

# 现代操作系统应用开发实验报告

学号： 14331098

班级： 周四班

姓名： 黄建武

实验名称： HW12

## 一 . 参考资料

爆炸动画素材

<http://www.2gei.com/game2d/>

按住方向键一直响应

<https://segmentfault.com/q/1010000000578296>

回调函数的使用

<http://blog.csdn.net/elloop/article/details/50438032>

C++ list 容器 erase

[http://blog.sina.com.cn/s/blog\\_66f74d9f0100om0f.html](http://blog.sina.com.cn/s/blog_66f74d9f0100om0f.html)

触摸事件

<http://www.it165.net/pro/html/201503/35651.html>

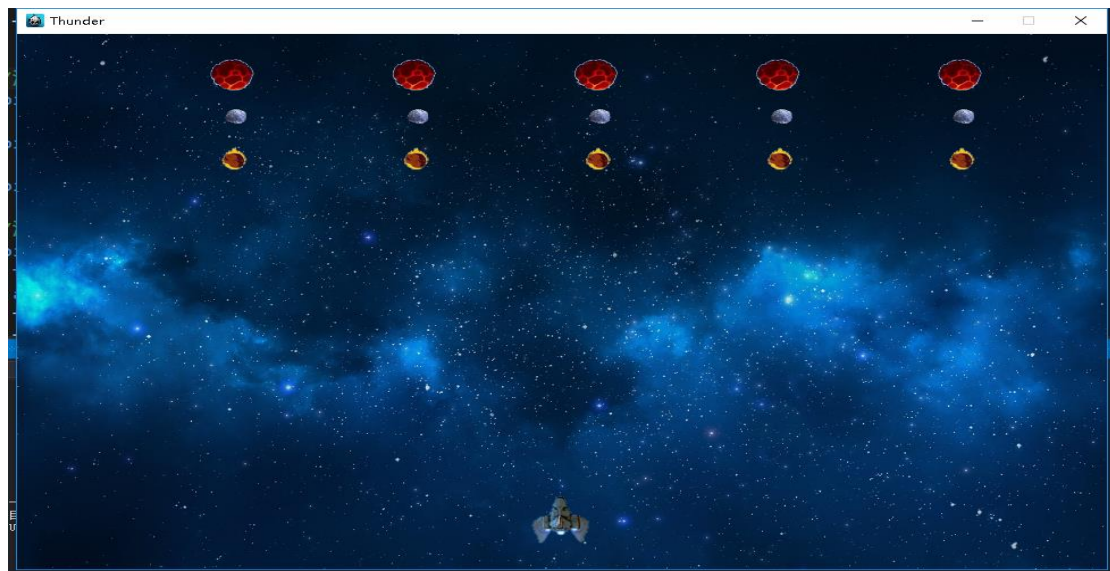
## 二 . 实验步骤

- ① 阅读 demo 的源代码，理解代码的思路。
- ② 上次作业已经实现了按住键盘控制移动，于是重用上次作业的代码，进行修改迁移；
- ③ 实现发射子弹并播放音效，播放背景音乐；
- ④ 实现碰撞检测，子弹打到陨石两者都消失并播放爆炸音效；

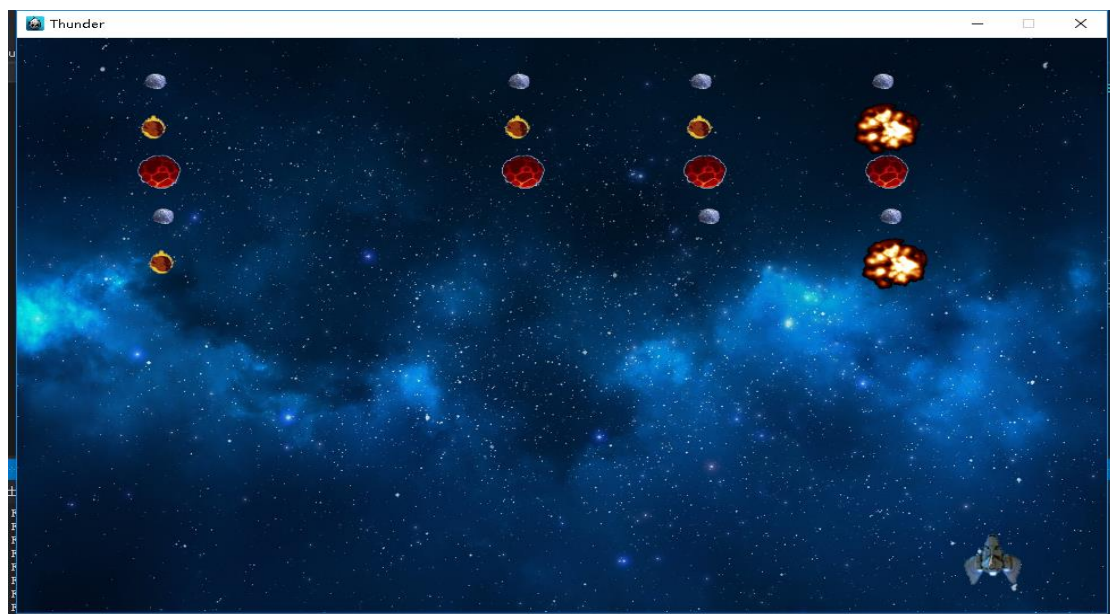
- ⑤ 实现游戏结束的检测；
- ⑥ 实现触摸事件，移动飞船和发射子弹；
- ⑦ 实现陨石来回移动后向下运动并且生成新的一行陨石；
- ⑧ 增加爆炸动画；
- ⑨ 进行调试，打包 exe

