扫雷游戏项目计划书

一、概述

1.1 选题的目的及意义

本次课程设计的主要目的是为了通过具体的程序来加深Java语言的掌握，提高自己的编程水平。选择的题目来自《Java课程设计（第三版）》中的扫雷游戏，这是一个综合性的题目，可以对Java中的各项功能有更好的理解和使用。

1.2 程序设计任务与要求

使用Java语言编写一个与其类似的扫雷游戏。具体要求如下：
（1）、扫雷游戏分为初级、中级和高级，扫雷英雄榜存储每一个级别的最好成绩，即挖出全部的地雷且用时最少者。单击游戏菜单可以选择、初级、中级或高级查看英雄榜。
（2）、选择级别后将出现相应级别的扫雷区域，这时用户单击雷区中的任何一个方块便启动计时器。
（3）、用户要揭开某一个方块，可单击它。若所揭开的方块是雷，用户便输了这一局，程序发出爆炸的声音。若所揭开方块不是雷，则显示一个数字，该数字表示和和该方块相邻的方块中是雷的方块总数（相邻方块最多可有8个），同时将周围不是雷的方块揭开。
（4）、如果用户认为某个方块是雷，在方块上右击，可以在方块上标示一个用户认为是雷的图标（再单击一次可取消所做的标记），即给出一个扫雷标记，相当于扫雷期间在怀疑是雷的方块上插个小红旗。用户每标记出一个扫雷标记（无论用户的标记是否正确），程序就把“剩余雷数”减少一个，并显示该剩余雷数。
（5）、扫雷胜利后，如果成绩进入前三名，程序会弹出保存成绩的对话框。

二、需求分析

2.1 扫雷游戏的需求分析

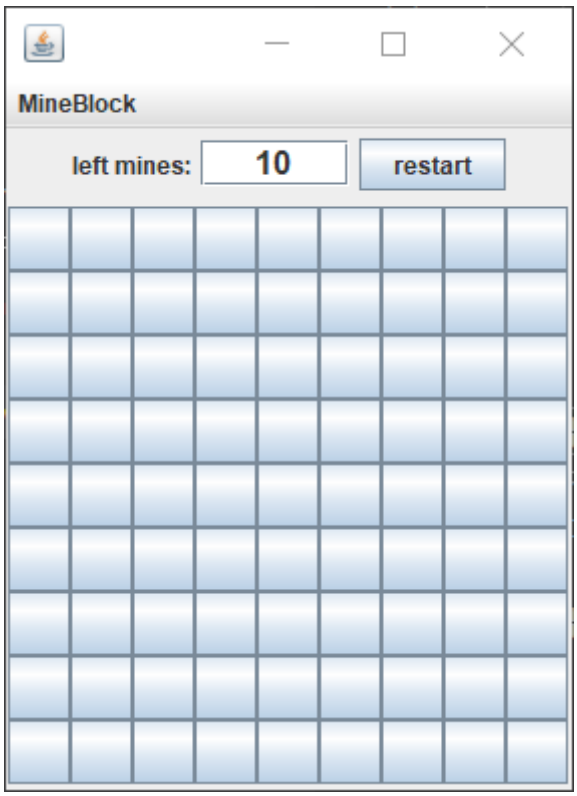
随着社会进步，人们的压力也不断的提高，在游戏层出不穷的时刻，扫雷游戏仍然受到不少人的青睐。不需要特殊技巧、无需太多熟悉，程序小巧，且具有一定锻炼逻辑思维能力的功效。

2.2 程序需要实现的主要功能：

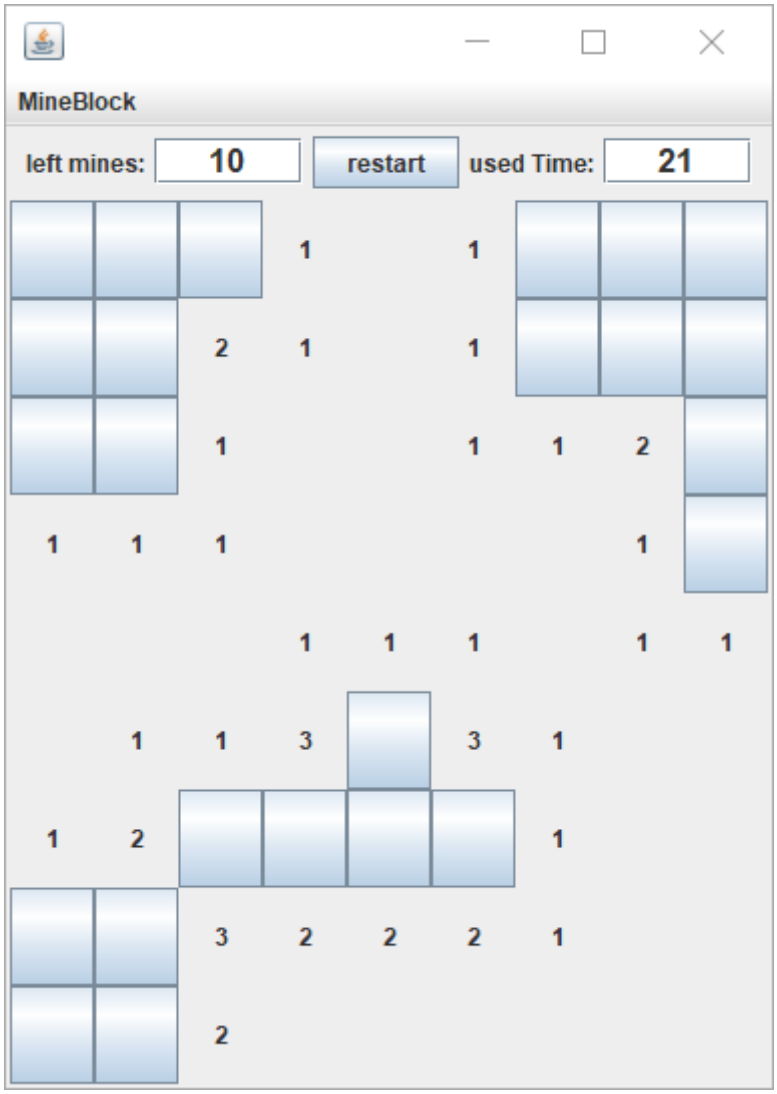
(1) 用户可以自定义级别，并确定雷的个数。
(2) 具有计时功能，即显示用户完成扫雷所花费的时间。
(3) 点击开始进行扫雷。
(4) 用户识别雷右击可标记“雷”。

