**现代操作系统应用开发实验报告**

姓名：陈衍斌

学号：16340042

实验名称：Lab4

1. 参考资料

<https://github.com/gfzheng/MOSAD>

<http://blog.csdn.net/zhanghefu/article/details/38466801>

<https://blog.csdn.net/tsbyj/article/details/47083397>

<http://www.cocos.com/docs/>

1. 实验步骤

Week13:

在上次作业基础上，加上怪物，实现以下功能：

1. 随机产生怪物并且怪物会向角色靠近

2. 怪物碰到角色后，角色掉血，角色血量为空则播放死亡动画并解除所有事件

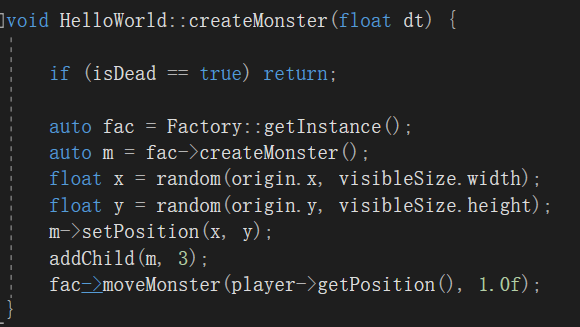
3. 角色可以攻击怪物

4. 使用tilemap创建地图

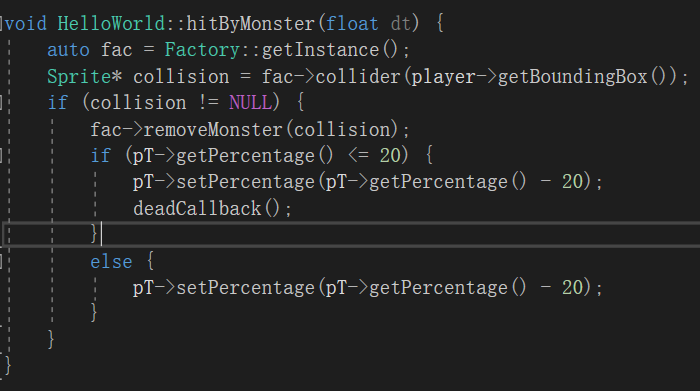
5. 加分项：使用本地数据存储，记录打到的怪物数量，同时在游戏中显示打倒数量

怪物的基本行为TA给的代码已经基本实现了，我们需要做的就是使用它们。

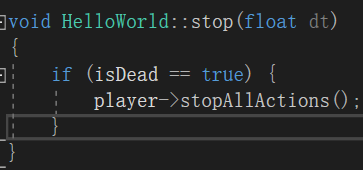
首先是随机产生怪物，ppt上已经给了代码，我们只需要直接拷贝进去并添加向角色靠近的行为：



然后是被怪物攻击，角色会掉血，血量为0时调用deadCallback()方法播放死亡动画：



接触所有事件用调度器实现，代码如下：

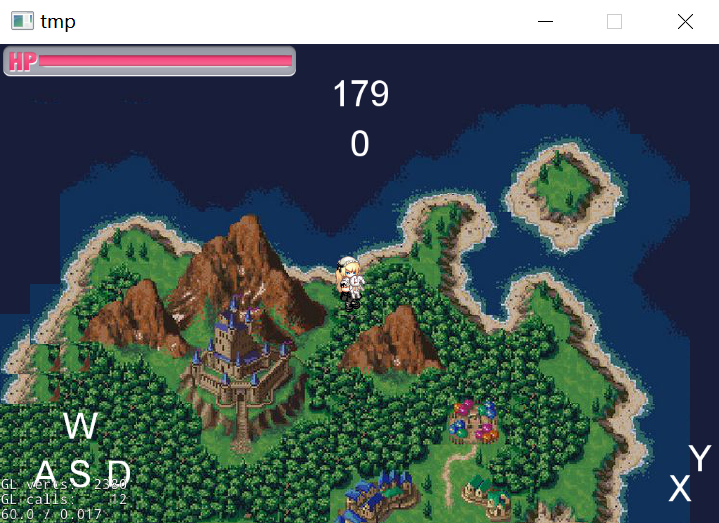


角色攻击使用碰撞检测，在ppt上基本思路已经给出来了，这里使用attackMonster()函数实现。在控制Y键的方法action中调用函数，如果函数判断有怪物，那么会返回true，score增加，怪物消灭，角色回血；如果没有则返回false。

