高级语言程序设计

实验报告

南开大学 计算机大类

蒋灿

2413779

网络空间安全模拟一班

2025年5月14日

目录

高级语言程序设计大作业实验报告

一. 作业题目

二. 开发软件

三. 课题要求

四. 主要流程

1，主要流程

2，各个类的功能

3，细节处理

五. 收获

高级语言程序设计大作业实验报告

**一、作业题目**

俯视角像素肉鸽弹幕射击游戏

**二、开发软件**

QT6，aseprite，

**三、课题要求**

面向对象。

组件化设计。

**四、主要流程**

**1，整体流程**

实现思路：通过计时器实现60帧的世界更新，使用QObject为父类实现空实体，实体通过维护和更新组件实现具体效果，具体参考了Unity的实现机制，并且通过Globals全局单例来传递数据，对于弹幕机制则通过组件化来实现弹幕命中，消失，计时的触发效果，从而实现肉鸽类游戏的局内build。

**2，各个类的功能**

AnimatedSprite组件继承自Component和QGraphicsPixmapView,参考了godot游戏引擎来实现序列帧动画的播放，通过Animation结构体来存储动画帧图片；

MotionComponent负责处理Entity的运动；

HealthComponent负责Entity的死亡，生命值与攻击机制，通过维护受击框和攻击框实现攻击与死亡，并提供了layer判断是否为目标（这里简化了Godot中碰撞层与遮罩层的处理）；

mainwindow类负责世界逻辑的更新，并且拥有freelist成员变量负责定时清除死亡的实体；

同时负责敌人与BOSS的生成机制；

Globals实现了常用数据在不同对象间传递，以及玩家的升级机制；

弹幕射击机制上，主要由玩家拥有Spell,Spell有Projectile成员变量作为投掷物，通过Spell

来实现射击；

Gameview负责监听玩家的输入事件 以及屏幕的初始化操作；

GameObject为游戏空对象，含有组件列表，提供了添加，删除，更新组件的功能；

Componenet是组件的基类，提供了添加，移除和更新时的空函数；

hud则负责游戏的UI管理，包括血条，法力值更新，道具的显示和其他信息；

MotionPath负责实现投掷物运动轨迹的实现，通过数组来实现路径形状的编程模式，并且在游戏中可以通过ModifySpell来替换以实现多样性；

Card为更新的实体，玩家击杀卡牌后可获得对应的升级，卡牌会定时随机生成。

**3，细节处理**

通过lerp插值实现玩家眼睛跟随鼠标平滑移动；

通过信号处理受伤，死亡，世界更新，升级机制；

利用单例与静态函数实现全局数据共享从而让各部分解耦；

卡牌与大部分敌人的素材来自itch.io，道具素材来自游戏《以撒的结合》解包，其他亲自画的；

音效来自爱给网；

**五，收获**

全面了解了c++的基础知识，充分运用了面对对象的概念来解决问题；

学会了基础的Qt知识和简单应用程序的开发；

充分意识到注释的重要性和短点调试的意义，学会看报错信息和debug；

学会了游戏开发的大体流程与组件化设计的机制；

掌握了释放对象的方法并认识到了其重要性；