### 三．实验结果截图

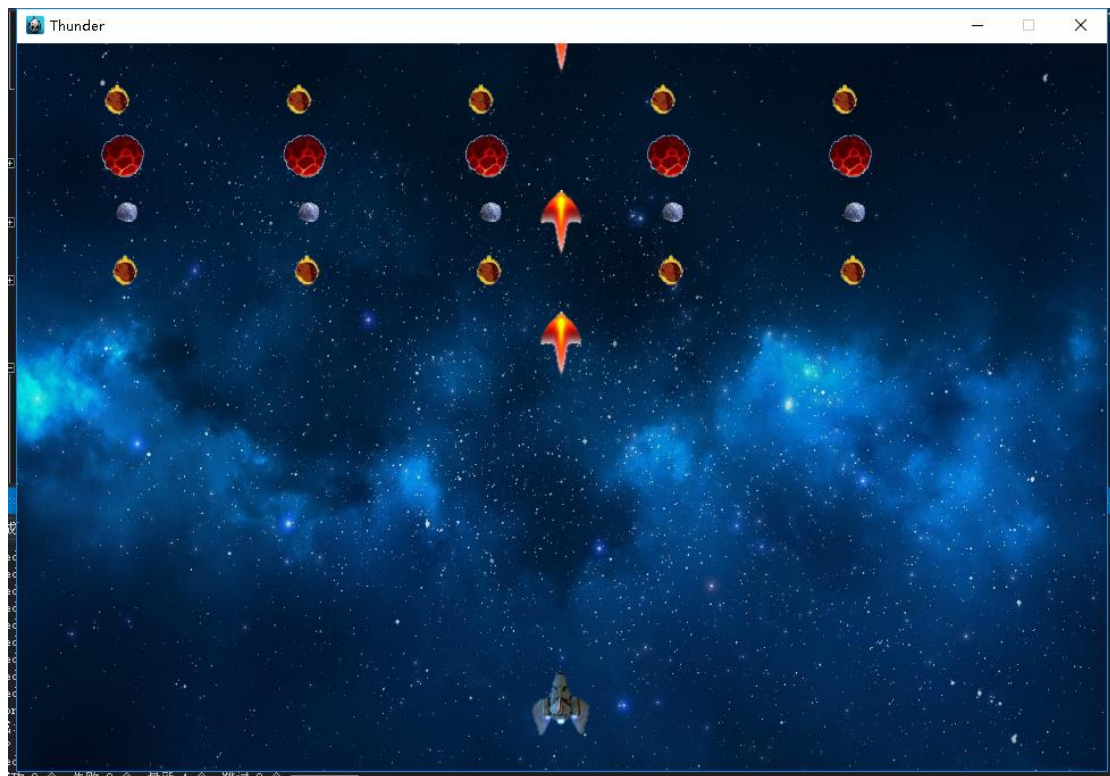
开始界面



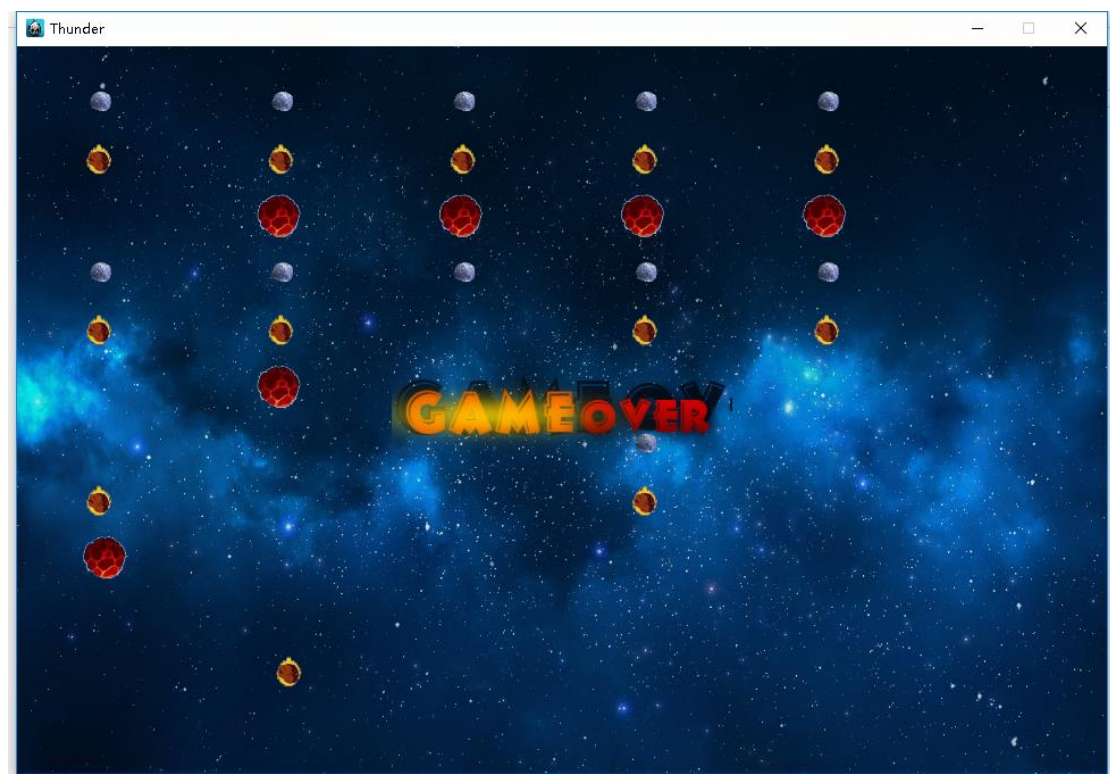
子弹打中怪物



一次发射多发子弹



游戏结束，飞机爆炸



#### 四．实验过程遇到的问题

- ① 判断子弹与陨石是否发生碰撞时，需要两层循环遍历，需要经常删除元素，使用数

组删除的开销很大，于是改用 list 储存，插入和删除都是  $O(1)$ 。

- ② 当子弹和陨石相撞就将两者都移除，一开始 list 的 erase 使用错误，而且移除后没有跳出内层循环，对空指针解引用导致程序崩溃。

```
for (auto it1 = bullets.begin(); it1 != bullets.end(); ) { //判断子弹是否打中陨石
    boom = false;
    for (auto it2 = enemys.begin(); it2 != enemys.end(); ++it2) {
        if (*it1 && *it2 && (*it1->getPosition().getDistance(*it2->getPosition()) < 25) {
            temp = *it2;
            (*it1->removeFromParentAndCleanup(true);
            (*it2->runAction(Sequence::create(Animate::create(Animation::createWithSpriteFrames(explore, 0.05f, 1)),
            CallFunc::create([this, temp] {
                temp->removeFromParentAndCleanup(true);
            }, nullptr)));
            SimpleAudioEngine::getInstance()->playEffect("music/explore.wav", false);
            boom = true;
            it1 = bullets.erase(it1);
            it2 = enemys.erase(it2);
            break;
        }
    }
    if (!boom) ++it1;
}
```

- ③ 判断游戏结束时移除飞机后也没有跳出循环，出现相同的问题。

```
for (auto it = enemys.begin(); it != enemys.end(); ++it) { //判断游戏是否结束
