

现代操作系统应用开发实验报告

学号：15331180

班级：早上班，软工教务二班

姓名：练启业

实验名称：homework12

一．参考资料

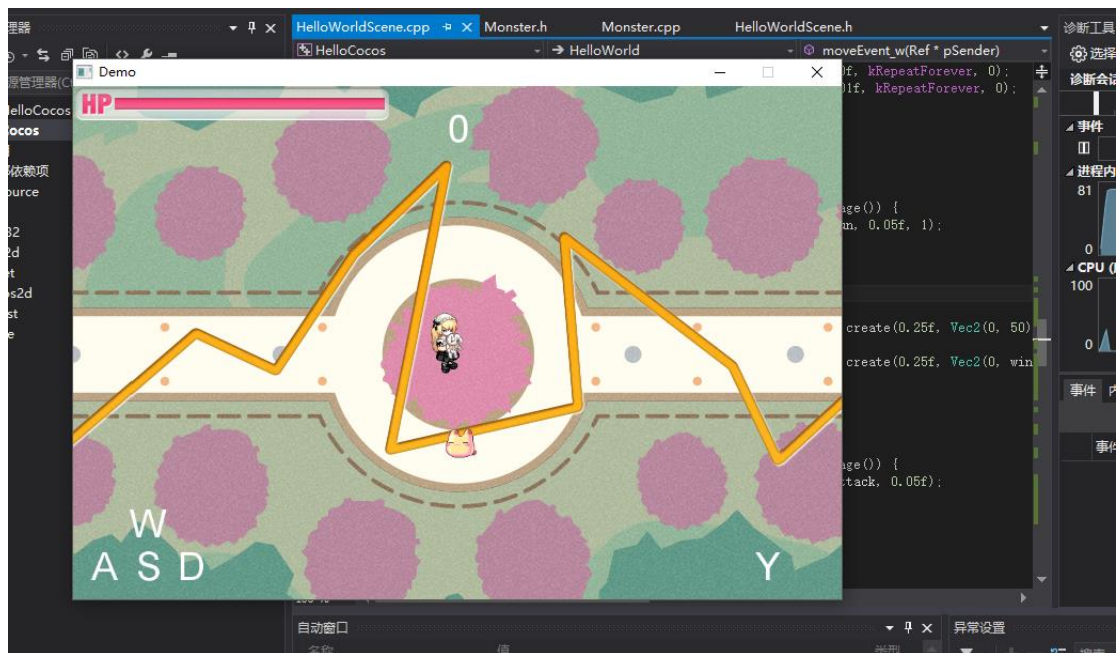
上课的课件

Demo 效果

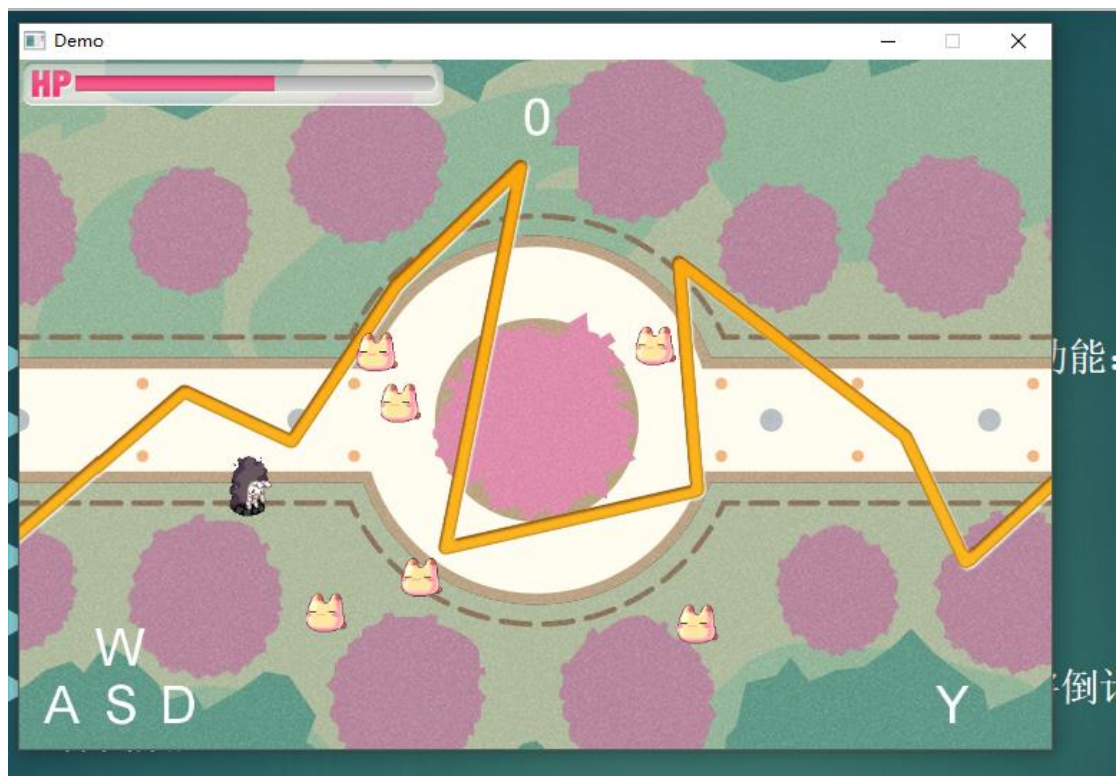
二．实验步骤

1. 为界面添加瓦片地图
2. 补充完整 Monster 类中的函数
3. 在 HelloWorld 类中利用调度器每秒生成怪物，并使其移动
4. 在 HelloWorld 类中利用调度器进行碰撞检验，角色碰到怪物掉血
5. 修改 Y 的回到函数，使其可以打死怪物并回血
6. 增加 XML 文件保存打死的怪物数量