2.3 功能模块如下：

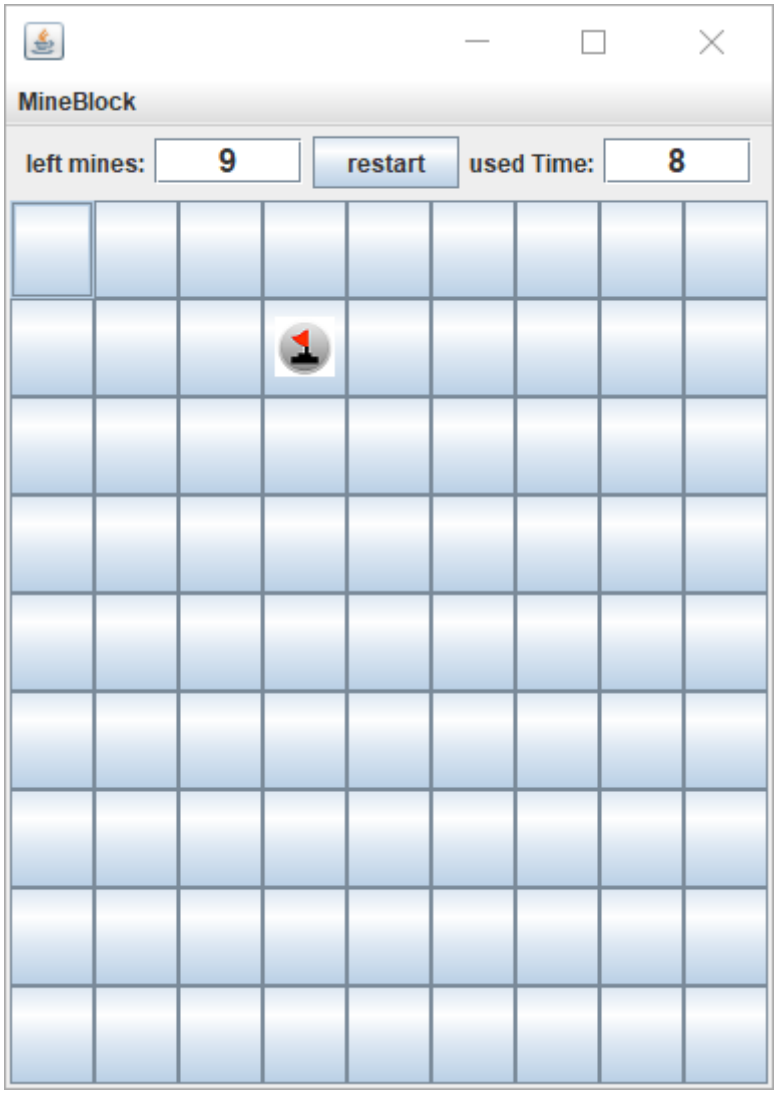
（1） 游戏界面（图）



(2) 鼠标事件（图）

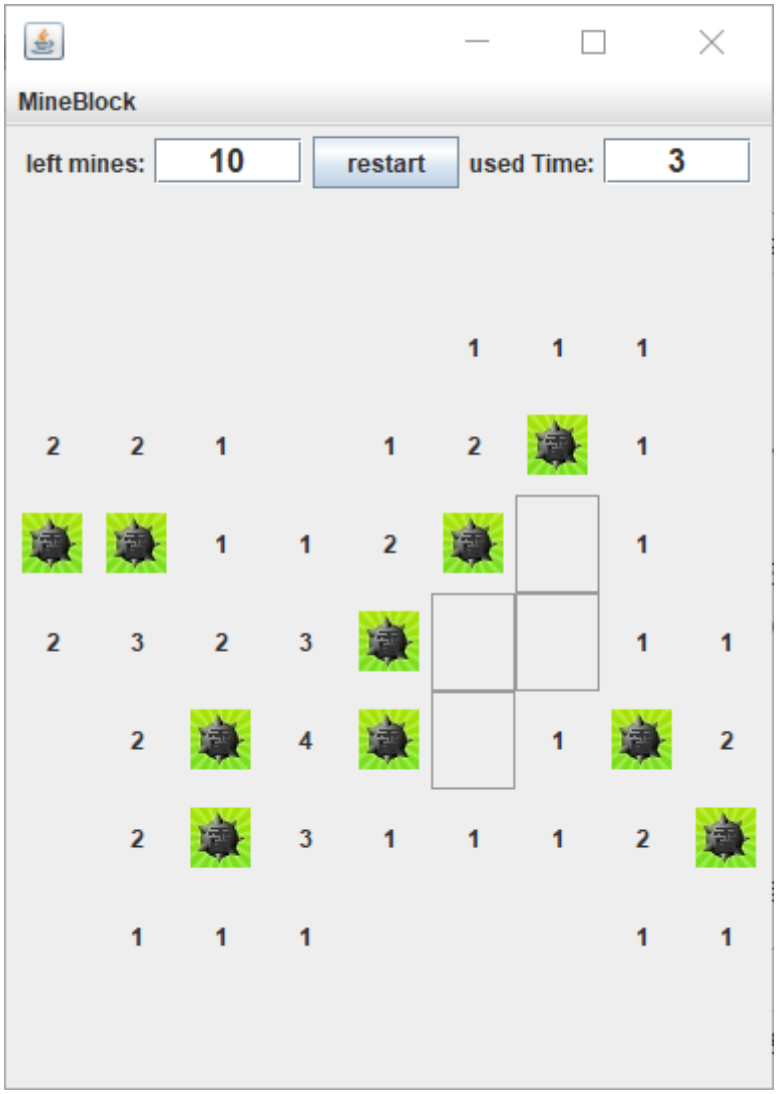


(3) 地雷判断（图）

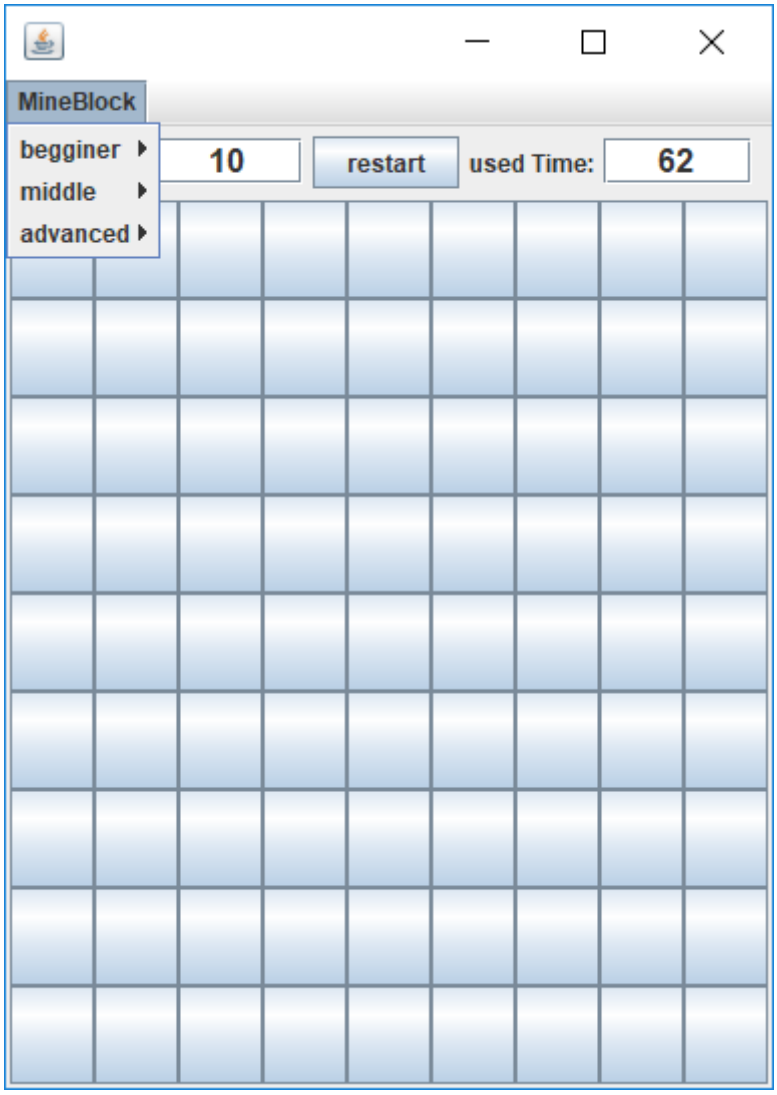


(4) 游戏胜利（图）

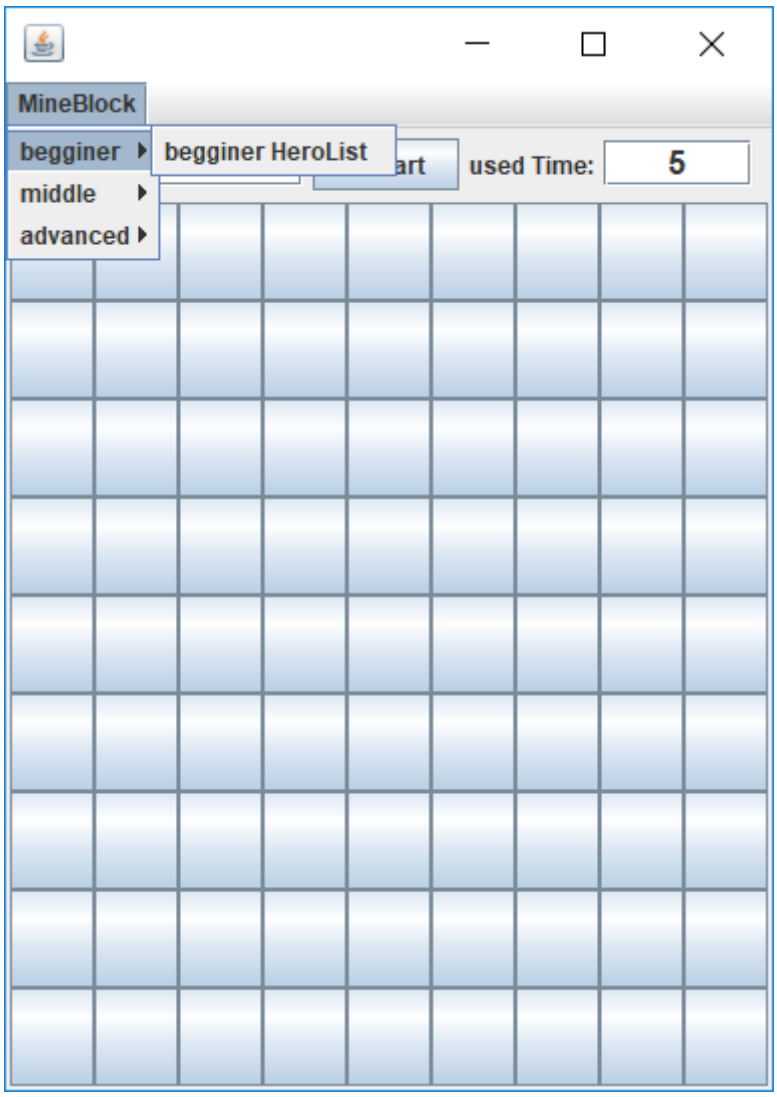
(5) 游戏结束（图）



(6) 游戏设置（图）


MineBlock
begginer
middle
advanced
10
restart
used Time:
62

(8) 查看英雄榜（图）



三、总体设计

3.1 算法思想

（1）计算每个方块周围存在的雷数。

（2）随机分布。扫雷游戏要求在M行N列的网格游戏区随机的布上n个雷，且n的取值应小于M和N的乘积:这可以在一张M*N的网格上通过均匀分布的随即算法视线。其中n的网格是雷区，剩下的网格是非雷区。游戏的目标是尽可能快地找到所有的雷区，而不踩到地雷。Java提供了视线随即算法的类Math，通过Math类的 random()方法这可以很方便的实现随即布雷的功能。

