

# 现代操作系统应用开发实验报告

学号： 14331006

班级： 教务二班

姓名： 毕心然

实验名称： HW13

## 一．参考资料

<http://www.aiuxian.com/article/p-2955040.html>

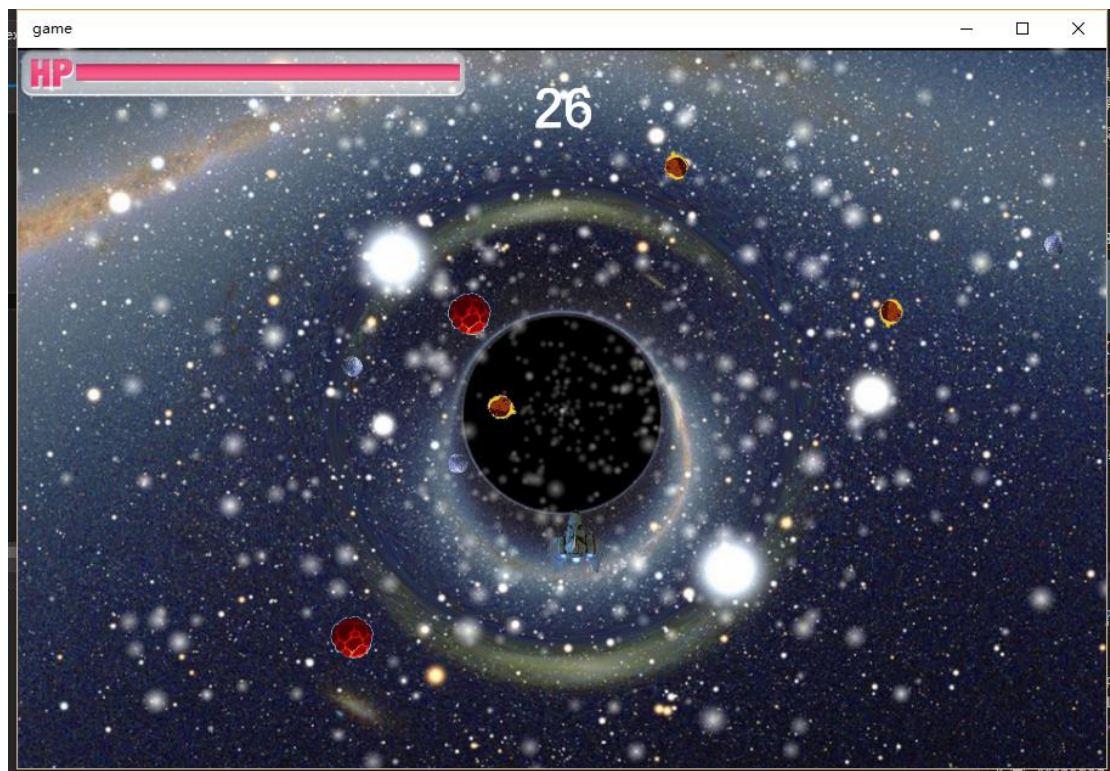
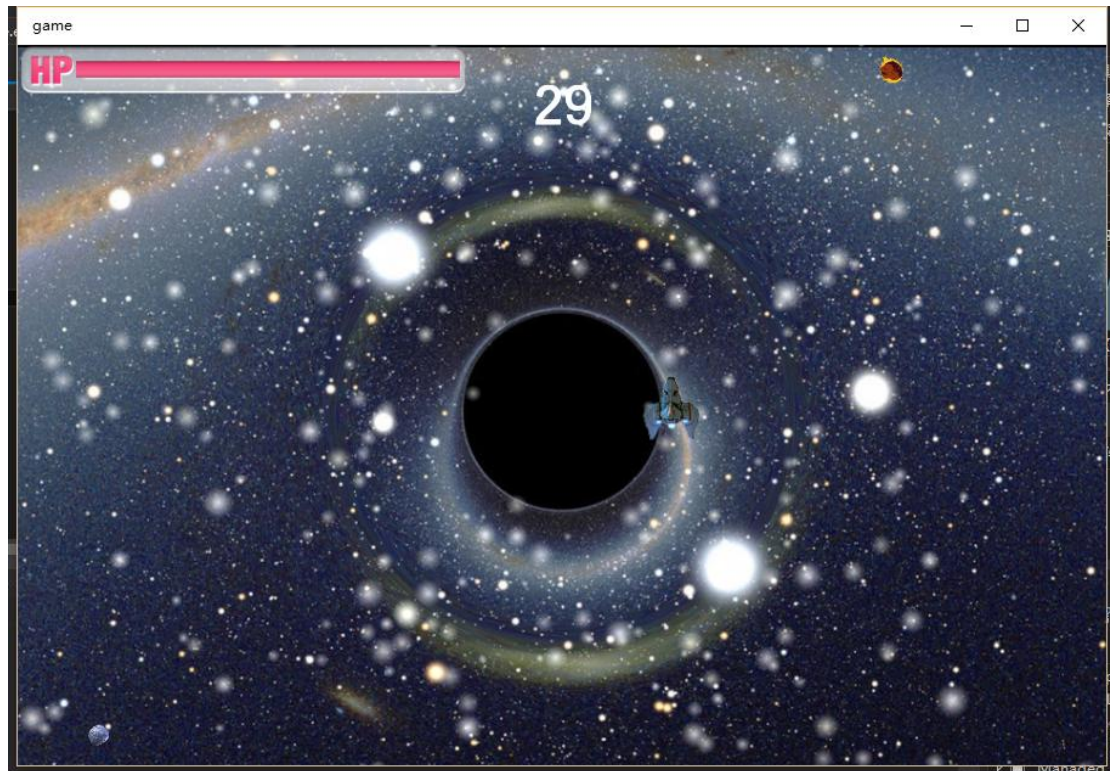
<http://www.cocoachina.com/bbs/read.php?tid=221969>

