****

****

2020级软件学院 Java程序设计课程设计

街机游戏：三国无双

一、分工

|  |  |  |
| --- | --- | --- |
| 成员姓名 | 成员学号 | 主要职责 |
| 朱晨宇（组长） | 18373549 | 项目经理 |
| 王熙林 | 18373491 | 项目设计和实现 |
| 于鹏 | 17376273 | 项目实现和测试 |

二、项目目的

电子游戏是近几十年风靡世界的一项娱乐活动，一个好的游戏凝聚了开发人员的许多心血。从编程角度来看，游戏开发过程中充分使用了面向对象的思想——人物、物品、背景都是对象，可以从类的角度进行抽象，然后用编程语言进行实现。

我们小组的三个同学都是街机游戏爱好者，综合作业说明中提供了“实现横板游戏”这个项目，因此我们选择用Java实现一款经典街机游戏——三国无双。

本项目的目的是增强对面向对象思想的理解、提升代码水平、巩固课程知识。本项目的编写目标是实现一个用户体验好、功能齐全、画面美观的横板过关游戏。

三、项目运行环境

处理器（CPU）：Intel Xeon E5-2680 2.50GHz 以上

内存：2GB以上

硬盘容量：40GB以上

外设：鼠标、键盘

网络设备：宽带或无线网络

操作系统：Windows

JDK版本： 1.8

四、类图

如下图所示，由于内容较多，不是很清楚，所以在压缩包中放置了一个类图文件。



**图1 类图**

五、项目优势

（一）播放音乐

做一个游戏，背景音乐不可缺少，需要读取音乐文件并进行播放。

传统的AudioClip已经不被推荐使用，因此我们选择使用AudioInputStream类和AudioFormat类来接收音乐字节流，通过SourceDataLine不断write读到的数据，达到播放的效果。

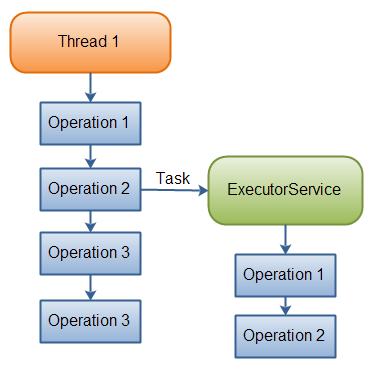
由于游戏时间很长，背景音乐需要循环播放，所以采用BufferedInputStream对AudioStream进行包装——因为后者不支持mark和reset。每次读到文件结尾之后reset，达到循环的效果。

（二）处理背景音乐与游戏线程的冲突

为了保证音乐播放时不对游戏画面产生干扰，我们选择多线程编程，将实现了Runnable接口的Music类实例创建为守护线程，在游戏开始时进行播放。

（三）处理攻击音效

人物攻击时需要播放攻击音效，我们通过Music类实现了这个功能。但是由于人物攻击次数非常多，攻击音效只有一秒钟，因此每次都新建一个线程的方法非常浪费资源，对性能有很大影响，所以我们使用了java.util.concurrent包中的ExecutorService，创建一个线程池，将播放音效作为一个任务，让 ExecutorService 中的某个线程去执行。这个流程如下图所示。



**图2 任务委托机制**

（四）使用线程安全的集合

用户按下按键后系统监听到信息，需要进行移动坐标、切换图片等操作，如果敌人被杀死还要将敌人从画面中移除，用一般的ArrayList进行删除增加时会出现ConcurrentModificationException，因此我们选用了写时复制的CopyOnWriteArrayList，它来自于java.util.concurrent包，是一个线程安全的集合类。

