**2017软件学院C++期末项目**

**——泡泡堂**

**1. 欢迎界面**

**1.1 开始游戏**

1.1.1 单人训练：一个人在地图中训练

1.1.1.1 操作帮助：指明操作键位

1.1.2 双人对战：两人在同一电脑上用不同键位操作进行对战

1.1.2.1 地图选择：提供三个不同风格的地图以供选择

1.1.2.1 操作帮助：指明操作键位

1.1.3 局域网对战：只实现了聊天功能，输入IP地址后可两人轮流发言进行聊天，一人发言后必须等待另一人发言

**局域网聊天使用方法：（两台笔记本连接同一wifi）**    1.打开局域网聊天室场景：开始游戏->局域网对战（并未真正实现局域网对战）  
    2.进入后，直接输入服务端电脑的ip地址（需要提前决定哪台电脑充当服务端），按下回车结束输入  
    3.商量好作为服务器的电脑点击“创建房间”按钮，客户端电脑点击“加入房间”按钮  
    4.此时，界面右侧显示的是本机收到的消息（服务器和客户端自动发送的“HelloWorld”），界面左侧则用来显示输入时已经输入的内容。  
    5.此时服务器端开始输入（英文），并按回车进行发送。（注意先为“服务端”）  
    6.此时客户端收到消息，在界面右侧显示。客户端再输入信息并发送。  
    7.重复5-6步即可进行简单的局域网聊天  
**注意**：   
    1.由于接收消息的recv()函数在接受到消息之前处于阻断状态，无法进行其他操作，所以该界面并未设置返回按钮。需要关闭后重新打开，才可以进入其他界面。   
    2.由于接收消息的recv()函数是阻断的。所以，发送消息的步骤需要严格按照5-6进行重复（即服务端-客户端依次发送消息），具体原因将在随后写自己的经历中解释。 

1.2 游戏设置：可选择背景音乐和音效的开关

1.3 捐助：可对开发人员进行捐助支持开发

1.4 退出游戏

**2.亮点**

2.1 泡泡能连环引爆

2.2 建筑物与泡泡实现了与人物的碰撞

（**注意：**从逻辑上讲，泡泡应走到正确位置上再放，不应使用在泡泡上的时间进行位置多次调整，因此在我们的游戏中，利用这种技巧将会使人物退出泡泡所在格）

2.3 人物的位移得到修正，使得移动更平滑

**3.那些踩过的坑**

**PS:本次开发使用了更新日志来保存每次开发的变化与问题，以供参考。**

**林磊：**

3.1 人物移动时动画的图片变化会使得人物锚点不停改变，因此不能用锚点进行逻辑判断，因此将人物的有效范围抽象为与瓦片大小等同的正方形，对所有用到人物位置的函数都进行了重构，增设成员，防止了判断的误差和移动到边界时的bug。

3.2 人物移动时的碰撞判断如果只用碰撞点实现会使得操作十分僵硬，移动十分不平滑，碰见凹字型的地图无法方便的进入，如果缩小抽象的碰撞体就会使得人物能在一个格子中移动，这并不符合逻辑，最后使用了检测与补偿位移的方法，使得移动更平滑，经常把人物位置修正到中心线上。

3.3 人物、瓦片的遮盖问题，尝试了调整Z轴、调整创建顺序等方法都没用，原因是地图是一体在一层中……后来发现实现遮盖关系需要把人物与泡泡加入地图中，而我们在很晚才意识到这一点，没有时间做大规模的坐标修正，只得搁置，地图中瓦片的遮挡关系应把障碍物与可炸物放在一个层才有正确的渲染顺序，因为最后才增设地图所以发现时间太晚，也没有进行重构。

3.4变量在不同函数中的传递一定要注意作用域！不然传进去的并不是原来的东西而是新的本地变量……一般方法是在类中增设成员，这样就可以在各处使用同一成员。

3.5 对队伍的能力、工作量错误预估，一开始定的分工、工作与推进表队员们并没有能很好地推进。队伍的沟通与交流十分重要，没有良好的沟通交流就不能开展良好的合作，项目的开展需要正确的自我审视与对他人的了解才能制定正确的计划。

3.6 图片的制作与代码的配合非常重要，虽然我在网上找到了泡泡堂的素材包，但还是需要自己一个个抠图调整大小，制作动画时需要生成plist文件，使用瓦片地图时还需要把散图合称瓦片集导入，遮罩关系也有赖于图片的渲染顺序，这些都需要对图片、代码、计算机工作方式的理解。