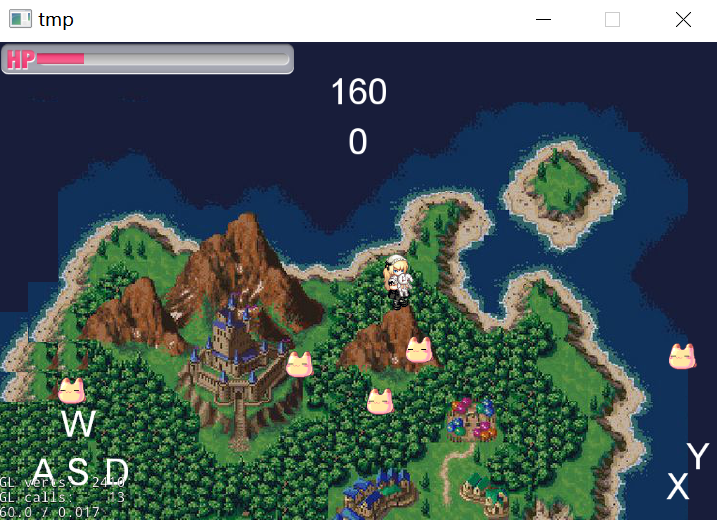


然后是使用tilemap创建地图。这个代码ppt上已经完全给出来了，就不贴了。这个主要是一些与代码本身无关的问题，后面在“遇到的问题里“会讲。

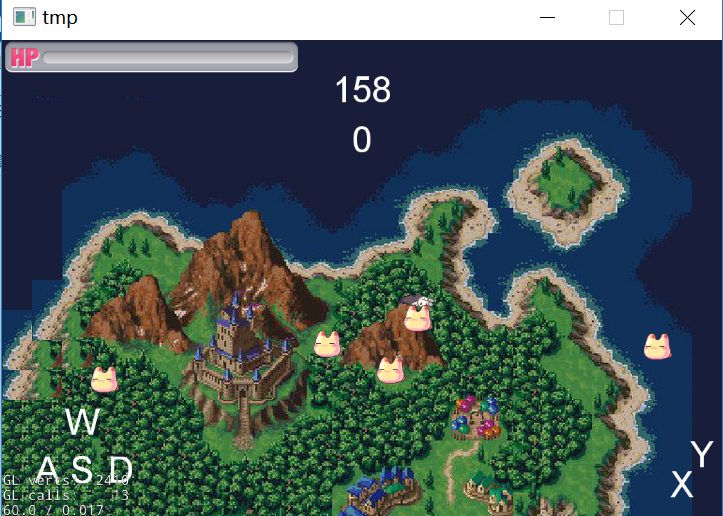
运行效果：初始界面如下：



然后每隔2秒会产生怪物，攻击命中怪物会回血，被怪物接触会扣血：



血量为0后计时器停止，所有按键功能都会被禁用：



Week14：

要求 ：

利用键盘事件实现飞船左右移动。

利用键盘和触摸事件实现子弹发射。

用自定义事件实现：子弹和陨石相距小于一定距离时，陨石爆炸，子弹消失。

游戏过程中有背景音乐，发射子弹、击中陨石有音效。

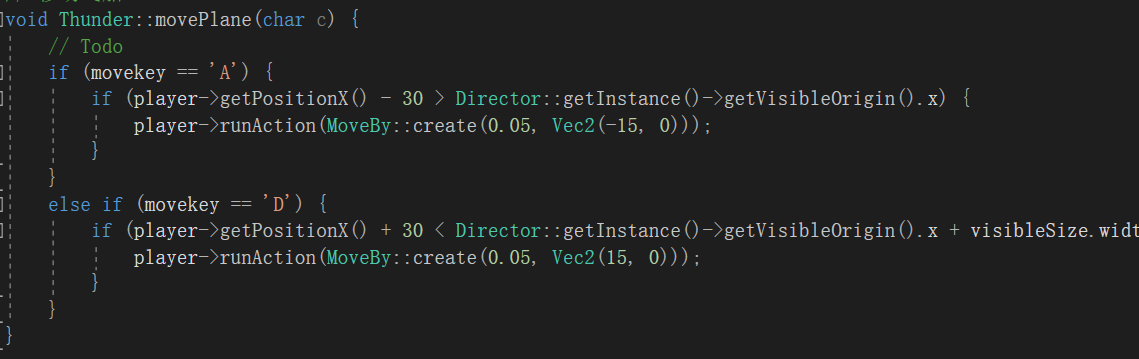
注意飞船、子弹的移动范围。

游戏结束飞船爆炸，移除所有监听器。

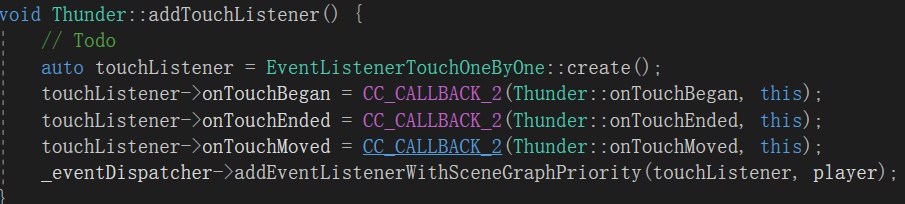
加分项 ：利用触摸事件实现飞船移动。（点击飞船后拖动鼠标）。陨石向下移动并生成新的一行陨石。子弹和陨石的数量显示正确。