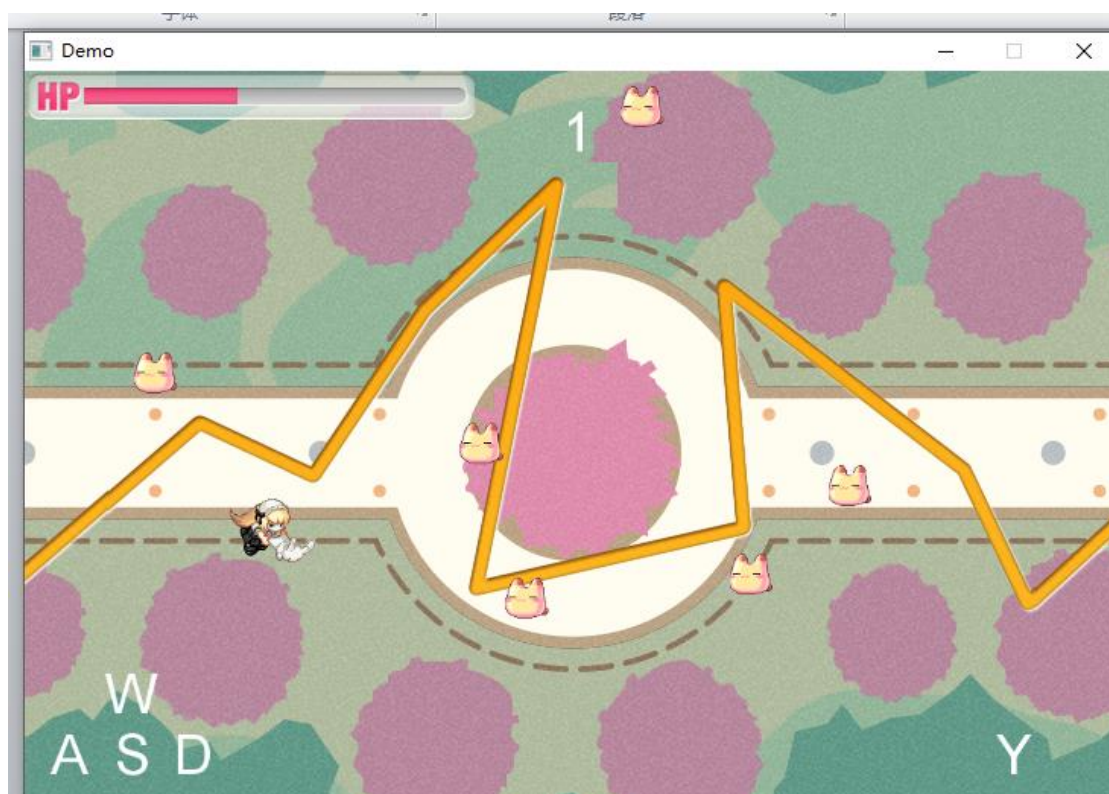
三．实验结果截图



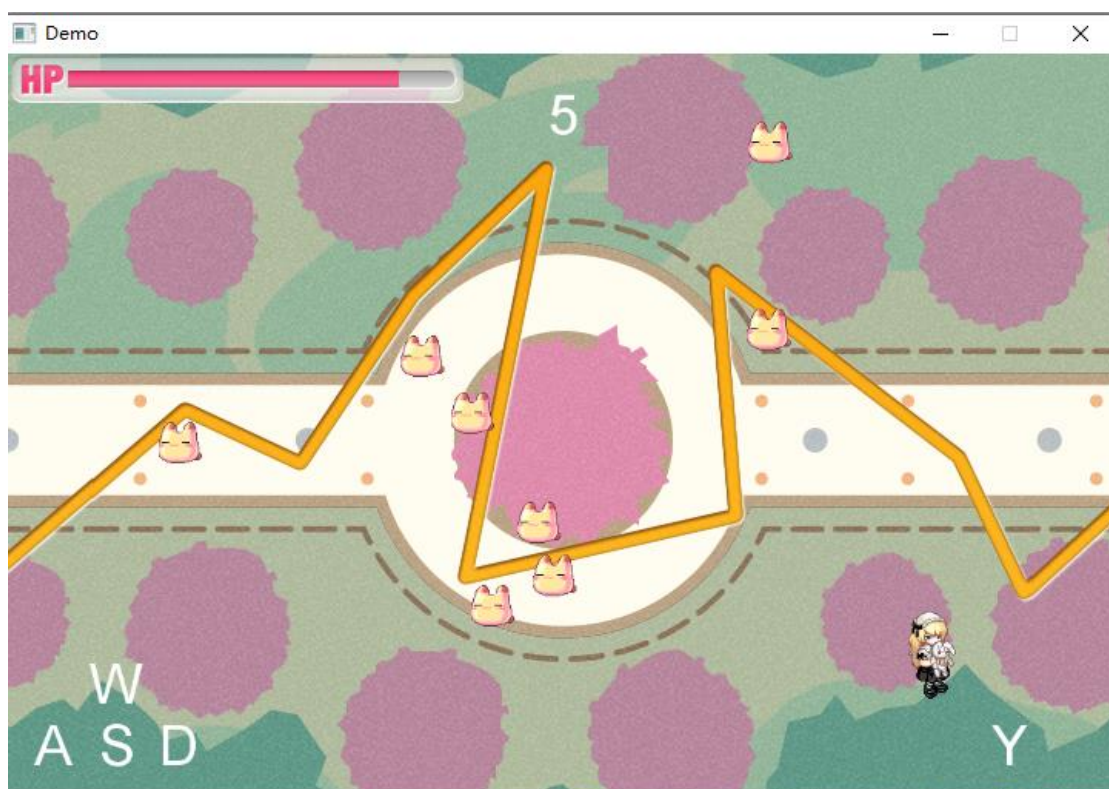
游戏运行，自动产生怪物，被怪物触碰会播放死亡动画并掉血

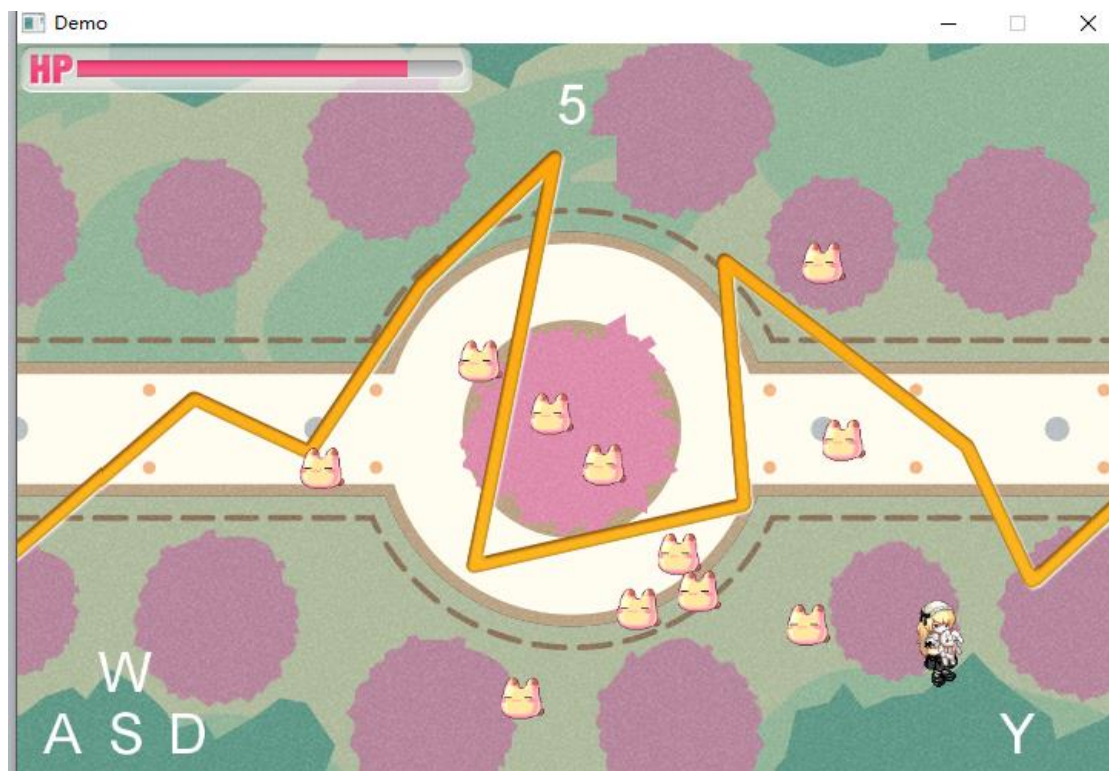


打死怪物，数量增加为 1，同时回一定的血量

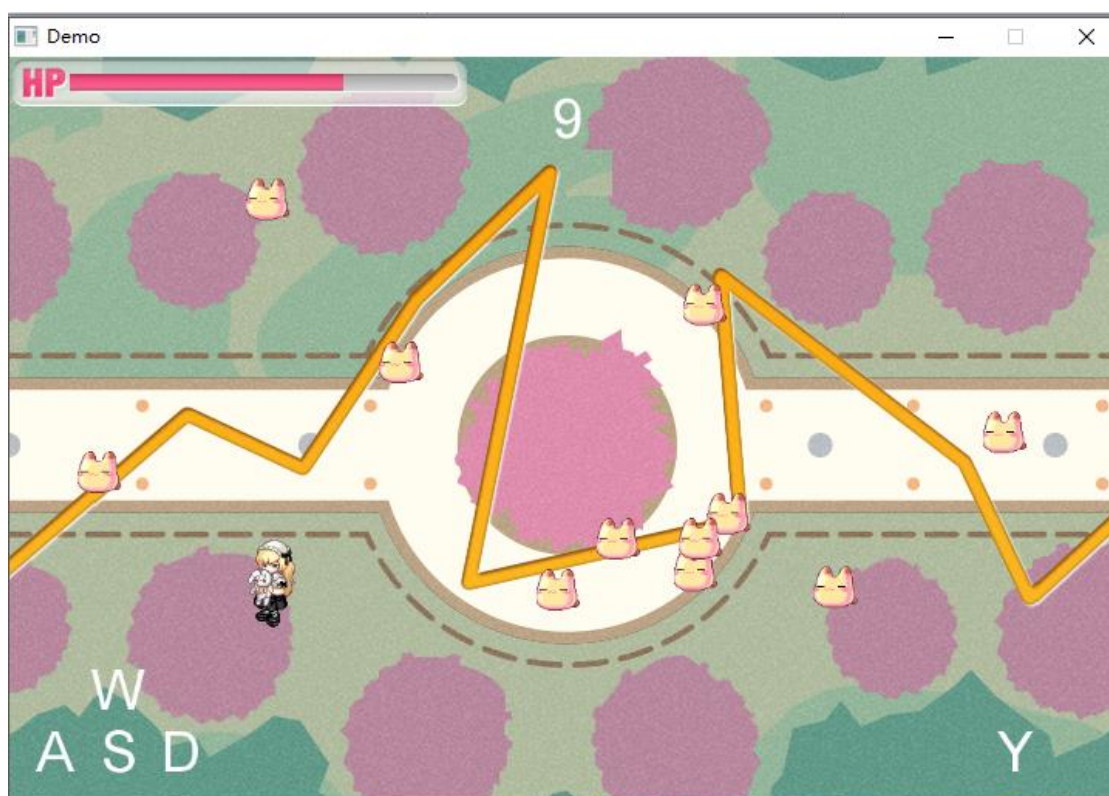


怪物会朝角色的方向前进





实现了人物左右移动的翻转

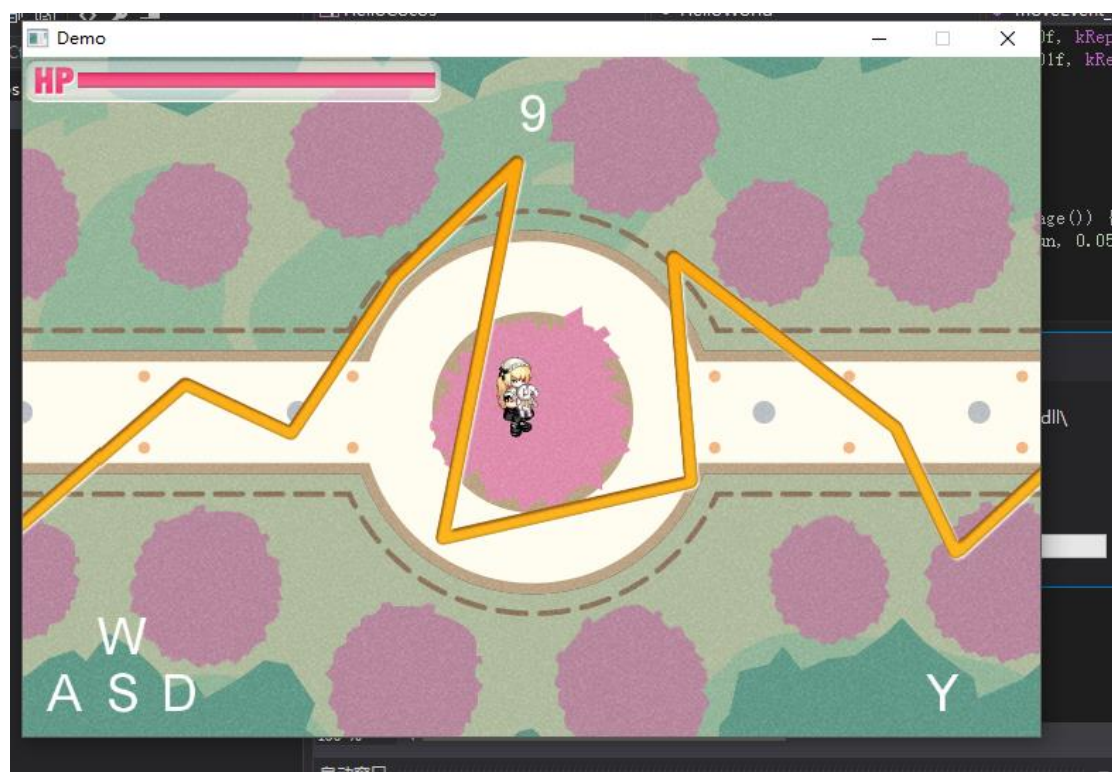


再次运行游戏，杀怪数量还是显示为 9