**孙浩然：**

遇到过耗费精力的事情等：（这个比较多。。。）   
    整个项目制作工程中，我们参考的书籍是关东升写的《cocos2d-x实战 c++卷》第二版（这本书好贵。。。）我主要做了泡泡类、菜单场景和局域网聊天三大项。其中bubble类是最耗费我心血的一项了。完成的过程也遇到了很多问题，虽然现在看来还有很多地方做的不够好，很多地方还可更加完善，但是由于最后没有时间了，没有对bubble类进行最后的修改优化。下面来说一下，我在泡泡实现过程中，曾经遇到的一些问题。   
   1. 在bubble类里面，最耗费我时间的就是泡泡的动画的实现了：动画的实现耗费了我很多的时间，由于向四个方向爆炸的水花需要同时产生和消失，一开始怎么也想不到怎么让他们同时产生并开始做动画，然后就开始翻书，然而书上也没有给任何相似的例子，最后只能通过搜索引擎才知道了Spawn这个东西解决了这个问题。紧接着我发现CallFunc::create();这个东西居然只能带一个参数！但是泡泡爆炸一共需要两个量：泡泡的威力，泡泡放置的坐标。于是，我在参数方面尝试了很多次，从传递power改为传递Vec2 position，又改回去，来回反反复复，尝试过全局变量，甚至尝试传递过一个Vec3，并把其中的z坐标的值赋值为power，总之尝试了很多很多方法。最后，这些方法都能因为种种原因失败了。最后写了一个包含泡泡基本信息的结构体解决了这个问题。   
    2.泡泡连环爆炸也花费了我的很多的时间：要想实现泡泡的连环爆炸，我就需要一个容器来记录场景中每一个已经放置、但还未爆炸的泡泡的基本信息。这个容器，我最初尝试过queue，因为泡泡爆炸应该是按照放置的先后顺序进行爆炸的。但是由于可能遇到连环爆炸，那么爆炸的顺序就可能不是绝对按照泡泡的放置的顺序进行的，所以我就将queue改为了vector，但是后来我发现，连环爆炸需要erase掉中间的一起爆炸的泡泡的数据，但是vector在去除中间数据不如list好，所以最后又改为使用了list。   
    3遇到过最坑的地方要数是：瓦片地图和整个场景的坐标原点是不一样的（一个是左上，一个是左下）！！刚制作完地图的那一阵我还清楚地记得这个事情。但是等到做泡泡炸人物的判断时，就完全忘记了这个事情。。。这么个小事，导致当时找了很久很久的Bug。。。   
    4.最后在实现局域网对战的过程中，由于时间紧张，我快速浏览了一些教程，但是只是学会了最简单的创建连接之后，进行简单的发送消息的操作。遇到了recv()函数在没有接收到消息的时候会阻塞，不能进行其他操作。想要这个问题我思考过是否可以通过多线程进行实现，但是我经过尝试发现短时间内我也没能学会多线程操作；之后考虑到给recv设置等待时间，但是没有想明白，如果没有收到数据，该怎么做。之后的局域网对战，也是由于没想明白如何解决recv函数的阻断等问题最终没有能实现。   
    5.文本框实现过程中用TextFieldTTF::textFieldWithPlaceHolder()函数中的第二个参数是字体，但是不能直接填字体名称，应该是写字体存放路径。这个事情也困扰了我很久。

**李万葶：**

在大项目中主要参与的工作有调整图片、整理素材、制作界面、剪辑音乐。详细的工作有配合队友把人物素材调整到相应的尺寸、制作了单人、双人游戏介绍的界面。

一些没有成功但是付出了一些时间的尝试：我尝试了将游戏移植到iOS手机平台，经过少量的修改游戏是可以在iPhone6s平台上运行的，屏幕适配等问题也基本解决了，但是在人物控制部分出现了问题，在电脑上人物是由上下左右按键控制的，我做了一组虚拟的按键来取代键盘，想用cocos的ButtonControl创建自定义按钮，代替键盘监听来控制人物，但是可能是由于Cocos的版本和网络教程有区别，存在各种bug，运行一直有问题，因为没有解决所以没有更多进展了。以后会继续尝试。

**周裕：**

前期准备：小组成员先聚集到一起，拟定了一个工作日程推进表，并进行了工作的划分。然后各自用了一周的时间大概地对cocos进行了学习探究，工作开始。

主要工作：主要负责美工制作，帮助图片的制作，寻找图片资源，背景音乐资源等。

尝试：用tiled制作了不同场景的地图，但是由于电脑原因，最后没有使用。尝试进行Android移植，在配备好环境之后，在进行编译时，首先选择了命令行进行尝试。在尝试使用命令行时，总是卡在adb命令，无法运行。

后来经过查阅发现，eclipse编译方式可能更为可行，但是最后没有时间了。