3.2 设计思想

3.2.1 扫雷棋盘的布局设计

系统的整体布局为: Cardlavout布局，采用了菜单、按钮、画板等组件。
菜单主要包括开始，选择级别，标记，按钮的功能是重新开始新的游戏。

3.2.2 类的设计

Block类:其实例是雷区中的方块。
Laymines类:其实例负责在雷区布雷
Peoplescoutmine类:其实例负责在雷区扫雷。
Viewforblock接口:规定了为方块制作视图的方法
Recordorshowrecord类:其实例负责读/写英雄榜。

3.2.3 数据模型

（1）方块

Bock的实例是雷区中的方块，方块可以是雷也可以不是雷。如果方块是雷，该方块的is Mine属性值就是tue，否则是flse。当方块的 imine属性值是 false时，该方块的aroundmine Number属性值是和该方块相邻且是雷的方块数目(一个方块最多可以有8个相邻的方块)。当该方块的 imine属性值是tue时，minelcon属性值是一个 Imagelcon图标的实例(地雷的样子)。

（2）布雷

Laymines类的实例负责在雷区布雷，即随机设置某些方块是雷。

（3）扫雷

Peoplescoutmine类的实例负责在雷区扫雷。该实例使用方法 Stack < Block > getnominearoundblock(Block bk)寻找不是雷的方块，并将找到的方块压入堆栈，然后返回该堆栈。
如果参数bk不是雷，但bk相邻的方块中有方块是雷，那么找到的不是雷的方块就是bk。
如果bk不是雷，但bk相邻的方块中没有任何一个方块是雷，那么就把相邻的方块作为getnominearoundblock(Block bk)方法的参数继续调用该方法，即 Peoplescoutmine类的实例用递归方法寻找一个方块周围区域内不是雷的方块，并将这些方块压入堆栈，返回该堆栈。
该实例使用方法 public boolean verify Wino判断用户是否扫雷成功。如果剩余的、没有揭开的方块数目刚好等于雷区的总雷数，该方法返回tue，否则返回 false。

（4）视图接口

方块需要一个外观提供给游戏的玩家，以便玩家单击方块或标记方块进行扫雷Viewforblock接口封装了给出视图的方法，例如 void acceptblock(Block block)方法确定该视图为哪个 Block实例提供视图。实现 Viewfor接口的类将在视图(View)设计部分给出，见稍后的 Blockview类。

（5）英雄榜

使用内置 Derby数据库 record存放玩家的成绩(有关内置 Derby数据库的知识点可参见Java2实用教程》第5版的第11章或本书的第3章)。数据库使用表存放成绩，即表示英雄榜。表中的字段 p name的值是玩家的名字，字段 p time是玩家的用时。玩家只要排进前3名就可以进入英雄榜，英雄榜上原有的第3名就退居到第4名(英雄榜记录着曾经的扫雷英雄)。 Recordorshow Record类的实例可以向英雄榜插入记录或查看英雄榜。

