例，通过类来创建对象，同一类的所有对象具有相同的属性和行为特征。类具有三个基本特征：封装、继承、多态。

封装就是将对象的属性和行为特征包装到一个程序单元（即类）中，把实现细节隐藏起来，通过公用的方法来展现类对外提供的功能，提高了类的内聚性，降低了对象之间的耦合性。

继承是对原有类的拓展，举例说明：我现在有一个Person类，但是我想要一个学生对象，他拥有Person类的所有属性和方法，此外他还有学号属性，及上课、写作业等一些方法，我可以创建一个Student类，但是我不想重复写Person类中已经有了的属性和方法，那么，此时我就可以用Student类继承Person类，Student类就拥有了Person类里的属性和方法了，我只需要在Student类里添加另外的新的属性和方法就可以了。Person类就成为父类，Student类就称为子类。父类和子类之间是一般和特殊的关系，子类是一种特殊的父类。此外，子类还可以通过重写来改变父类中的方法，重写可以改变方法的返回类型和访问权限，不能改变方法名称。

多态是建立在继承的基础上的，是指子类类型的对象可以赋值给父类类型的引用变量，但运行时仍表现子类的行为特征。也就是说，同一种类型的对象执行同一个方法时可以表现出不同的行为特征。

对比面相过程思想，面相对象程序思想在设计复杂程序中具有很大的优势。

# 8心得体会

通过此次课程设计，我深刻体会了面向对象程序设计的思想，java语言万物皆对象，在解决一个问题时，java语言将问题看作对象进行处理，而不是将问题拆分成子模块，因此开阔的程序设计思路，而不会被程序设计的惯性思想而使程序设计复杂化。目前程序设计的还是有些面相过程的痕迹，很多抽象出来的对象还有些不合理，在以后我将不断实践，更加深刻理解面向对象的精髓。