# 实验目的

学习视频，完成2048游戏，积累实战经验。

# 实验内容

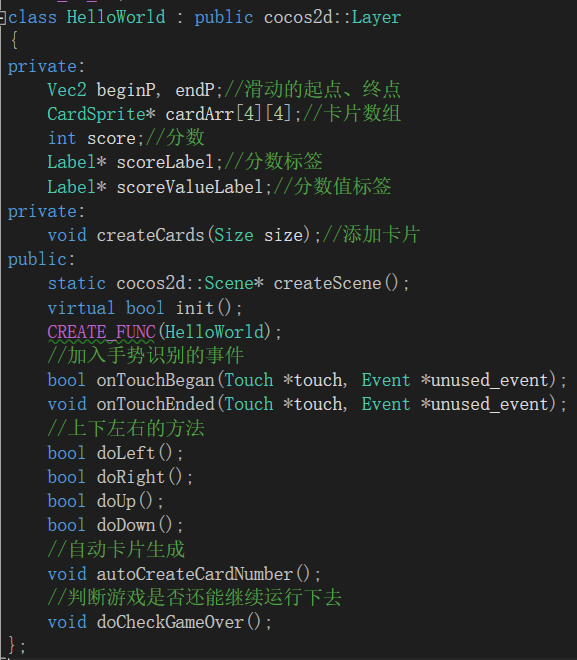
完成2048益智游戏项目，要求上交实验报告（包括核心代码截图）和项目（项目上交两个压缩包，1个是debug文件夹，另一个是包括resources和classes的文件夹的压缩包），两个压缩包均在压缩前命名为学号+姓名。务必测试好debug文件夹内的项目可以脱机运行。