基本的框架TA已经提供了代码，我们需要做的就是将各个函数的功能实现。

首先是飞船的左右移动，通过movekey来判断键盘指令，然后加上边界判断，最后移动：



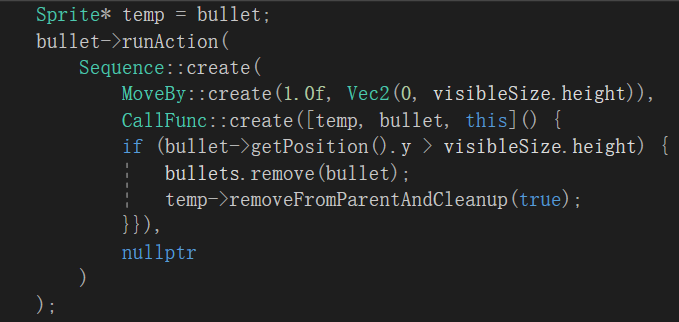
通过键盘和触摸事件让子弹发射：首先添加触摸事件监听器：



键盘事件监听器与触摸事件大同小异，这里不多累述。

然后调用fire()就可以发射。不过这里需要先判断一下点击位置，如果离飞船的位置太远的话则视为无效点击，就不发射。

fire函数我们只需要实现移除飞出屏幕外的子弹。基本的思路ppt上也给了，利用回调函数实现即可：

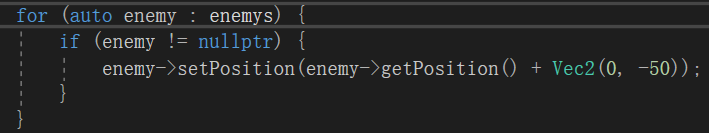


判断子弹是否大众陨石并执行对应操作：ppt上已经给出课核心代码，先判断两者之间的距离，如果距离很近就发生碰撞，然后将子弹和石头移除并播放爆炸音效：

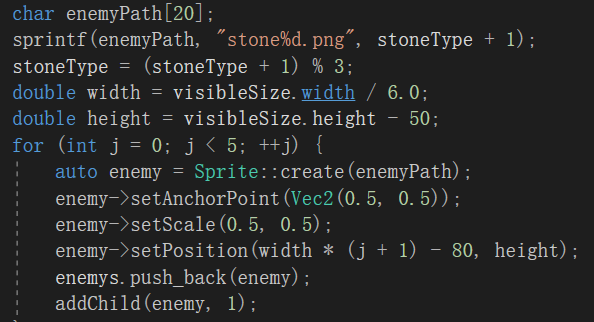


显示子弹和陨石的数量：update函数的代码TA已经全部给出来了，所以我们就不需要管了。

陨石向下移动并生成新的一行陨石：其实就是实现newEnemy函数。首先将现有的陨石全部位置都向下移动一行：



剩下部分的代码在addEnemy函数中已经提供了一个大体的思路，我们直接套用这个模板，稍微修改一下即可：



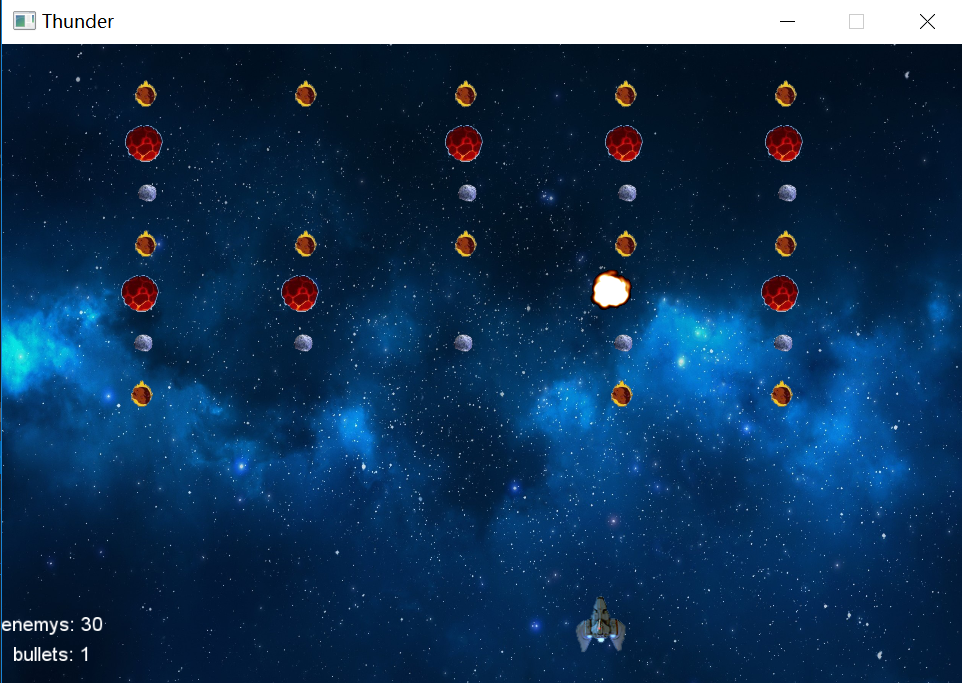
最后是游戏结束飞船爆炸并移除所有监听器：与“子弹与陨石碰撞”的处理方法差不多，直接将代码添加到meet函数里即可，没什么新的东西，这里就不贴代码了。