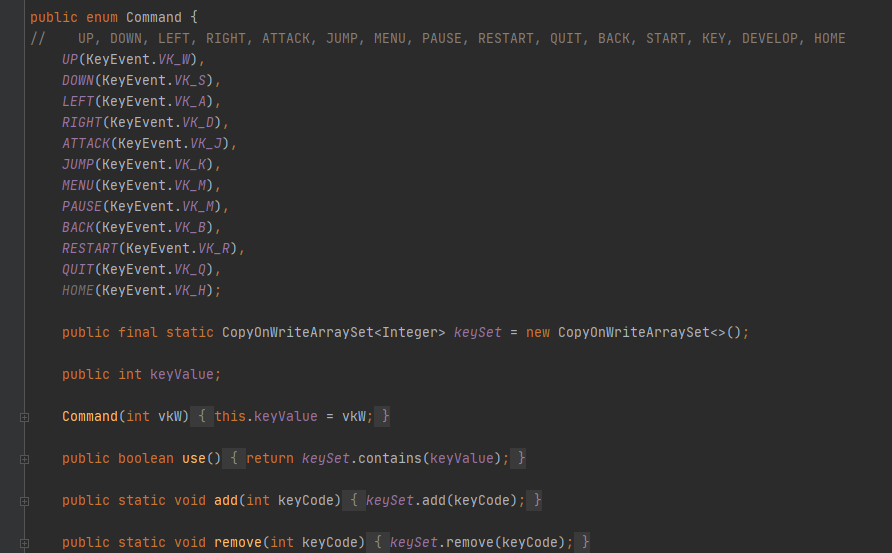
（五）用悲观锁来处理多线程问题

我们设置了一个主线程，用来重画画面，对游戏进行判定——人物HP、菜单操作等，还设置了一个控制敌人线程，用来控制敌人移动、修改坐标、判断生命值等，因此需要用锁来调度两个线程。

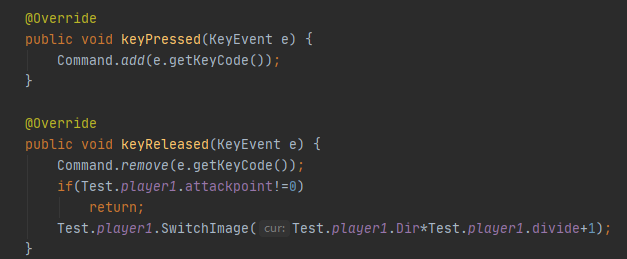
在主线程中，每进入一次循环都获得锁，然后进行操作，在一次操作结束后唤醒其它线程。控制敌人线程每次进入循环也要获得锁，在一次操作结束后唤醒其它线程，然后进入睡眠。

（六）提升键盘输入“手感”

如果简单地采用重写keyPressed方法，在每次按键压下之后直接操作，移动玩家、发起攻击等，会导致人物动作僵硬，玩家“手感”体验会比较差，于是我们对指令操作进行修改，在存储指令的枚举类中添加了add、use、remove这几个方法，并对keyPressed进行修改，如下图所示。



**图2-1 优化键盘输入：枚举类的修改**



**图2-2 优化键盘输入：重写监听者方法**

在每次监听到一个键入时，不立刻读取，而是存储到一个线程安全的CopyOnArrayList中。主线程中读取这个List，进行统一处理，这样就可以一次性地处理多个键入，让玩家可以向各个方向移动，而不是只能沿轴方向移动，有效提升了玩家体验。

六、项目实现

（一）流程实现

设置两个线程，主线程中不断循环，绘制图片，并对玩家血量等进行判断；另一个线程控制敌人移动。

使用JFrame和JPlane对画面进行绘制，为了便于输出，使用一个photo接口，将对象放在一个photo列表中，绘制时进行调用。

玩家和敌人类都继承自People类，每次调用People父类的方法进行移动、攻击即可，简化了代码。

（二）细节实现

我们设计了欢迎页、选择难度页、菜单页、修改自身属性页等页面。

在游戏进入时进入欢迎页，然后可以选择难度，之后便可以正式进入游戏。在游戏过程中，可以打开菜单暂停游戏，在菜单中修改自身属性、冲玩游戏等。在每个关卡结束后设置提升属性环节，增强了游戏体验。

七、错误，不足和改进空间

在游戏过程中，特别是游戏前期，由于图片加载等原因，人物可能会出现虚影，我们分析这可能与同时对图片进行读和写有关，由于自身能力和时间限制，我们没能很好地解决这个问题，希望在未来可以解决。

除此之外，我们的游戏还比较粗糙，相对正版游戏来说可玩性还不够强，没能实现物品掉落、多种武器、多种音效等内容，计划在未来进一步完善。