# 实验步骤

视频中有许多不合理的地方，所以我对其部分代码做了修改。**主要修改的地方已经用红色字体标红。**

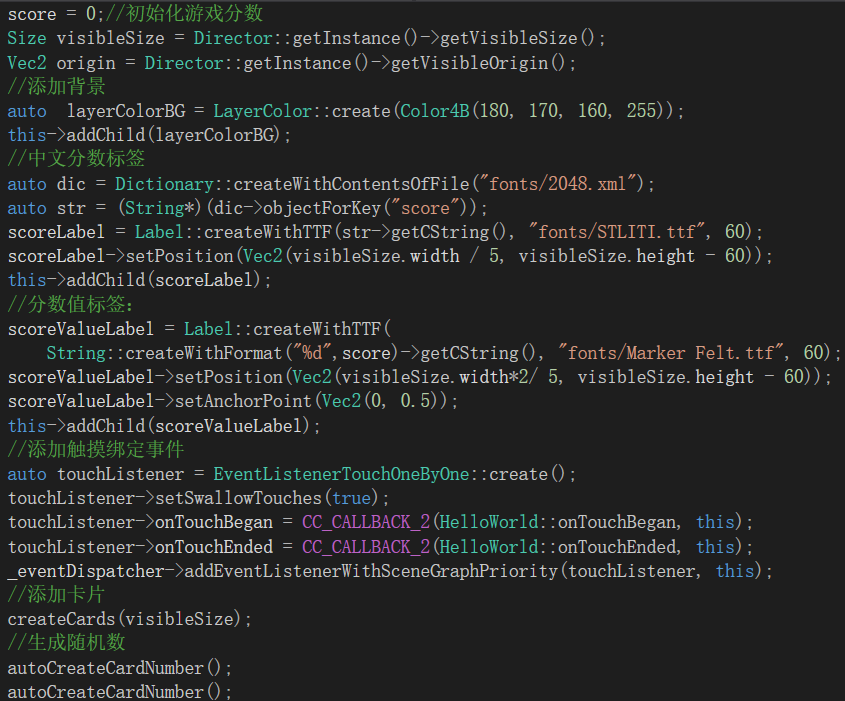
## 游戏场景和卡片

### HelloWorld.h



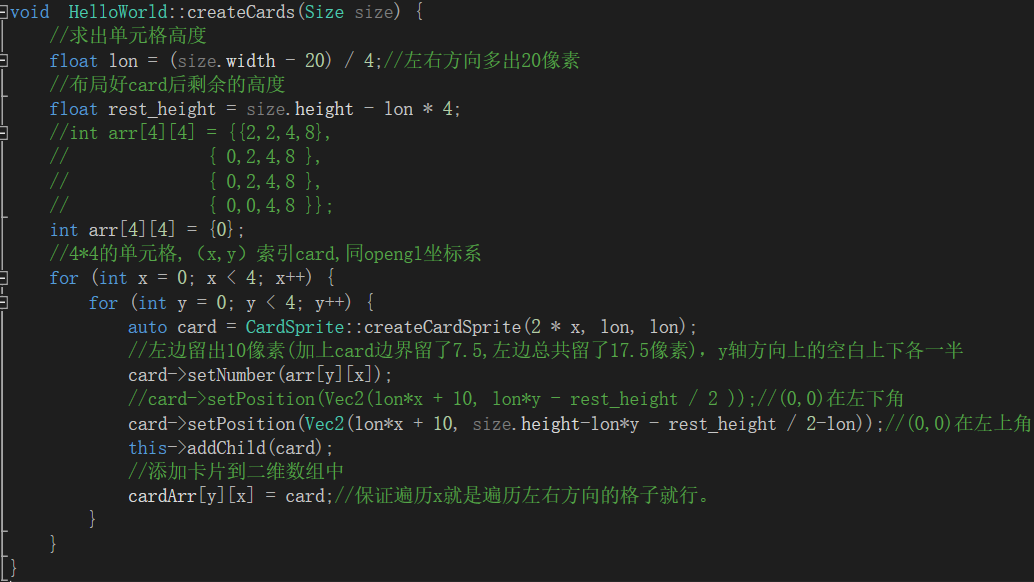
### init函数

添加了背景、标签，进行了触摸事件的绑定，后面调用了createCards、autoCreateCardNumber函数。



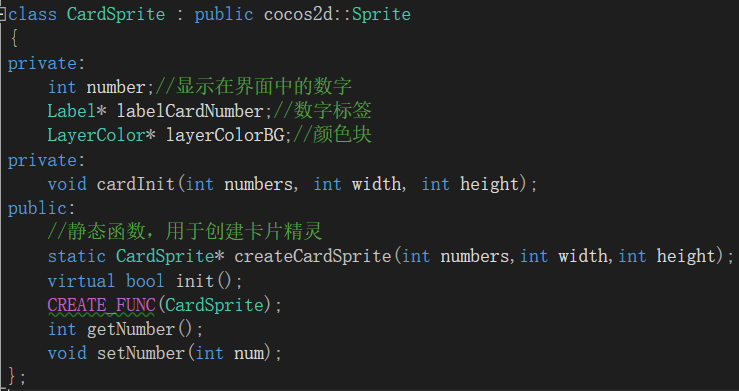
### createCards函数

调用了CardSprite类的createCardSprite函数。

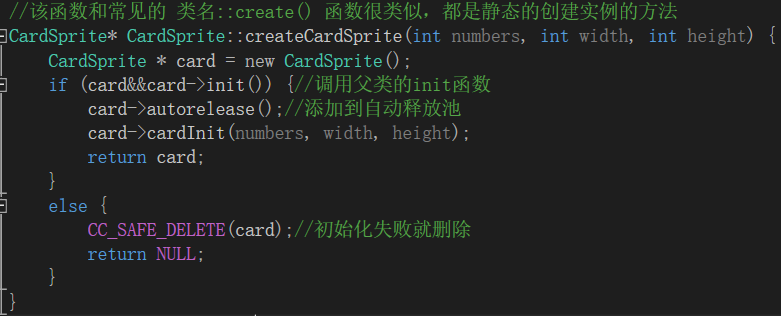


### CardSprite类

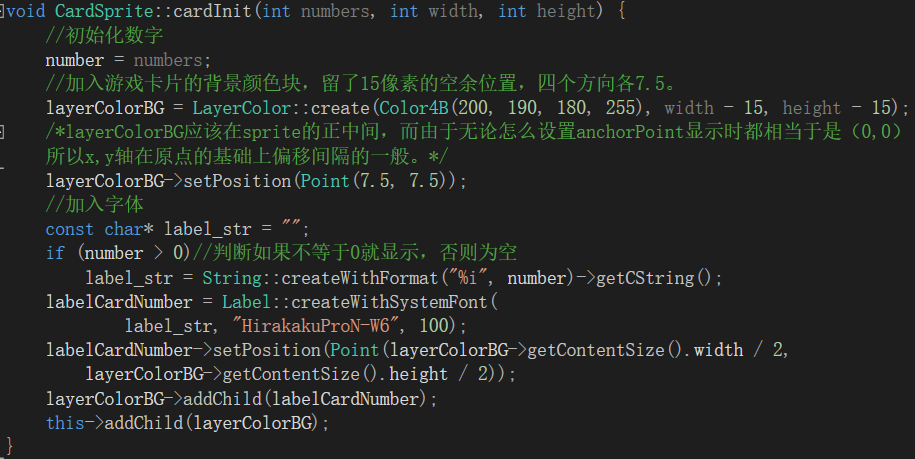
