

# 现代操作系统应用开发实验报告

姓名：袁之浩

学号：16340282

实验名称：lab5

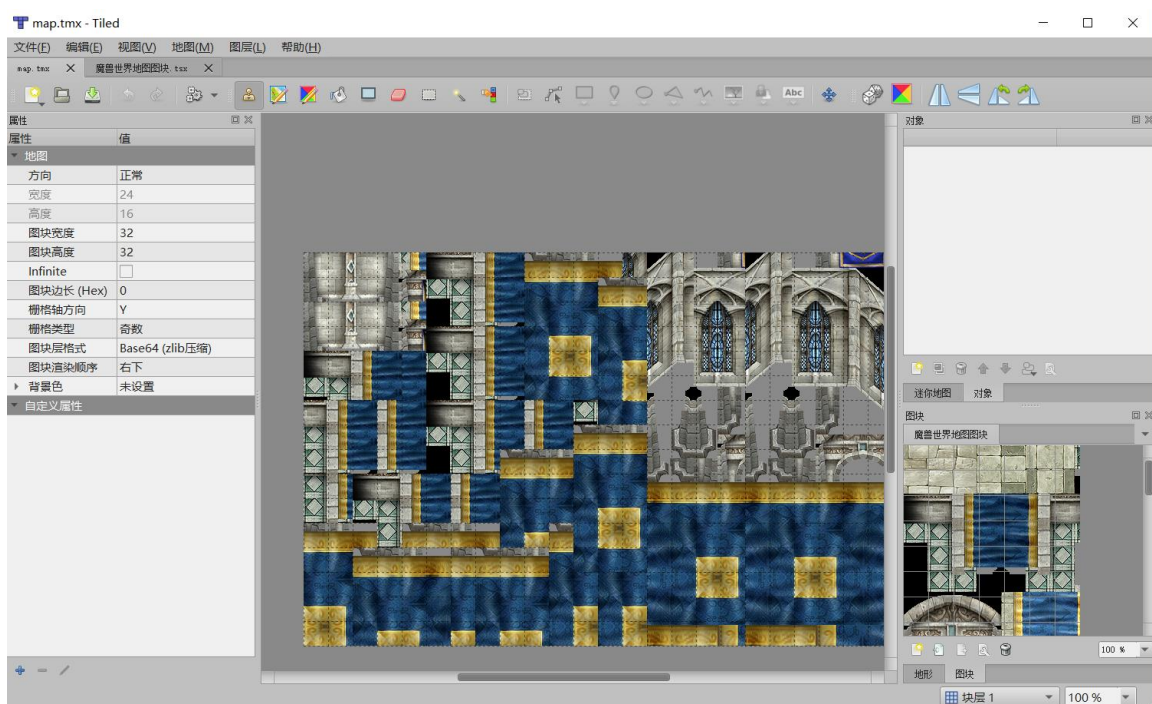
## 一、参考资料

<http://docs.cocos.com/cocos2d-x/manual/zh/>

## 二、实验步骤

week13

使用 Tiled 创建地图



使用一个循环产生若干怪物，并随机设置位置，调度器新增一个函数，调用工厂类的 moveMonster 函数，并传入 player 当前的位置，就可以使怪物向角色移动，同时调用 hitByMonster 检测怪物是否与角色碰撞，如果 fac->collider(player->getBoundingBox())返回值不为空的话，表明发生了碰撞，移出怪物，角色掉血，播放动画，如果血量为 0，gameOver，执行结束回调函数。角色能够攻击怪物也是采用同样的方法，写在攻击动画播放完的回调函数里。

## week14

本次作业的主要目的是学习事件处理和音效控制。

添加事件监听器，同时设置事件处理函数，记录按键

```
auto keyboardListener = EventListenerKeyboard::create();
keyboardListener->onKeyPressed = CC_CALLBACK_2(Thunder::onKeyPressed, this);
keyboardListener->onKeyReleased = CC_CALLBACK_2(Thunder::onKeyReleased, this);
_eventDispatcher->addEventListenerWithSceneGraphPriority(keyboardListener, this);
```