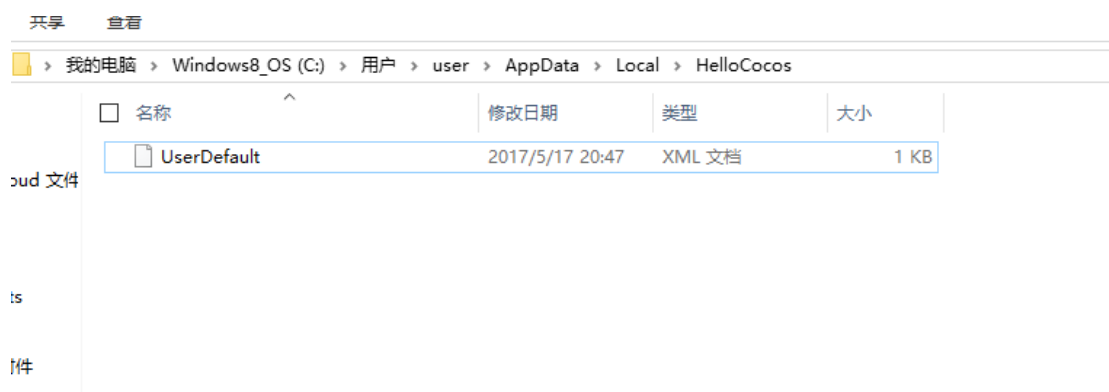
```
项目(P) 生成(B) 调试(D) 团队(M) 工具(T) 测试(S) 分析(N) 窗口(W) 帮助(H)
Debug Win32 本地 Windows 调试器 自动
Monster.h Monster 本地 Windows 调试器 HelloWorld
schedule(schedule_selector(HelloWorld::creatMonster), 3.0f, kRepeatForever, 0);
schedule(schedule_selector(HelloWorld::hitByMonster), 0.01f, kRepeatForever, 0);
this->addChild(time, 1);

