

高级语言程序设计实验报告

南开大学 工科试验班

廖望

2210556

2023-04-24

Contents

1. 课题要求	3
2. 开发软件	3
3. 代码流程	3
3.1. 流程概览	3
3.2. 类简介	4
4. 技术难点与解决方式	5
4.1. 游戏配置常量繁杂	5
4.2. 地图滚动	5
4.3. 鼠标控制飞机	5
4.4. 控制子弹、敌机和爆炸效果的出现和消失	5
4.5. 弹匣	5
4.6. 碰撞检测和爆炸效果	5
5. 收获	6

1. 课题要求

设计一款名为“飞机大战”的游戏。玩家控制一架可以自动发射子弹的小飞机，如果子弹打到了飞下来的敌机，则射杀敌机，并触发爆炸效果。

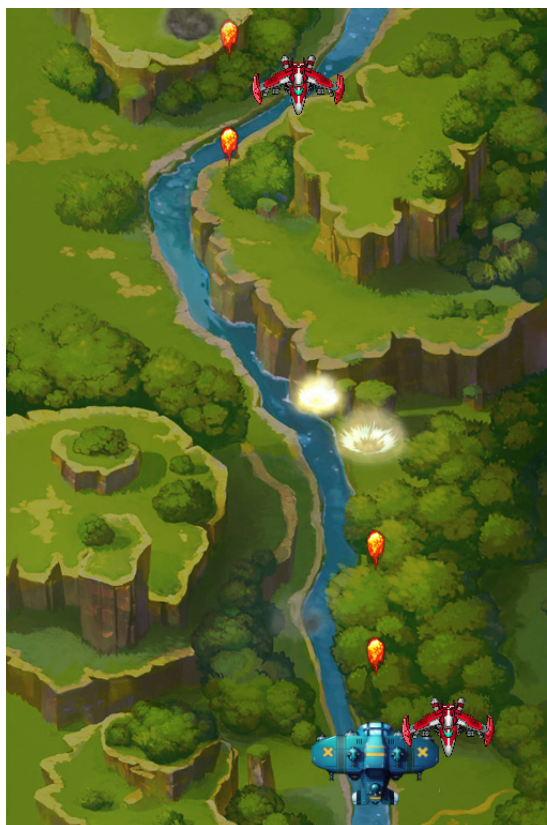


Figure 1: 游戏效果图

2. 开发软件

QT6.5、QT Creator 9.0.2.

3. 代码流程

3.1. 流程概览

1. 滚动的背景地图

2. 飞机的制作和控制
3. 子弹的制作和射击
4. 敌机的制作
5. 碰撞检测
6. 爆炸效果

3.2. 类简介

`MainScene`类继承自`QWidget`类，主要包含以下对象和函数：

1. 地图、计时器、英雄飞机、敌机和爆炸效果。
2. `initScene()`函数初始化游戏场景。
3. `playGame()`函数启动游戏和计时器。
4. `updatePosition()`函数更新坐标。
5. `paintEvent()`函数画图。
6. `mouseMoveEvent()`函数实现鼠标控制。
7. `enemyToScene()`函数实现敌机出场。
8. `collisionDetection()`函数实现敌机和子弹的碰撞检测。

`Map`类主要包含以下对象和函数：

1. 地图。
2. `mapPosition()`函数循环更新地图坐标。

`EnemyPlane`类主要包含以下对象和函数：

1. 英雄飞机及其边框和弹匣（一个`Bullet`类数组）。
2. `shoot()`函数实现子弹发射。

`Bullet`类主要包含以下对象和函数：

1. 子弹及其边框和使用状态。
2. `updatePosition()`函数实现子弹移动。

`HeroPlane`类主要包含以下对象和函数：

1. 敌机及其边框和使用状态。
2. `updatePosition()`函数实现敌机移动。

`Bomb`类主要包含以下对象和函数：

1. 爆炸效果及其使用状态。
2. `updateInfo()`函数实现爆炸切图。

4. 技术难点与解决方式

4.1. 游戏配置常量繁杂

游戏中存在十分繁杂的配置常量，如地图图片路径、飞机速度和子弹数量等等。我们建立一个`config.h`配置文件，统一存储这些常量，方便使用和修改。

4.2. 地图滚动

创建两个`QPixmap`类存储同一张背景图片，以均匀速度**循环**（意思是坐标加到上限就置0）增加两张图片的Y坐标实现滚动效果。

4.3. 鼠标控制飞机

通过`QMouseEvent`类获取鼠标坐标，之后就好办。

4.4. 控制子弹、敌机和爆炸效果的出现和消失

`Bullet`类、`EnemyPlane`类和`Bomb`类都有一个`bool`变量成员标示使用状态，值为0时表示非空闲，值为1时表示空闲。

`MainScene::paintEvent()`会枚举所有`Bullet`、`EnemyPlane`和`Bomb`对象，**只画出非空闲的的对象**。

4.5. 弹匣

将子弹处理成一个独立的`Bullet`类，它可以自行向前移动，这样`HeroPlane`类就只需要实现子弹发射，也就是在飞机正前方创建一个`Bullet`对象。

怎样实现弹匣（子弹数量控制）呢？

每次执行`HeroPlane::shoot()`函数时枚举一遍弹匣里的子弹，如果**非空闲**子弹就发射并**break**。

4.6. 碰撞检测和爆炸效果

`Bullet`类和`EnemyPlane`类都有一个矩形边框成员，每次执行`MainScene::updatePosition()`函数时，暴力枚举所有**非空闲**的子弹和敌机，判断它们的矩形边框是否相交。若相交，则把子弹和敌机的使用状态更新为空闲，并将一个空闲爆炸对象的使用情况更新为非空闲，坐标更新为碰撞位置的坐标。

5. 收获

1. 学习了 Qt 的安装技术，环境变量的配置等。
2. 学习了面向对象编程的方法。
3. 学习了 Qt 这款开发工具的使用方法。
4. 培养了做游戏的思路。
5. 收获了自己做出游戏的乐趣。