移动是通过 moveBy 动画来实现，通过记录的按键来确定移动方向，移动时需要防止出界。

发射子弹就是新建精灵，并使其向上运动，通过遍历判断 y 坐标位置来移除飞过上边界的子弹。

定义了两个自定义事件，用于检测子弹是否打中陨石，和飞船是否碰到了陨石，方法都是通过遍历计算距离，如果距离小于 25，就播放动画，并执行相应操作。通过 lambda 表达式来设置回调函数。移出子弹和陨石的话，在回调函数中调用节点的 removeFromParentAndCleanup 和容器的 erase 函数。

```
// 分发自定义事件
```

```
EventCustom e("meet");  
_eventDispatcher->dispatchEvent(&e);  
EventCustom e1("meet1");  
_eventDispatcher->dispatchEvent(&e1);
```

播放音效使用 SimpleAudioEngine 对应的函数即可。

wee15

本周主要是学习物理引擎的使用，setJoint 是将球和板用关节连起来，使用了 PhysicsJointPin

类型，参数的后两个为锚点的位置，然后设置球和板的刚体属性

```
physicsBody = PhysicsBody::createCircle(ball->getContentSize().width/2);  
physicsBody->setMass(100);  
physicsBody->getShape(0)->setRestitution(1);  
physicsBody->setDynamic(true);  
physicsBody->setGravityEnable(false);  
physicsBody->setVelocity(Vec2(0, 200));  
physicsBody->setCategoryBitmask(0xffffffff);  
physicsBody->setCollisionBitmask(0xffffffff);  
physicsBody->setContactTestBitmask(0xffffffff);  
ball->setPhysicsBody(physicsBody);
```

然后设置砖块及其刚体，方法类似，位置可以通过计算获得，不过这里位置要设置得比砖块稍

微大一点，不然砖块之间会发生碰撞。然后设置按键动作，在按下时给板一个初速度，在松开

按键时再将初速度设为 0，不过需要判断板是否出界，由于为了避免板和球碰撞时板被撞开，

我把板设为静态刚体，这样 cocos 就不会检测板和边界的碰撞，我自定义了一个 update 函数

来检测板的位置。

蓄力的话，定义一个变量保存力的大小，在按下时清零并调用 this->scheduleUpdate();然后

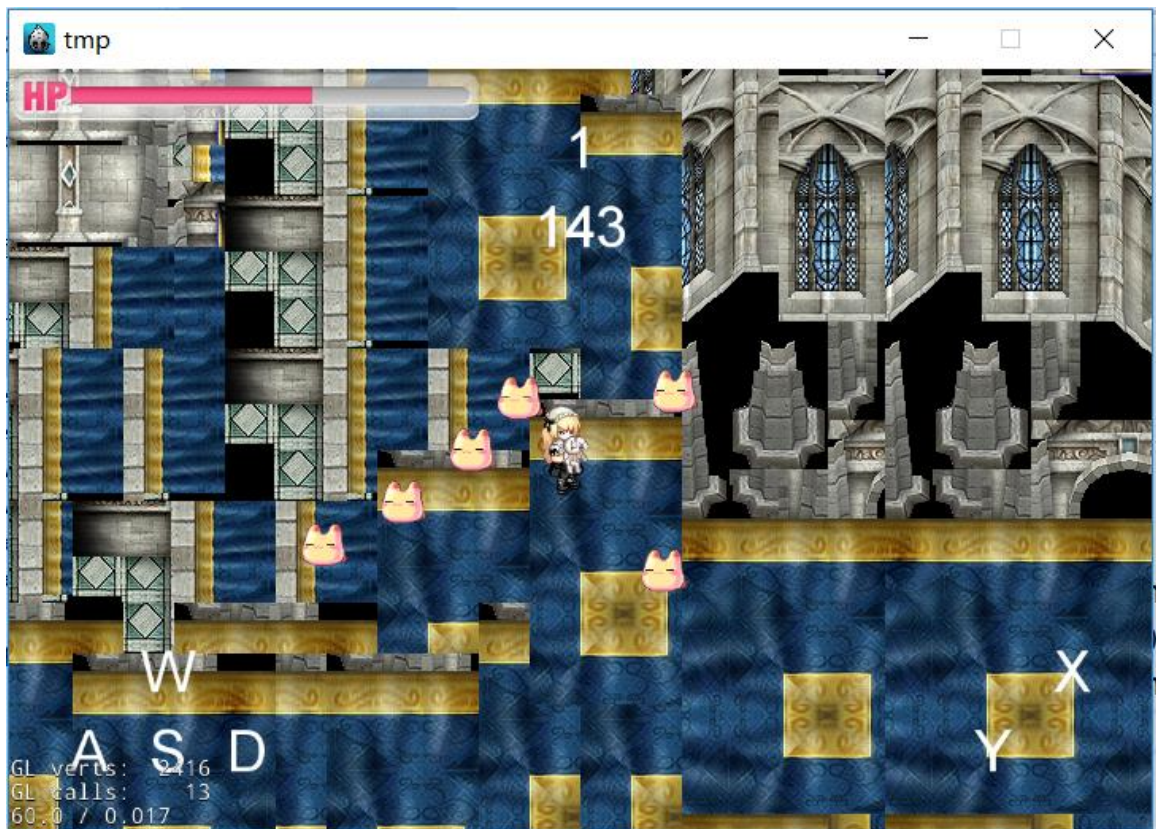
在 update 函数中增加力的值，在释放按键时，给球一个推进力。

```
ball->getPhysicsBody()->applyImpulse(Vec2(0, spFactor));
```