    if (*it && (*it->getBoundingBox().getMinY() <= player->getBoundingBox().getMaxY()) {
        temp = player;
        player->runAction(Sequence::create(Animate::create(Animation::createWithSpriteFrames(explore, 0.05f, 1)),
        CallFunc::create([this, temp] {
            temp->removeFromParentAndCleanup(true);
            auto gameover = Sprite::create("gameOver.png");
            gameover->setAnchorPoint(Vec2(0.5, 0.5));
            gameover->setPosition(visibleSize / 2);
            this->addChild(gameover, 1);
        }, nullptr));

        SimpleAudioEngine::getInstance()->playEffect("music/explore.wav", false);
        unschedule(schedule_selector(Thunder::update));
        this->getEventDispatcher()->removeAllEventListeners();
        return; // 游戏结束要退出循环，player已经是空指针，不能player->getBoundingBox()
    }
}
```

- ④ 播放爆炸动画的回调函数里想要删除对应的陨石或飞机，可是一直报错，访问权限冲突，最后只能新开一个变量赋值，再将该变量作为参数传入回调函数。

```
temp = *it2;
(*it1->removeFromParentAndCleanup(true);
(*it2->runAction(Sequence::create(Animate::create(Animation::createWithSpriteFrames(explore, 0.05f, 1)),
CallFunc::create([this, temp] {
    temp->removeFromParentAndCleanup(true);
}), nullptr));
```

## 五. 思考与总结

- ① 把程序分解成一个个小的部分，分而治之，更有效率而且更容易排错。
- ② 匿名函数功能很强大，在动画后面执行一个回调函数很方便，不过读写外部变量会产生读写权限冲突，以后使用要多注意。