#### CardSprite.h



#### createCardSprite函数

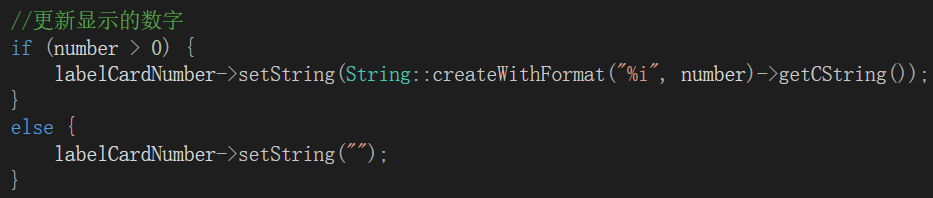


#### cardInit函数

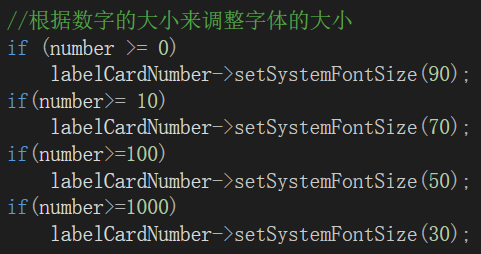


#### setNumber函数

更新卡片显示的数字



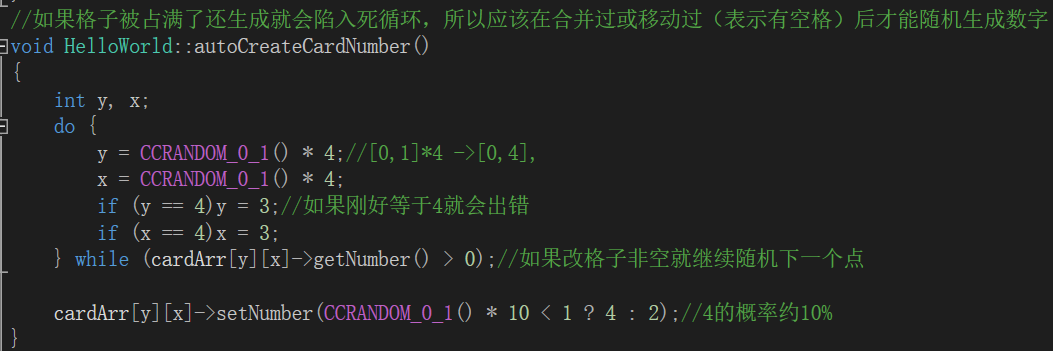
更新数字大小



更新卡片colorLayer的颜色



### autoCreateCardNumber函数

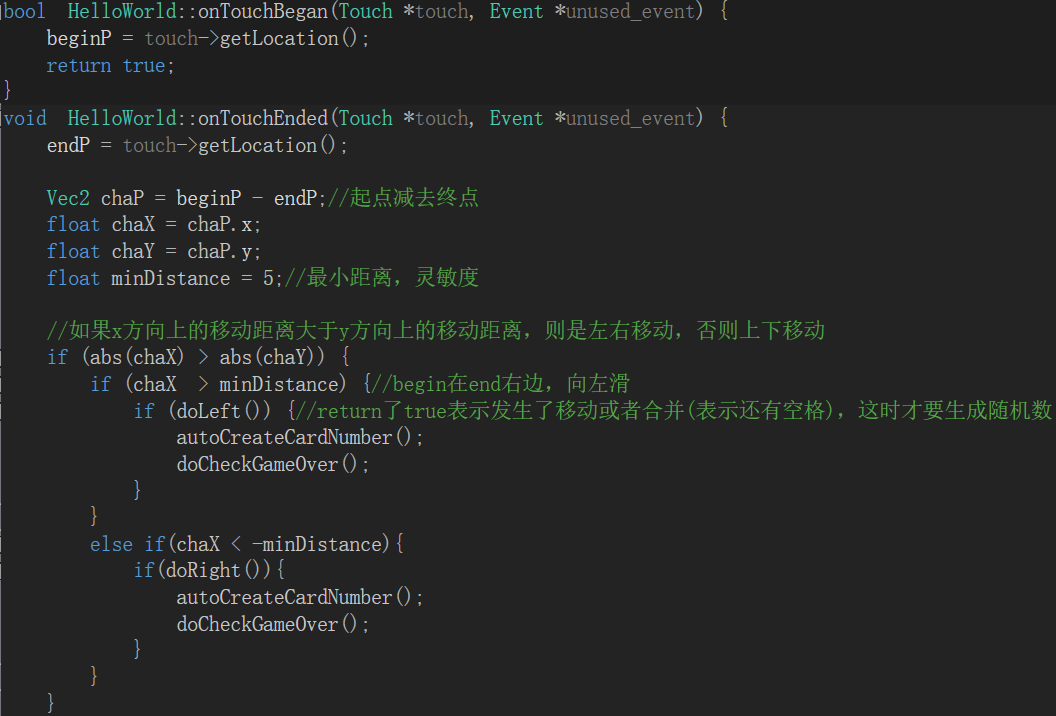


## 游戏逻辑相关函数

### 触摸回调函数

修正了灵敏度，只有当滑动像素大于5时操作才会起效，视频中的代码有一个问题，只要一点击没有滑动就会发生卡片doLeft等函数的调用。Y轴方向和X轴方向类似。

滑动后进行了合并或移动（表示还有空格）才要生成随机数。



### doLeft函数（游戏主要逻辑）

（我把游戏逻辑实现成了可以连续合并，实际上一次滑动每个格子最多只能合并一次，把do,while循环去掉即可，上交的代码为已更正版本）

（其他方向类似）

#### 游戏效果

往左划时

要点一:所有的card都会往左移（跳过值为0的card),在左边互相紧靠。

要点二:肯定是左边的先合并，如果有相同的就合并，合并后又会有空card,空card后面的card要填补空card的位置。

#### 样例分析

假如采用**先移动再合并的**逻辑:

**例一：4202**

4202移动->4220：从左往右遍历，如果格子非空且其左边为空，就把格子向左移动。

4220合并->4400：此时所有格子都在左边紧靠，从左往右遍历，有一对相同的就合并->4400,之后可能在左边会出现新的可合并格子对，所以**合并操作需要需要反复扫描行，直到一次扫描中没有出现合并才停止**。