## 二．实验步骤

- 1、结合课件看懂掩码后在 demo 的基础上完成对飞船和陨石的 bitmask 的设置
- 2、完全飞船和陨石碰撞时的事件函数，删除陨石节点，添加爆炸效果等
- 3、添加倒计时 30 秒和血条，胜利和失败条件判断（30s 内碰撞<5 次胜利）

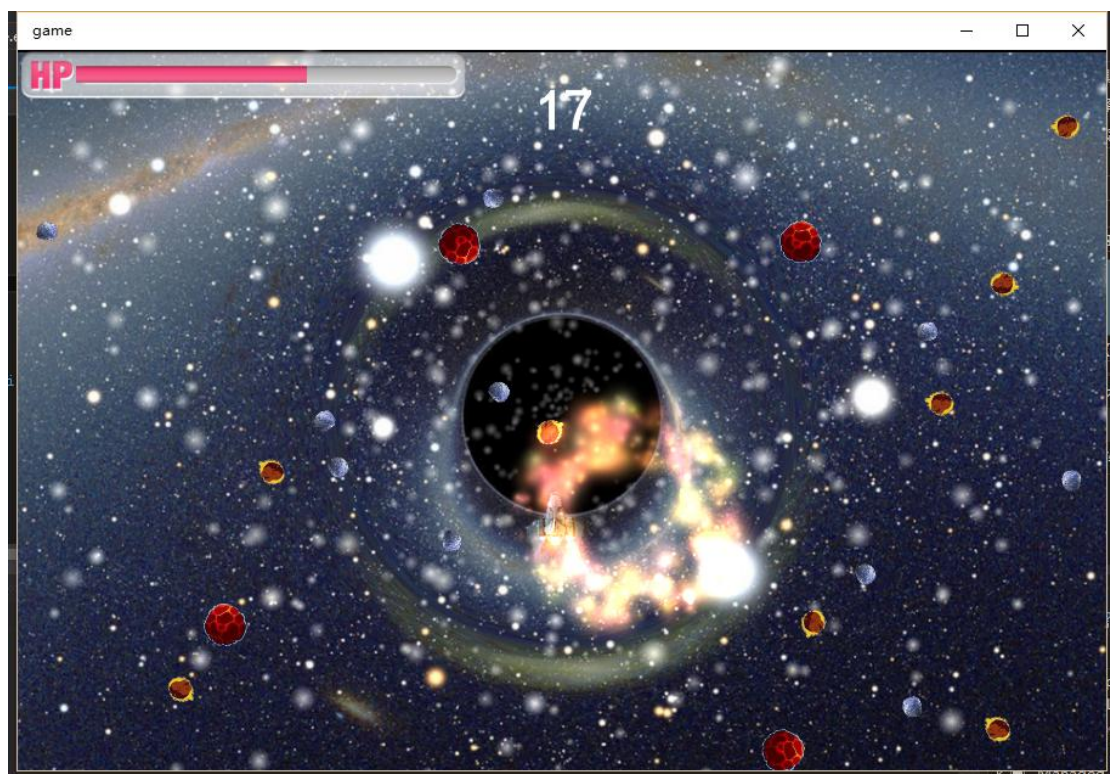
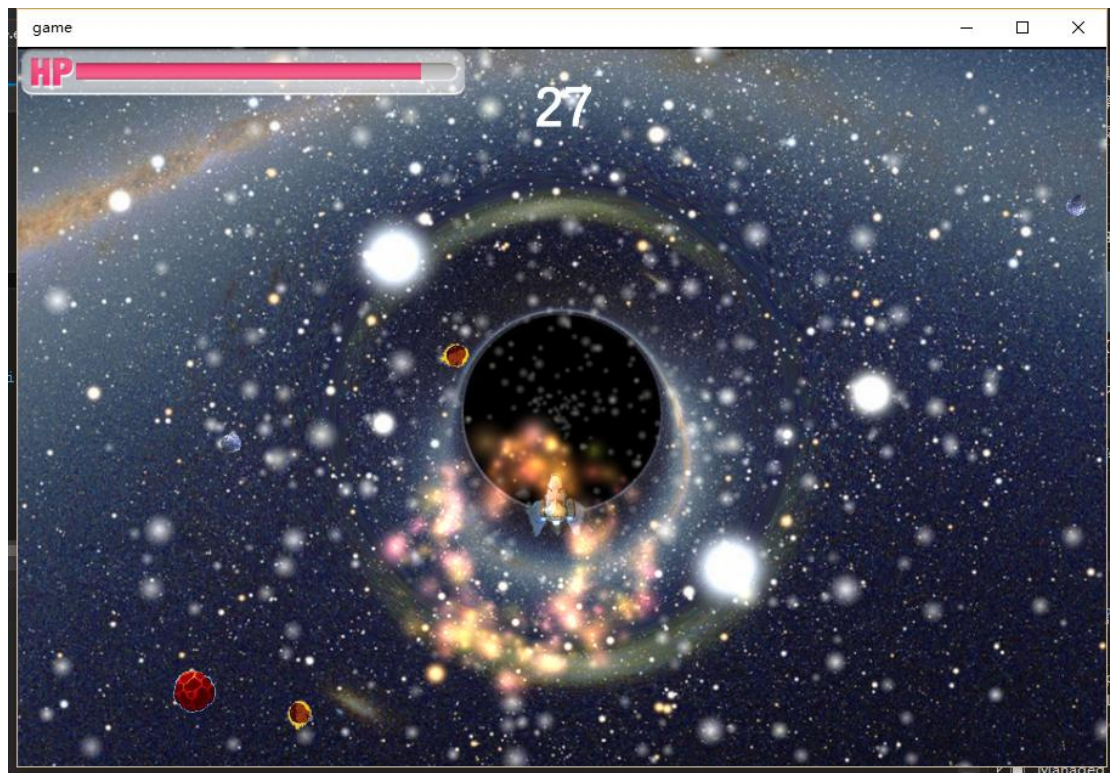
## 三．实验结果截图