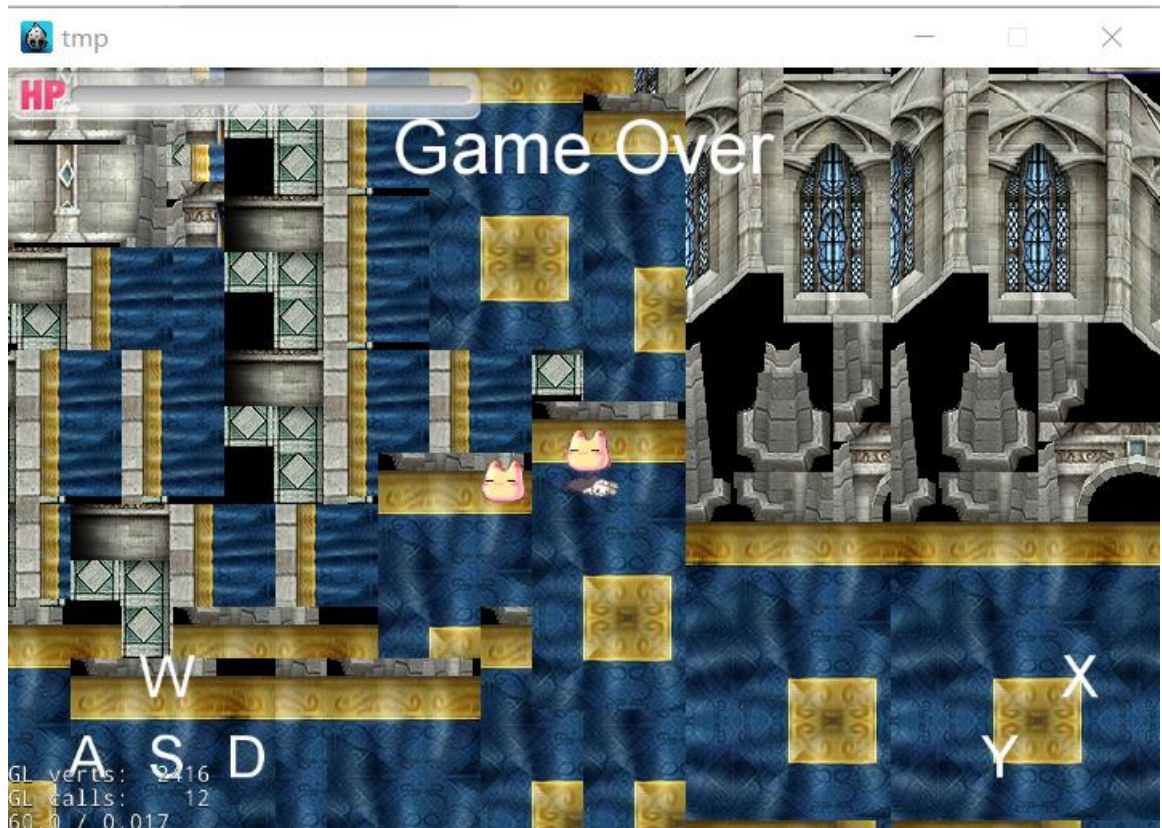
在碰撞检测函数 onContactBegin 判断碰撞体的 tag，如果是砖块的话，就移除，如果是 ship 的话，就调用 gameOver 函数。