Gameover后记得把调度器停止，监听器移除：

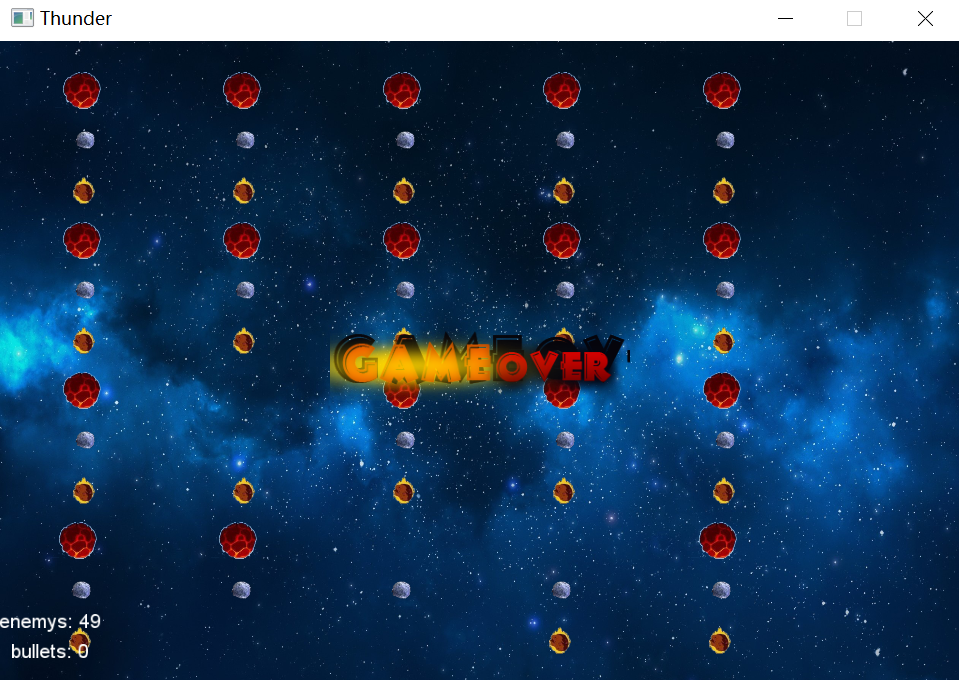


然后是运行结果截图：这次作业不太好用截图来反映做的效果，所以指截一些关键的图。

首先是碰撞的效果图，同时可以看到，下面的陨石和子弹的数量显示也是正确的：



然后是游戏结束时的效果图。这时飞船已经爆炸消失，监听器也移除：



飞船的键盘移动，触摸事件以及背景音乐之类的东西，这里无法截图，TA可以直接运行代码跑一跑即可。

Week15：

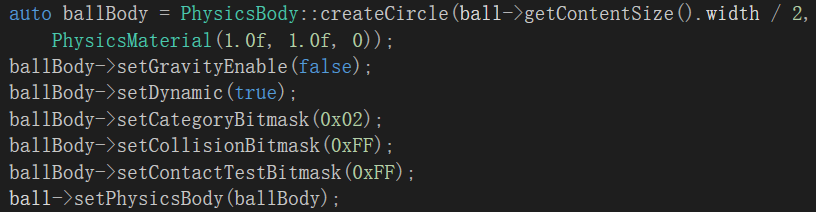
基本要求 ：难度逐渐加大

* 控制板子左右移动
* 在顶部生成小砖块
* 使用关节固定球与板子
* 为板子、球、以及砖块设置物理属性
* 砖、球碰撞则消去砖头，球与地板碰撞则游戏结束

加分项：至少使用一种粒子效果（加在球上面，etc）

首先我们需要给我们需要用到的每个精灵创建刚体，这样每个精灵就有对应的物理属性。其中PhysicMaterial有三个参数，分别为密度，弹性系数，摩擦系数。每个刚体有CategoryBitmask、CollisionBitmask、ContactTestBitmask属性，主要用来控制所需要监听的碰撞事件，具体监听过程ppt上面也已经说明了，这里就不叙述了。