1、游戏开始（倒计时 30s，血条为满）

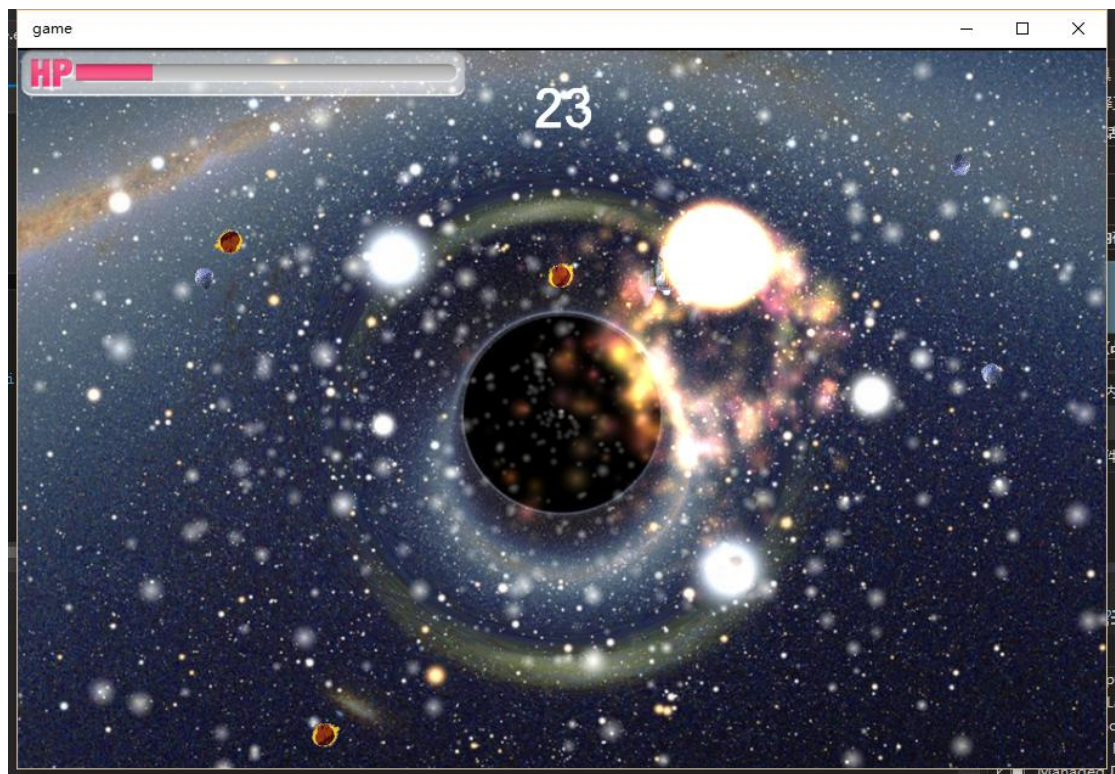




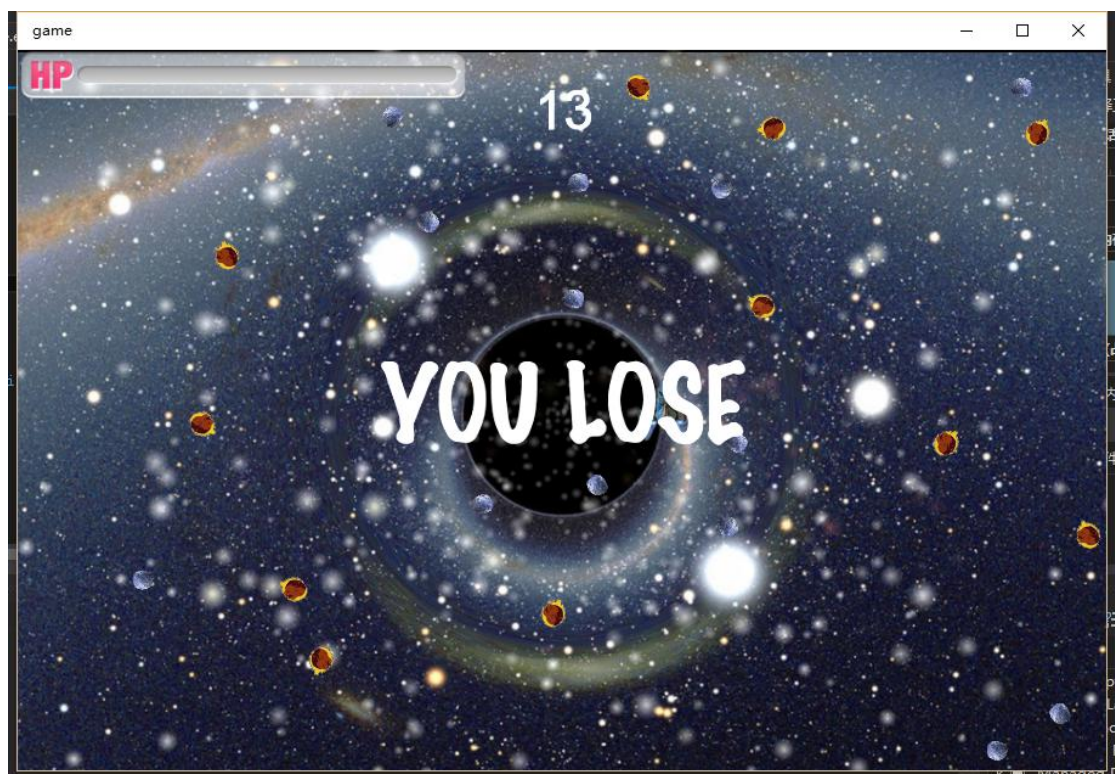
2、飞船碰到陨石，产生爆炸效果，陨石消失，血条减少（这里只减了百分之十...降低游戏难度方便截图，提交代码中碰撞一次减少百分之二十）