### 三、关键步骤截图

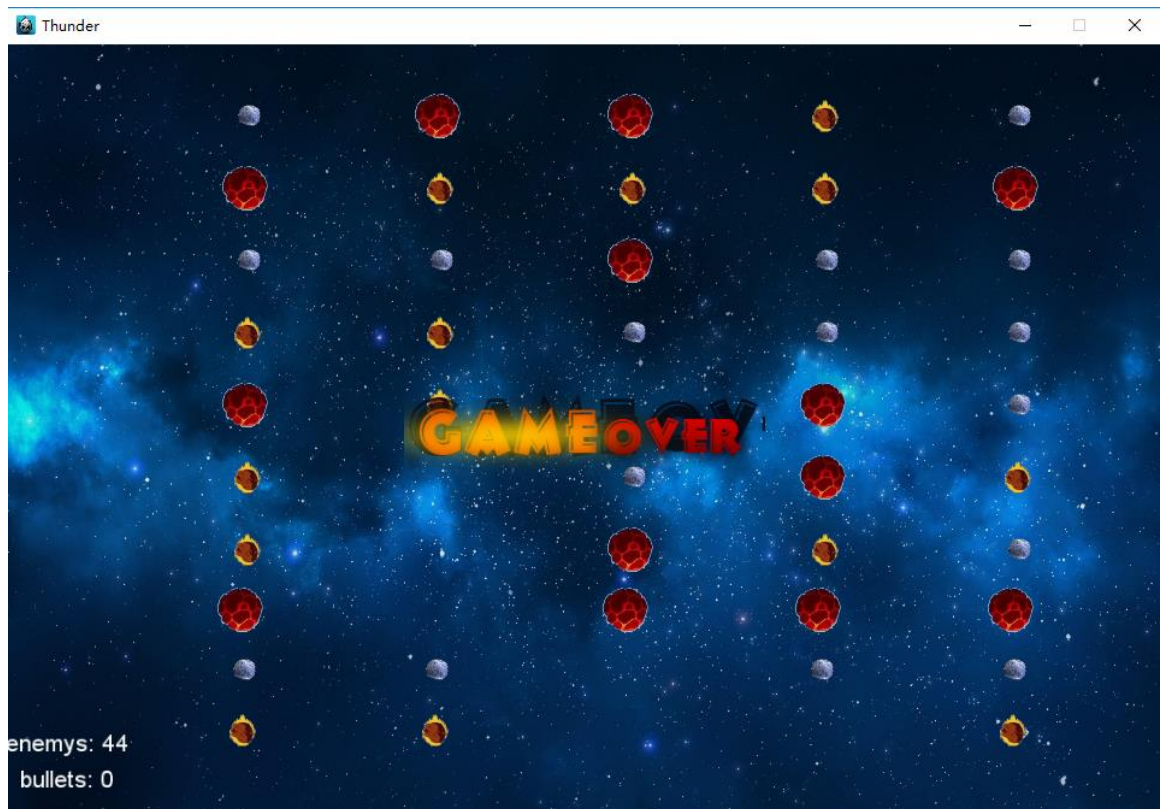
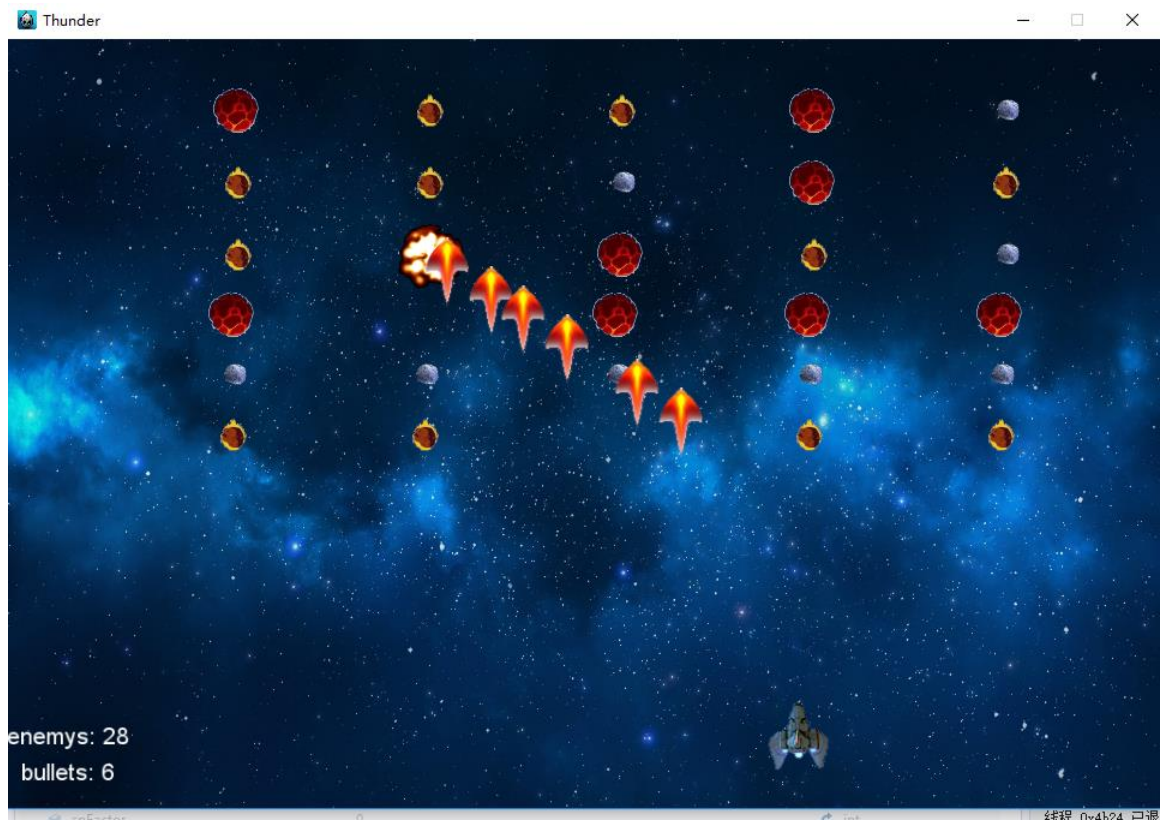
week13