每个刚体的创建过程只有一些细微的差别，这里就放创建球的代码：

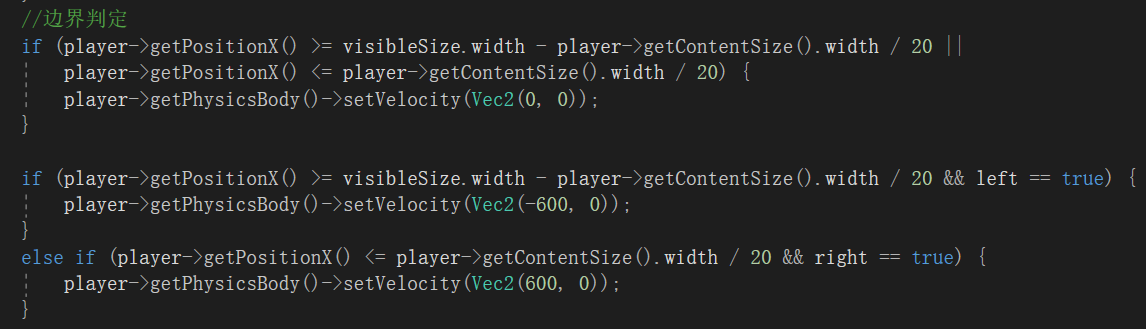


按照步骤，我们先实现板子的左右移动，这次通过给板子刚体一个初速度来实现，代码如下：



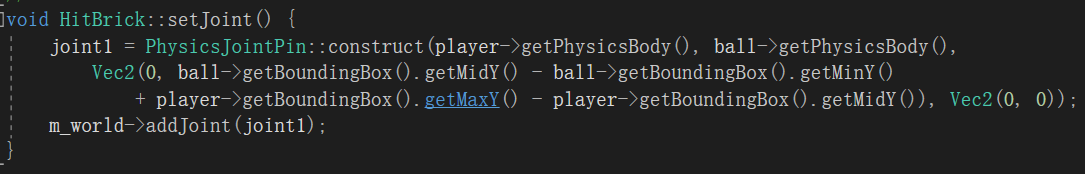
其中left和right用来表明板子是否能够向左/右移动。

与上次作业类似，我们一样需要处理边界问题，我们将下列代码加入update函数里：



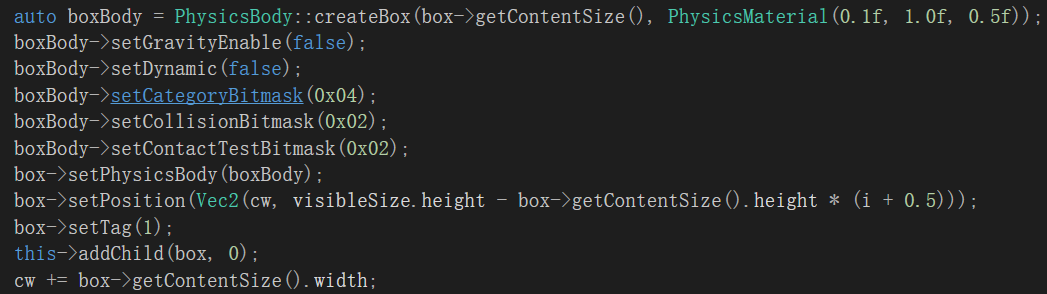
第一个if用来判断板子是否到达边界，如果在边界，则将板子速度置为0，限制板子的移动。然后通过第二个if来判断板子是在左边缘还是右边缘。如果到达左边缘，那么就不会限制向右的移动。

然后使用关节固定球和板子：