3.2.1数据库设计

字段名	类型	说明
name	varchar	获胜者名字
time	int	用时

扫雷游戏测试报告

一 引言

1.1 编写目的

1）查找并总结该程序所存在的问题。

2）为更改存在的问题，提供参考。

3）评估测试游戏执行与计划是否符合。

1.2 程序功能

1）扫雷游戏中各个功能的实现。

2）附加英雄榜功能实现。

1.3 测试对象

扫雷游戏程序

1.4 测试方法

黑盒测试

二 测试计划

2.1 测试环境

测试环境为window10系统，jdk8u222-b10；

2.2 测试条件

1）方块当前状态：方块初始状态、方块标记红旗，标识数字x且周围已标记了x个雷、标识数字x且周围没有标记完x个雷，标识数字x标识错误。

2）鼠标操作：左键、右键

3）方块状态：有雷、无雷

2.3 动作状态

1）方块白色

2）方块标识数字

3）方块旗子

4）炸弹爆炸，游戏结束

5）周围所以的非雷显示

2.4 规则

1） 方块当前状态 A

1.方块初始状态=A1

2.方块标识红旗=A2

2）鼠标动作 B

1.点击左键=B1

2.点击右键=B2

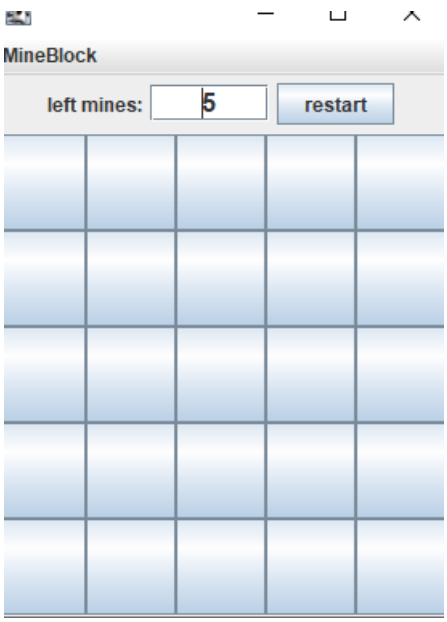
3）方块之后状态 C

1.有雷=C1

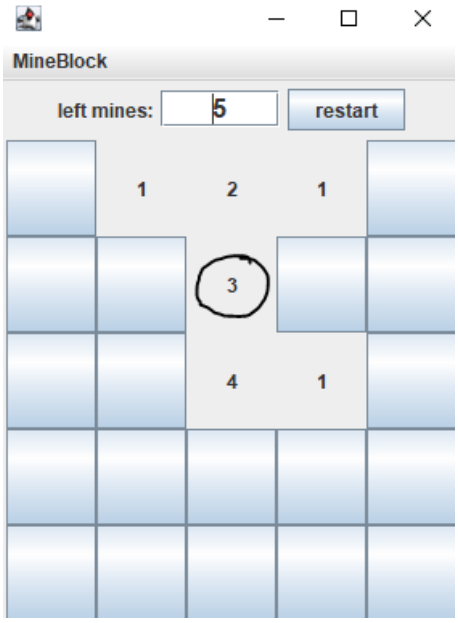
2.无雷=C2

2.5 扫雷操作图

1.运行扫雷程序

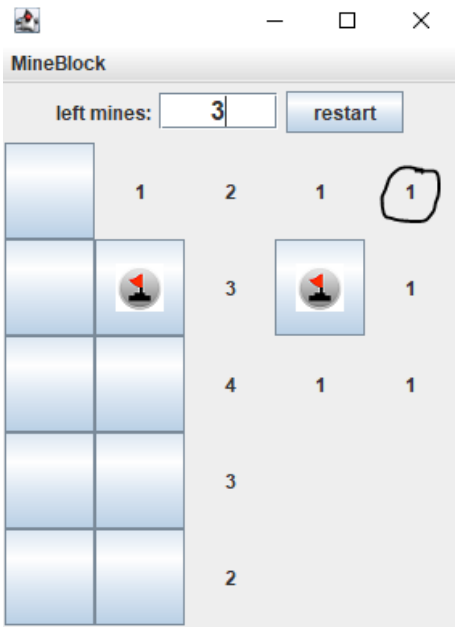


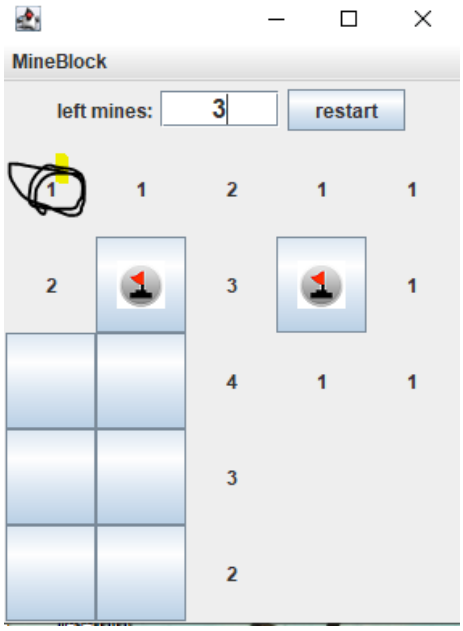
2.鼠标随机点击


MineBlock
left mines:
5
restart
1
2
1
3
4
1

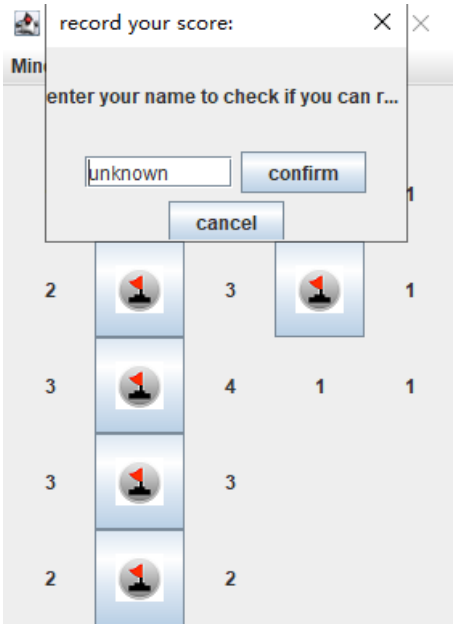
3.继续点击完成游戏


MineBlock
left mines:
3
restart
1
2
1
3
4
1


MineBlock
left mines:
3
restart
1
2
1
1
3
1
4
1
1
3
2

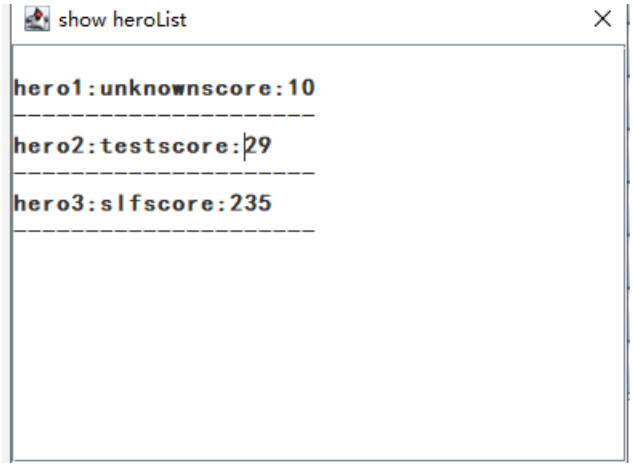


4.扫雷完成



5.点击红旗显示地雷

6.显示英雄榜



2.5 扫雷判断表

A	A1	A1	A1	A1	A1
B	B1	B2	B1	B1	B2
C	C2		C2	C2	
方块标识红旗		M			M
方块标识数字	3		1	1	
爆炸，游戏结束					
周围所以非雷显示	M			M	
上图扫雷步骤	1	2	3	4	5
完成游戏					M