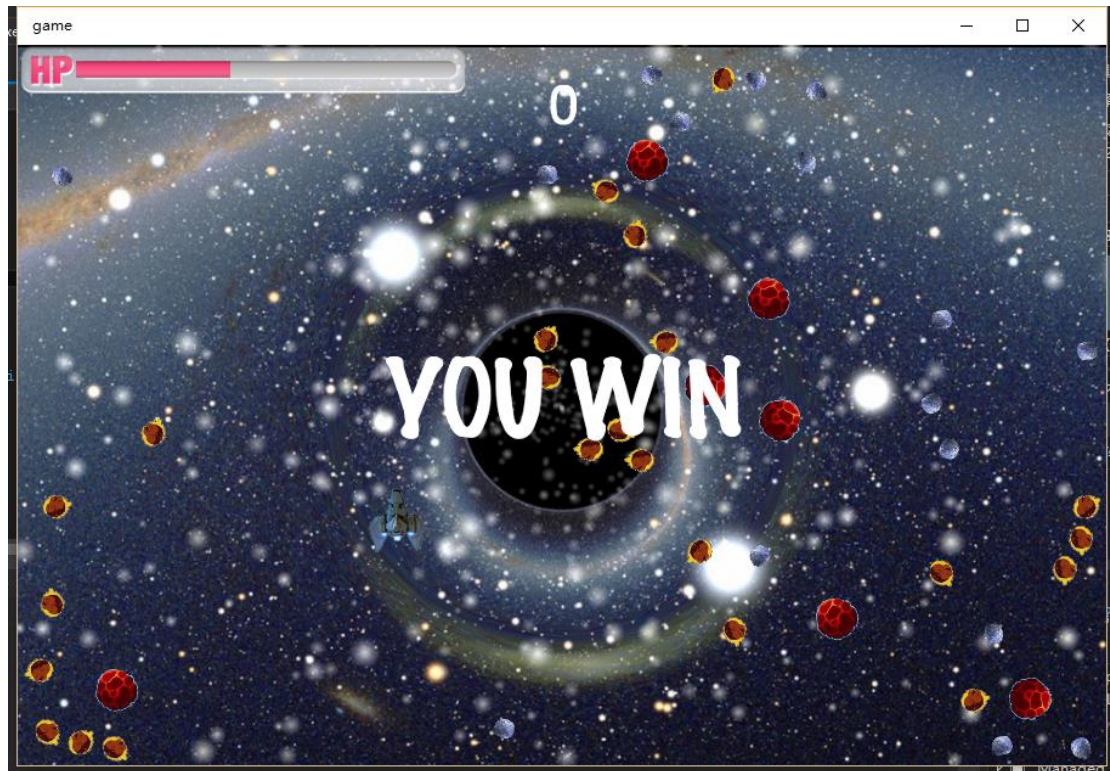




3、倒计时未到 0 前血条空了，输掉游戏



#### 4、倒计时到 0，血条未空，游戏胜利



#### 5、设置陨石和飞船的 bitmask 部分代码

```
// TODO set bitmask  
re->getPhysicsBody()->setCollisionBitmask(4);  
re->getPhysicsBody()->setContactTestBitmask(1);  
re->getPhysicsBody()->setCategoryBitmask(2);  
re->getPhysicsBody()->setGroup(16);
```

```
// TODO set bit mask  
//player->collisionBitmask(4);  
player->getPhysicsBody()->setContactTestBitmask(2);  
player->getPhysicsBody()->setCollisionBitmask(4);  
player->getPhysicsBody()->setCategoryBitmask(1);
```



## 6、飞船和陨石碰撞事件函数部分代码

```
bool Breakout::onContactBegan(PhysicsContact& contact) {  
    // TODO  
    Sprite* spriteA = (Sprite*)contact.getShapeA()->getBody()->getNode();  
    Sprite* spriteB = (Sprite*)contact.getShapeB()->getBody()->getNode();  
    auto explode = ParticleSystemQuad::create("explode.plist");  
    //auto explode = ParticleExplosion::create();  
    explode->setPosition(spriteA->getPosition());  
    this->addChild(explode);  
    if(spriteA->getPhysicsBody()->getTag() == 1)  
        spriteB->removeFromParentAndCleanup(true);  
    else spriteA->removeFromParentAndCleanup(true);  
  
    int temp = pT->getPercentage();  
    if (temp >= 20) {  
        pT->setPercentage(temp - 20);  
    }  
    return true;  
}
```

## 四 . 验过程遇到的问题

1、设置掩码时最开始只设置了 collisionbitmask 和 contacttestbitmask ,后来看网上博客教程和课件 pdf 中的部分源码才又设置了 categorybitmask ,但是陨石和陨石之间碰撞也会出发函数,于是又设置了陨石的 group ,完成了作业需求。

2、添加碰撞特效的时候编译总会报错,后来发现是.plist 文件的问题,用引擎自带的 explode 效果就可以正常运行。然后按照群里大神的说法将 black\_hole.plist 的最后一个标签加到 explode.plist 最后,重新编译就可以运行了。

3、碰撞的时候最初直接选择随意删除了一个节点,导致有时会误删飞船节点报各种错,后来加入了根据 tag 的判断,就不会有错了。

## 五 . 思考与总结

1、新的知识点需要结合课件和网上博客的教程及示例,从各种角度去理解并且进行尝试才能较为完整的理解一些函数及特性的用法。

2、现在各种游戏引擎已经非常强大,只要能好好学习利用好这些工具可以完成很多事实上并不难但可玩性较高的小游戏了。