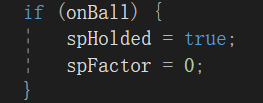
这里需要注意：如果用ball->getposition()来获取定位点的话，球与板子始终会有一段距离，这里需要手动设置将锚点设在球的圆心上。

接下来在顶部生成小砖块，通过for循环暴力创建即可。For循环提供的代码已经帮我们构建好了，我们只需要创建每个小砖块的刚体即可：

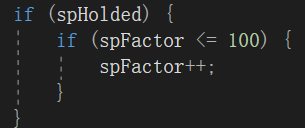


这里调整小砖块的位置有点小麻烦，不过多试几次就可以找到合适的位置。

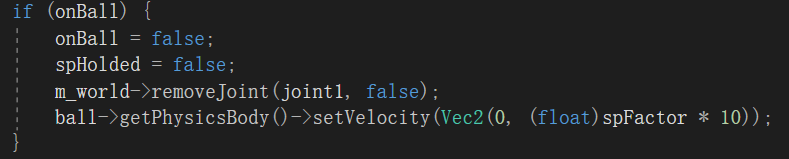
接下来实现空格发射球，这里没什么，记得标记球为蓄力状态：



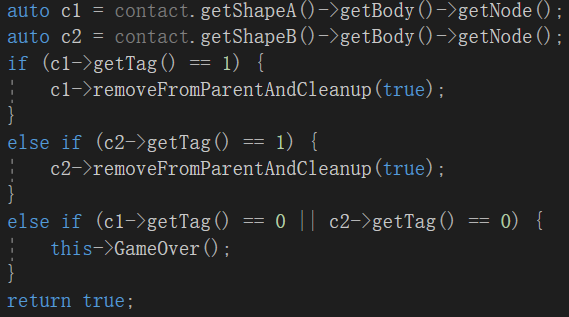
蓄力可以使球的速度变快（也就是ppt上的难度加大）,这里限制了一下最大蓄力数量：