上图程序运行与扫雷判断结果相同

2.6 例子

编号	难度	完成结果	布雷是否正确
1	简单	扫雷成功	正确
2	中等	扫雷失败	正确
3	困难	扫雷失败	正确

根据所显示方块中的数字与地雷个数对比，可知布雷正确。

2.7 测试代码

```
package ch8.test;


import ch8.data.***;

import java.util.Stack;
```

```
public class AppTest {
  public static void main(String[] args) {
    // 创建雷区
    Block block[][] = new Block[10][10];
    // 初始化雷区
    for (int i = 0; i < block.length; i++) {
      for (int j = 0; j < block[i].length; j++) {
        block[i][j] = new Block**();
      }
    }
    // 布雷者
    LayMines layMines = new LayMines**();
    // 扫雷者
    PeopleScoutMine peopleScoutMine = new PeopleScoutMine**();
    // 在雷区布雷，布雷10个
    layMines.layMinesForBlock(block, 10);
    System.out.println("雷区情况：");
    // self define method，显示布雷结果
    inputShow(block**);
    // prepare to scout mine
    peopleScoutMine.setBlock(block, 10);
    // 泛型,创建堆栈对象
    Stackstack = peopleScoutMine.getNoMineAroundBlock(block[0][0]);
    // 模拟扫雷，点击区域[0][0]
    if(block[0][0].isMine()){
      System.out.println("Oh my God! You Boomed!");
      return;
    }
    System.out.println("扫雷情况：");
    inputProcess(block,stack**);
    System.out.println("成功了吗："+peopleScoutMine.verifyWin());
  // 模拟扫雷，点击区域[3][3]
    if(block[3][3].isMine()){
```

```
    System.out.println("Oh my God! You Boomed!");
    return;
  }
  stack = peopleScoutMine.getNoMineAroundBlock(block[3][3]);
  System.out.println("扫雷情况：");
  inputProcess(block,stack**);
  System.out.println("成功了吗："+peopleScoutMine.verifyWin());
}
static void inputProcess(Block [][] block,Stack stack){
  // 这个k是干嘛的?
  // int k = 0;
  for(int i=0;i<block.length;i++){
    for(int j=0;j<block[i].length;j++){
      if(!stack.contains(block[i][j])&&block[i][j].getIsOpen()==false){
        System.out.printf("%2s","■ ");
      }else{
        // 显示周围雷的数量
        int m= block[i][j].getAroundMineNumber();
        System.out.printf("%2s","□ "+m);
      }
    }
    System.out.println();
  }
}

static void inputShow(Block [][] block){
  // 这个k是干嘛的?
  // int k = 0;
  for(int i=0;i<block.length;i++){
    for(int j=0;j<block[i].length;j++){
      if(block[i][j].isMine()){
        System.out.printf("%2s","#");
      }else{
        // 显示周围雷的数量
```

```
        int m= block[i][j].getAroundMineNumber();
        System.out.printf("%2s",m);
      }
    }
    System.out.println();
  }
 }

}
```

3.测试结果分析

3.1 结果分析

在程序代码基本完成后，经过不断的调试和修改，最后本次测试扫雷游戏能够正常运行且附加的英雄榜可以运行。但是还有一些不足，比如自定义模式下缺少英雄榜，希望以后可以做出改进。

扫雷游戏项目使用说明

一、环境

软件环境：jdk1.8

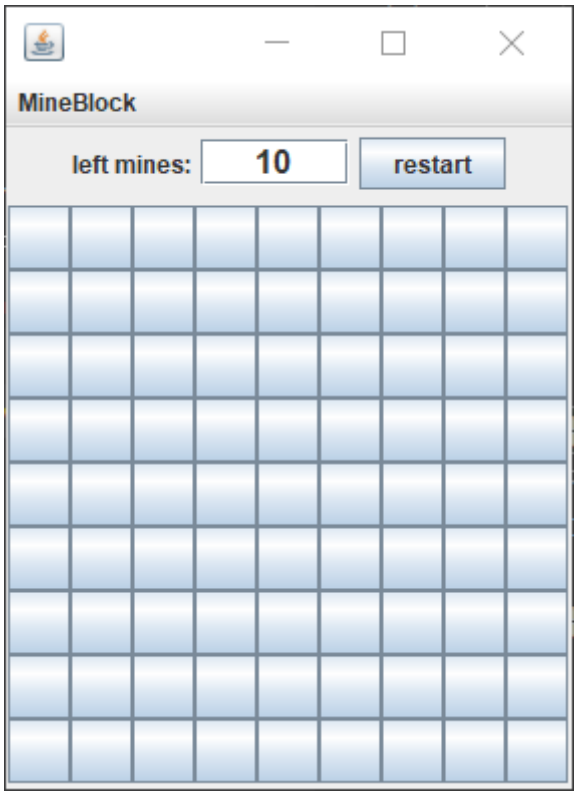
二、使用步骤

2.1 下载

1. https://github.com/uesugieriislf/mineSweeping/tree/1.0在github上下载压缩包
2. 解压文件，打开release文件下的mineSweeping.bat,运行前确保已经按照了java运行环境以及Derby数据库驱动配置，否则无法正常运行
   如何配置derby？将release文件夹中的derby，jar包放入jdk的jre/lib/ext中即可