return true;

HelloWorld::moveEvent_w(Ref* pSender) {
    if (player->getSpriteFrame() == frame0 && pT->getPercentage()) {
        auto animation = Animation::createWithSpriteFrames(run, 0.05f, 1);
        auto animate = Animate::create(animation);
        float winh = visibleSize.height; // 获取屏幕高度
        float playh = player->getContentSize().height;
        int nowy = player->getPosition().y;
        if (winh - nowy - playh / 2 >= 50) {
            player->runAction(Spawn::create(animate, MoveBy::create(0.25f, Vec2(0, 50)), nullptr));
        } else {
            player->runAction(Spawn::create(animate, MoveBy::create(0.25f, Vec2(0, winh - nowy - playh / 2)), nullptr));
        }
    }
}
```



Xml 文件保存在本地的文件中





四 . 实验过程遇到的问题

一开始实现碰撞函数的时候，发现人物在移动的过程中碰撞怪物，怪物消失，但是人物却没有播放死亡动画，后来发现是在上周的设置时，为了避免动画的重叠播放，设置了人物只有在第一帧的动画时，才能播放其他动画，因此适当调快了 move 动画，以减免这个问题的影响。

五 . 思考与总结

这次实验总的来说并不难，只要好好阅读课件，知道了类似 **vector** 的常用的数据结构的使用法，如何用 **xml** 保存数据，如何加载瓦片地图等就可以很容易的实现。当然实验过程中可能会出现很多问题，因此，我们要多调试，多检查，使效果更佳出色。