蓄力结束后发射球：



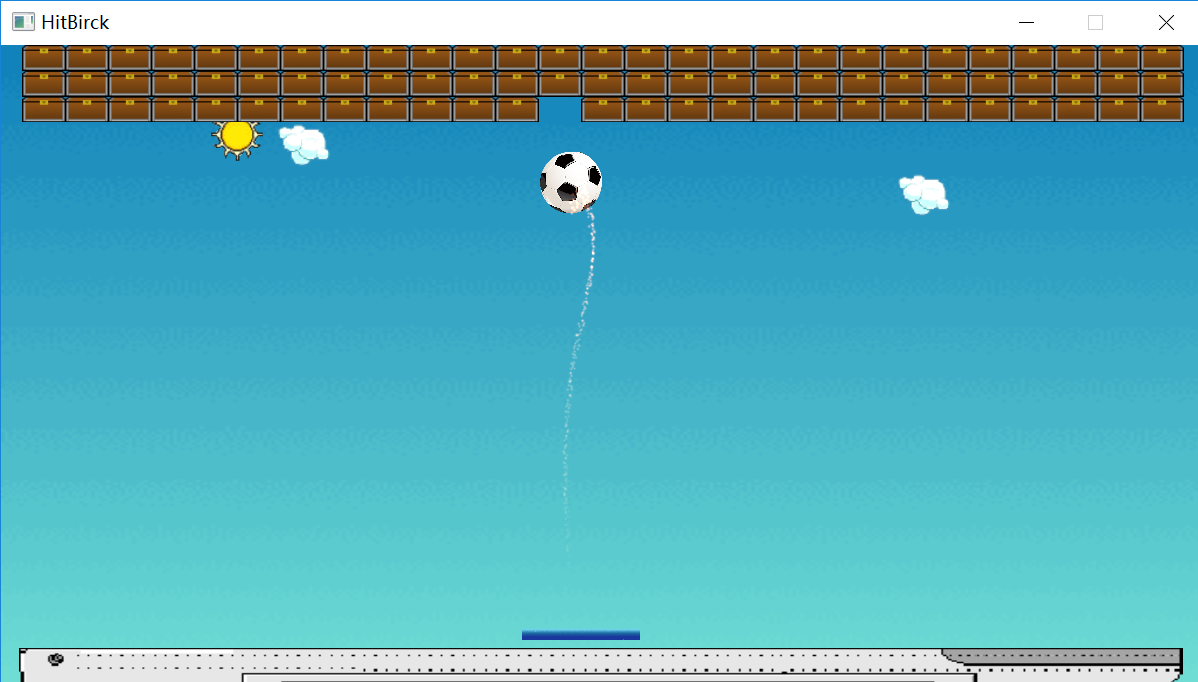
然后就是实现各个碰撞事件。这里通过之前设置的标签来判断碰撞的事物是什么，然后执行相应的事件：



效果，首先使开始界面：



在球碰到砖块后，砖块消失，球反弹回来：



当球落到地面时，游戏结束：



其他的像边界限制，板子的左右移动等等，截图不好展示出来，TA可以直接运行代码来查看。

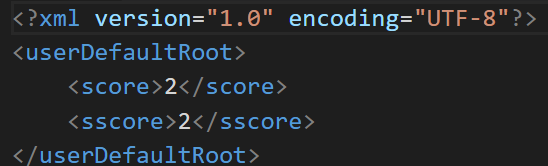
1. 亮点与改进

Week13：本地数据存储：将打倒的怪物数量存储到本地文件UserDefault.xml中：

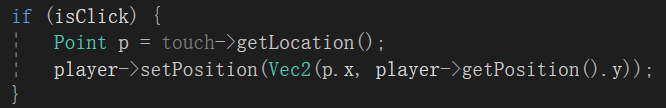


其中UserDefault.xml的文件路径为C:\Users\lcccy\AppData\Local\项目名。

打开文件，可以看到记录了上次的打怪数量：

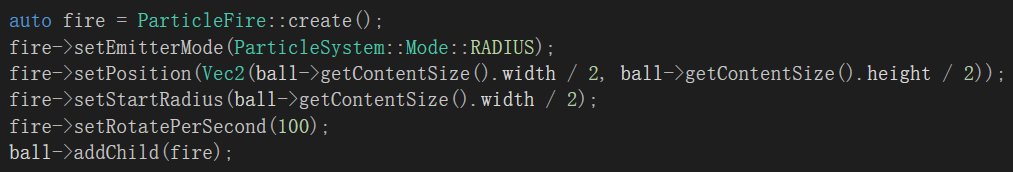


Week14：加分项大部分在实验步骤里说了（而且有些加分项必须做，比如陨石向下移动。如果不做的话，游戏永远不会结束）。这里放一下利用触摸事件实现飞船移动的代码。其实也就几行代码的事：



键盘移动进行了边界处理（飞船不会移动到屏幕外），触摸事件就没有处理了（懒）。

Week15：使用粒子效果。ppt上写着可以在球上加入粒子效果，那我就加在球上了。这里通过半径模式的粒子发射器来实现火焰粒子效果，cocox2d-x自带这种效果，稍微设置一下参数就可以使用：