4400合并->8000

**例二：2222**

2222合并->4022移动->4220合并->4400合并->8000, **其实合并之后不需要立马移动，因为后面还可能出现可合并对**

2222合并->4022合并->4040合并->8000

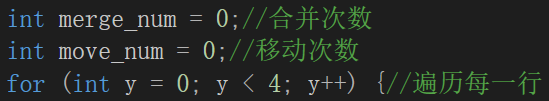
经分析采用**先合并再移动**的逻辑

#### 游戏逻辑

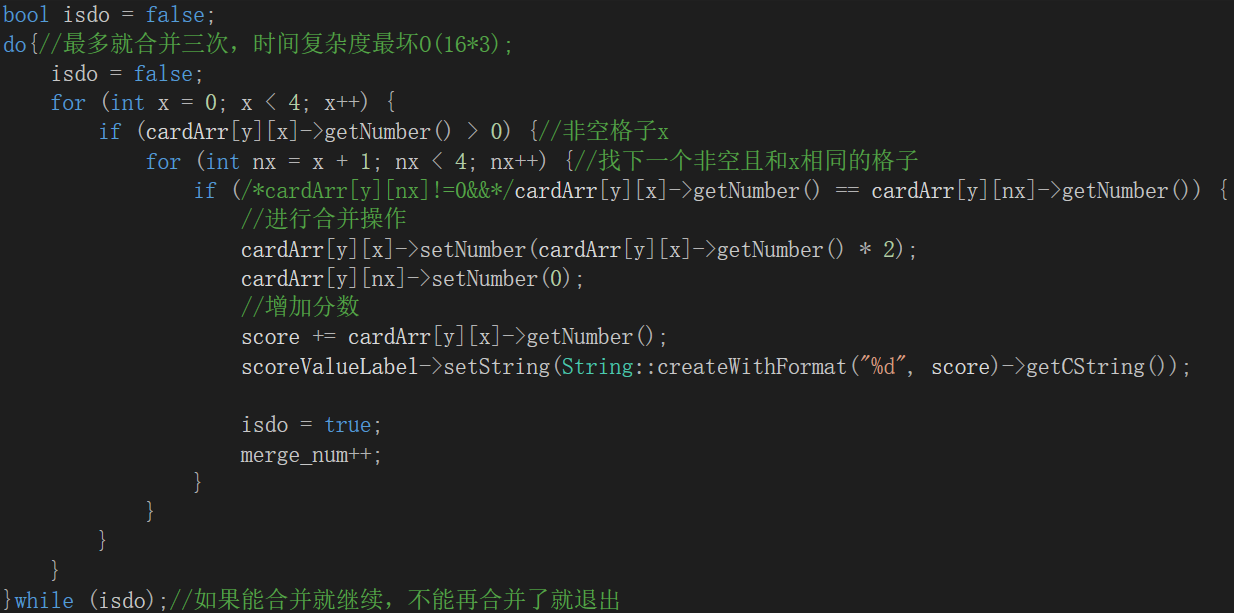
Step1:合并扫描。对每一行反复进行合并扫描，合并时忽略空格，直到该行没有能合并的格子。