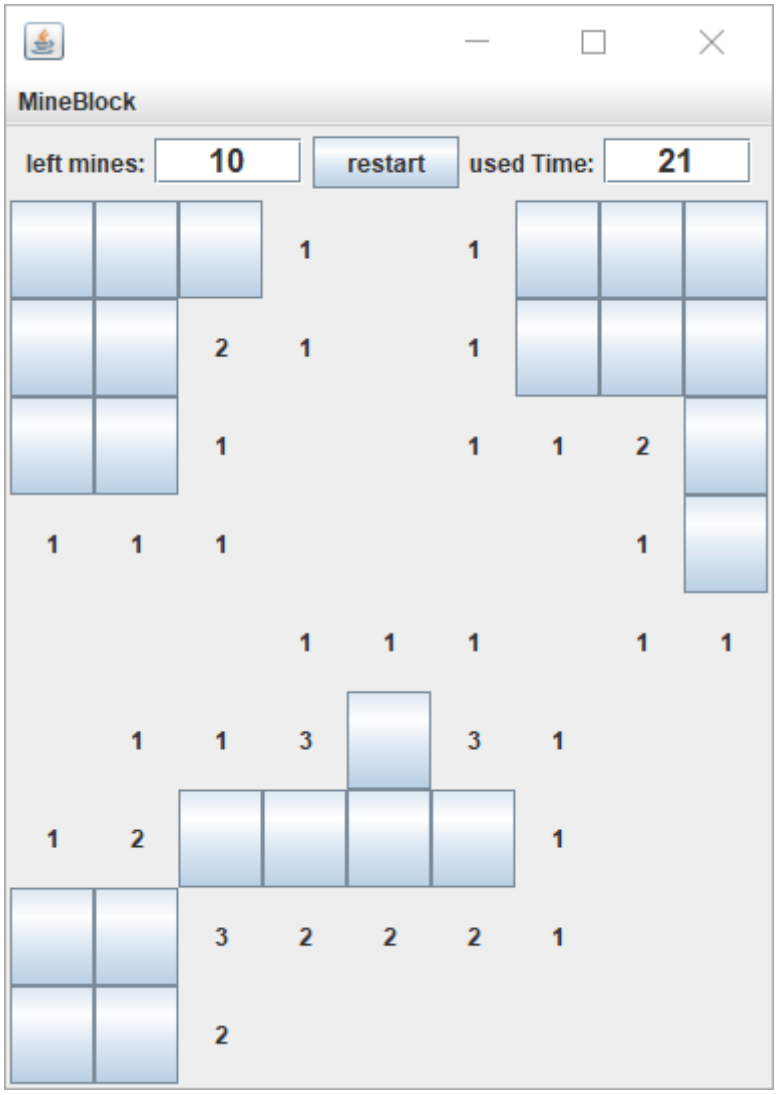
2.2 具体功能使用：

（1） 游戏界面（图）



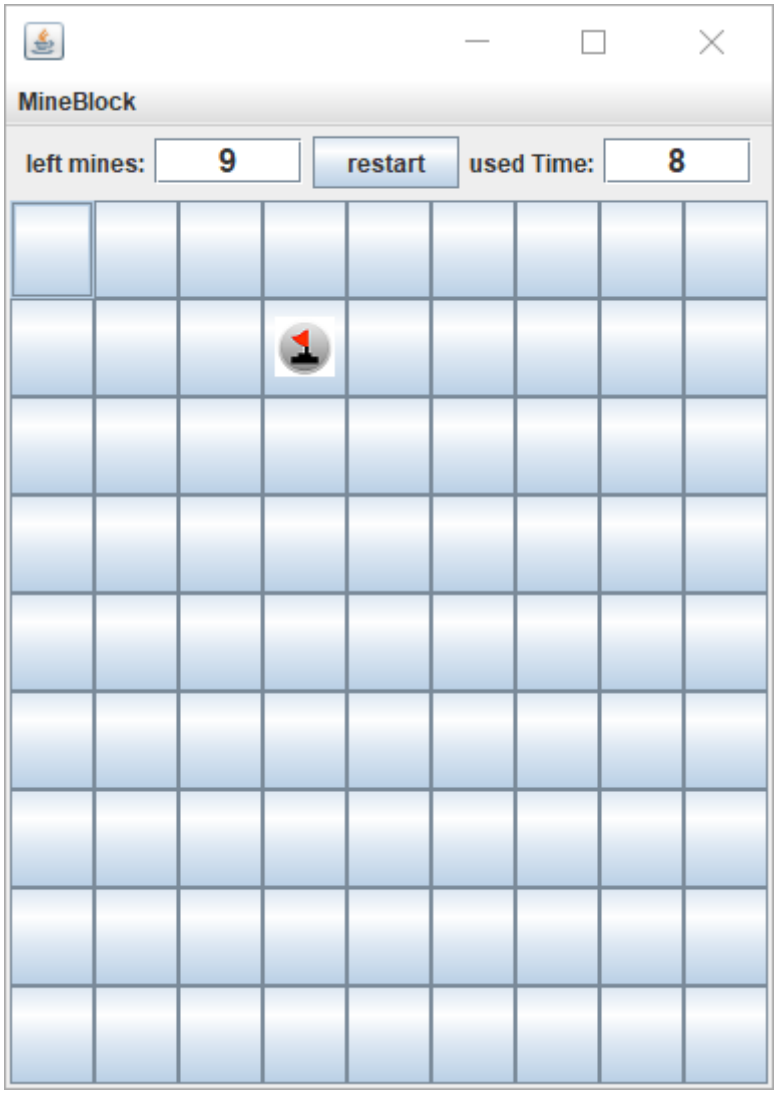
程序首先以初级难度运行

（2） 鼠标事件（图）



鼠标左键点击任意方块，如果没有碰到雷，则消除空白方块

（3） 地雷判断（图）

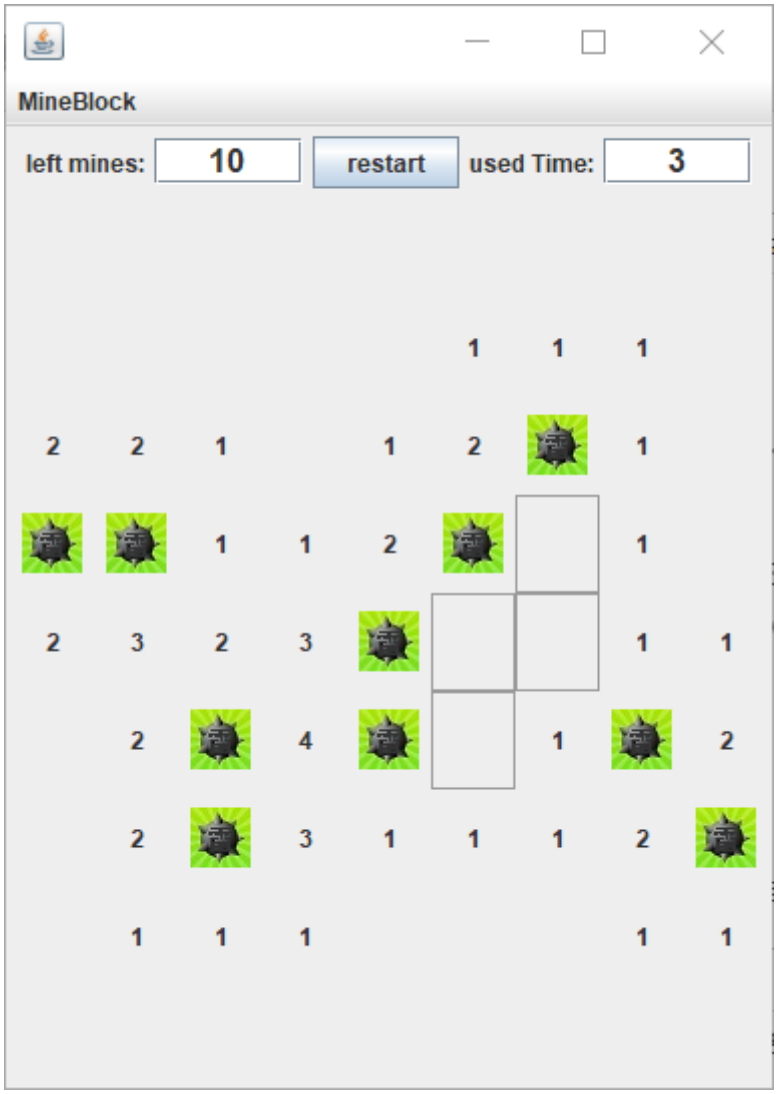


你也可以使用鼠标右键标记可能是雷的方块

（4） 游戏胜利（图）