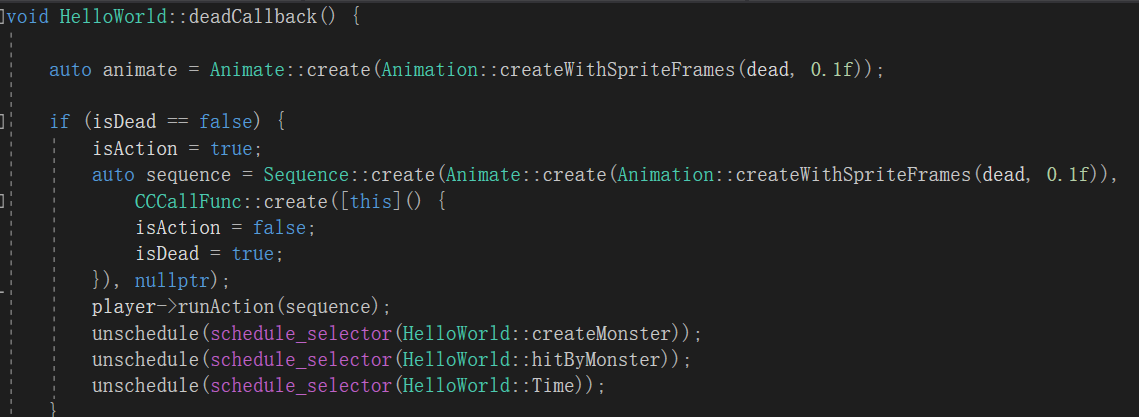
然后球会有粒子轨迹：



1. 遇到的问题

Week13：首先是tilemap的创建问题。最主要的问题是无法显示图片。网上的说法各种各样，有人说png图片不能用，有人说bmp图片不能用，然而换成jpg依然不能显示图片。还有人说tmx要写成绝对路径，试了一下，不仅加载不出来还会崩溃。后来在网上找了一种可行的方案：先把所有要用到的资源放到Resources中，然后利用资源创建新地图，保存到Resources中，然后导入图块，也必须用Resources文件夹中的图片，这样子图片可以显示。

然后是解除所有事件，直接用stopAllActions()方法可以实现。但是这样有个问题：解除事件的行为会覆盖掉播放死亡动画，导致死亡动画没有播放。所以这里使用回调函数来解决这个问题：



然后又产生了一个新的问题：死亡动画播放后，角色自动回到了初始状态而没有停在死亡动画中。问了其他同学，都没有这个问题，参考网上用setRestoreOriginalFrame来锁定动画最后一帧，好像也没有效果，结果现在也没解决。怀疑是代码框架出了问题，但是又不想重构代码，就索性没管了。

Week14：主要的问题在创建新陨石那里。主要是考虑如何按顺序读取图片来创建陨石。最后通过添加一个变量stoneType来实现。1，2，3分别代表不同的陨石精灵，每次创建陨石stoneType就会改变（1，2，3循环），这样就可以创建不同的陨石：



Week15：刚体的属性设计是个比较麻烦的点，这个在实现之前提前规划好比较方便。

然后就是关节固定板子那里，用ball->getposition()定位调节一直不行，最后还是参照了其他同学的方法才搞定。

1. 思考与总结

这次的实验作业对我来说非！常！困！难！完全零基础的情况下，通过自己摸索来实现功能

Week13的作业直接在之前的“伪崩坏学园”的基础上修改即可。TA提供了怪物的代码，这也大大减轻了我们的负担，而且也不用调各个精灵的位置，还是比较简单的（虽然有个bug没解决）。

Week14：TA已经给了实现的方法以及各个私有成员，整体框架已经构建好。我们主要是学习事件分发和响应，通过监听器，分发器来实现一些交互功能。同时也学习了如何添加背景音乐和音效。

Week15：这次作业主要是学习使用物理引擎和粒子系统。TA同样给出了方法和私有变量成员，我们只需要根据思路实现。同时，也复习了监听器的使用。

总而言之，这三次作业TA都已经提供了整体的代码框架，所以实现起来并不是很难。但是在实际设计中，整体的实现思路和框架是需要我们自己设计的，像期末项目就需要我们自己设计需要实现的功能。这么一想，我可能现在就要先规划一下期末项目的设计了。