Step2:格子移动。对每一行进行一次移动扫描，如果是空格，就从后找一个非空格子移动到它的位置。最后让有数字的卡片都紧靠的左边。

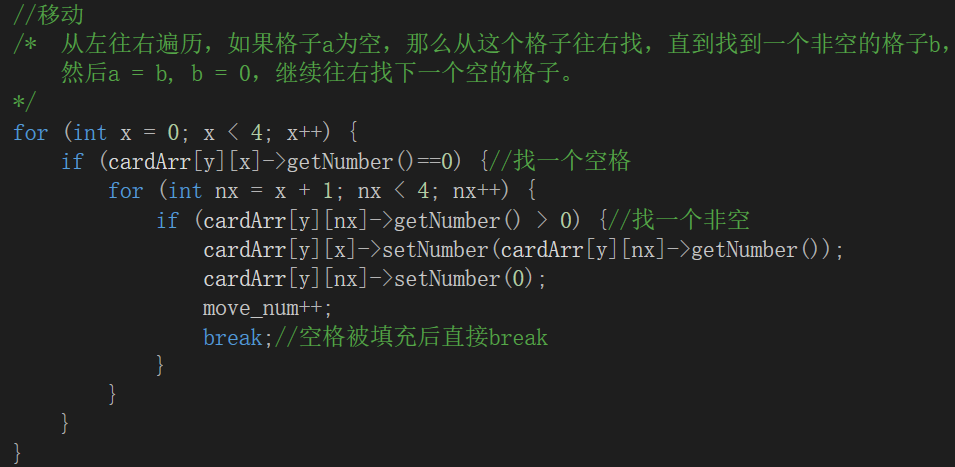
#### 代码实现



合并扫描

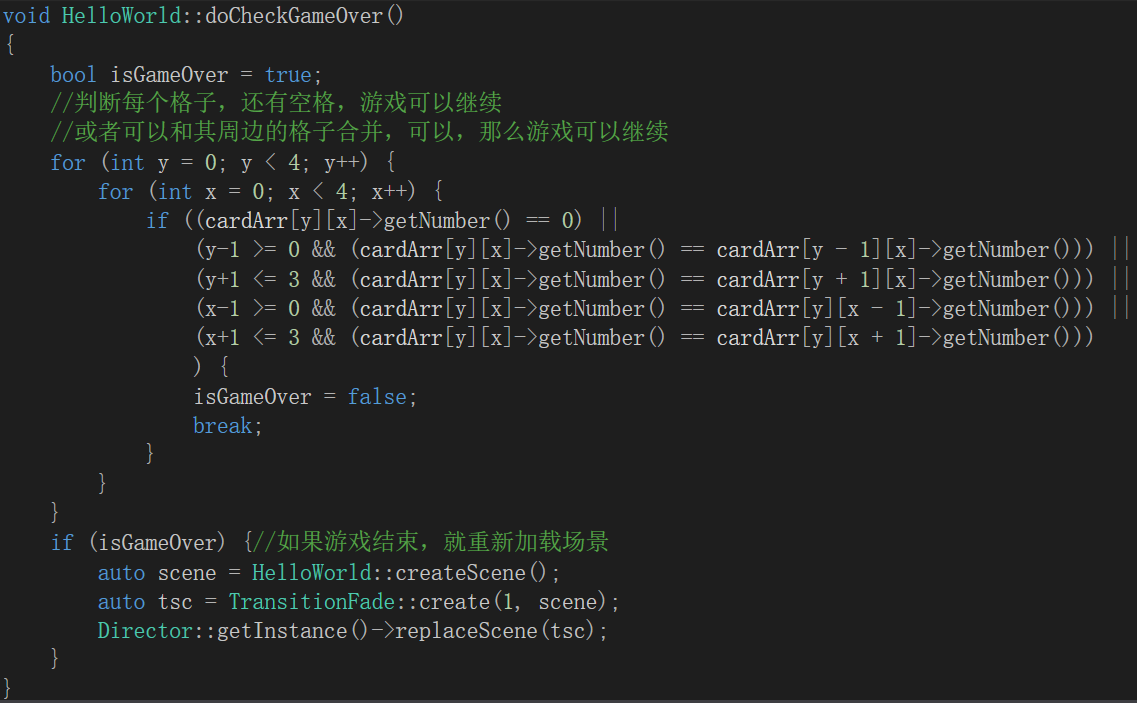


移动扫描





### doCheckGameOver函数



# 实验心得：

## 关于Node的默认锚点

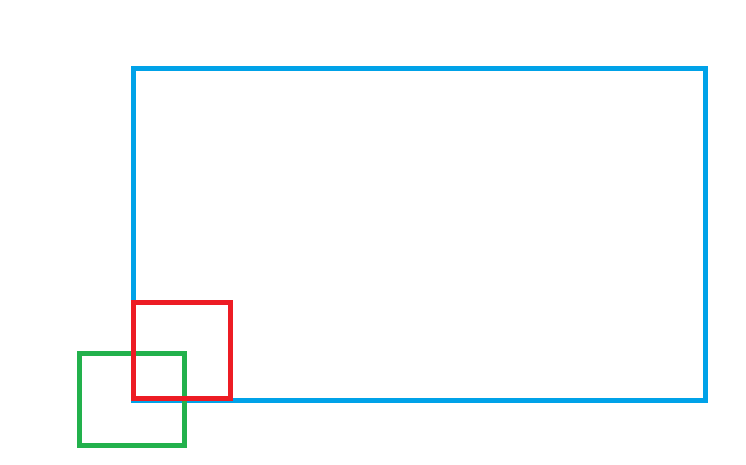
Node的默认锚点为Point(0.5,0.5),即中点。

但是Layer和Scene例外 Layer和Scene的默认锚点是Point(0,0),即左下角，且不能改变

无论父控件移动到哪里，子控件的getPosition和setPosition都是指相对于父控件锚点的位置。

## 在设置卡片及其corlorLayer等子对象位置时遇到一个问题

card位置是（0，0），corlorLayer位置也是（0,0），锚点**打印**显示都为**（0.5,0.5）**。corlorLayer是card的子对象，card是层的子对象，但是最终**显示**时，却相当他们的锚点于是**（0，0）**。把整个精灵完整的显示在了左下角（如下图红色矩形），而不是其右上角的那四分之一部分（如下图绿色矩形），蓝色表示屏幕。



**无论锚点设置成多少，把锚点当成（0,0）就可以得到预期效果。**