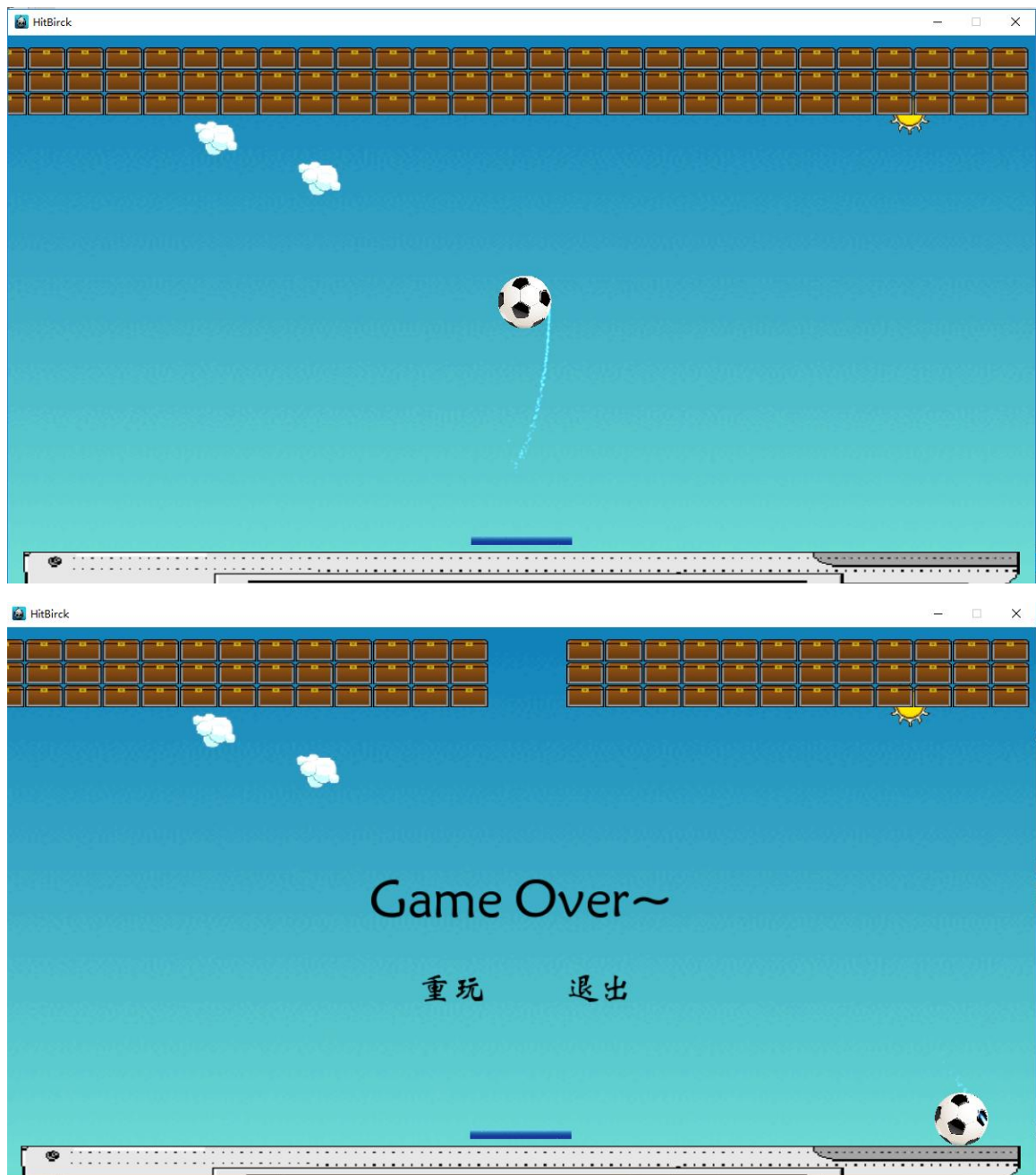




week14



week15



#### 四、亮点与改进（可选）

week13



使用本地数据存储，记录打到的怪物数量，同时在游戏中显示打倒数量。

使用 UserDefaults 存储数量，并在需要的时候通过关键字获取

```
UserDefaults::getInstance()->setIntegerForKey("score", sc);
```

int 转化为 string 并显示在 label 上

```
auto temp1 = CCString::createWithFormat("%d", sc);
```

```
score = Label::createWithTTF(temp1->getCString(), "fonts/arial.ttf", 36);
```

week14

1. 利用触摸事件实现飞船移动。(点击飞船后拖动鼠标)

通过 touch->getDelta() 可以获得表示鼠标移动方向和距离的向量，新建 moveBy 动画

让 player 运行即可

2. 陨石向下移动并生成新的一行陨石

随机加载陨石的图片，并把新建的精灵放在合适的位置即可，要同时添加到 enemys 容器中。

3. 子弹和陨石的数量显示正确