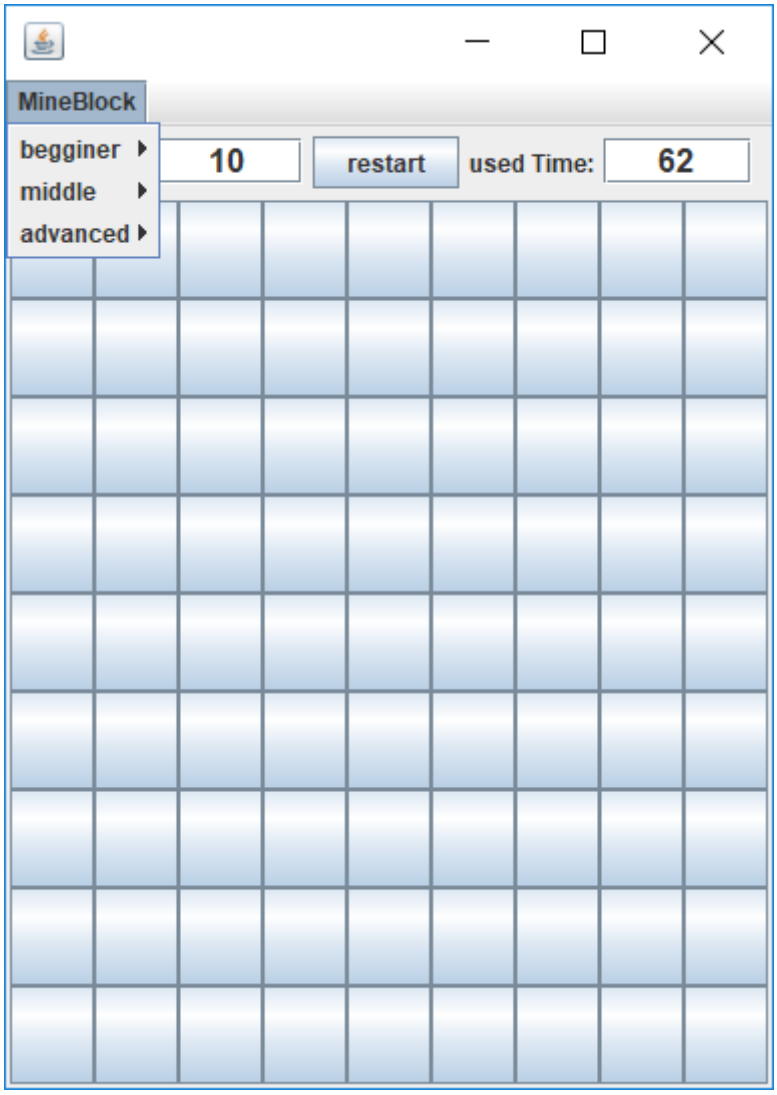
当你成功扫雷，会跳出输入框，输入你的名字，如果你的成绩达到英雄榜上榜条件，则可以在英雄榜中查看

（5） 游戏结束（图）



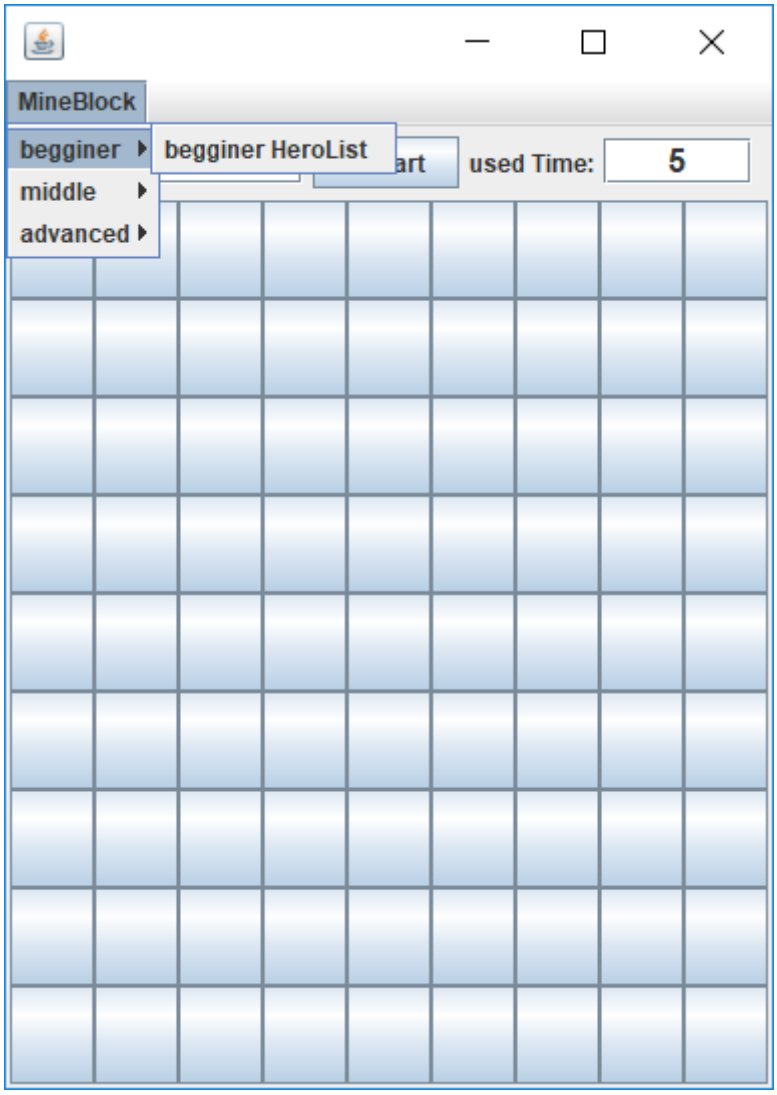
踩到雷则游戏结束

（6） 游戏设置（图）



游戏提供了3中预设模式，以及自定义模式。使用方法，点击MineBlock，之后鼠标移动即可切换，如果要跳出难度选项菜单，点击上方白色窗口即可

（8） 查看英雄榜（图）



难度选项菜单内部还有查看该难度英雄榜的功能