只要确保在增删子弹和陨石的时候，同时从 enemy 和 bullets 容器中操作即可。

week15

使用了粒子系统，并作为 ball 的子节点，跟随球移动。

## 五、遇到的问题



week13 在将打倒怪物的数量保存到本地时，需要考虑更新和同步的问题，还有就是血量变化完之后才 gameover，需要考虑先后顺序。

week14 上课没有仔细听 TA 的要求，检查的时候 TA 告诉我在陨石进过下边界的时候，游戏也要结束，我做的是陨石会消失，并同步数量更改，现在已经按要求改正，其实比较容易，因为我已经做了检测，只需要把判断成功后的动作换成游戏结束的操作即可。

week15 球和板发生碰撞后，板会移动，设置回弹系数行不通，最后只好把板设为静态，并手动检测板是否出界。

## 六、思考与总结

通过这几周的作业，学习到了 cocos2d-x 开发的很多重要方面，包括事件处理，音效的控制，地图的制作，本地的存储和物理引擎的使用，基本涵盖了游戏开发的基础内容，在学习过程中也遇到了很多问题，感觉 cocos2d-x 的网上资料和文档还是偏少，给学习带来了一定困难，但还是感受到 cocos2d-x 相对于 Unity 在构建游戏上的快速和简洁，不过与此同时在结构性和完整性上还是要弱一点，应该说各有优劣吧。前一周作业的完成和 TA 的要求略有不同，感受到了上课仔细听 TA 讲的重要性。