

# 现代操作系统应用开发实验报告

学号：153311117

班级：教务 2 班

姓名：黄楠绚

实验名称：homework

## 一 . 参考资料

请在这里列出对本实验有帮助你所参考的资料或者网站。

老师和 ta 的 ppt;

OglPMesh fatal error LNK1201: 写入程序数据库 “d:/OglPMesh/Debug/OglPMesh.pdb” 时出错 ;  
请检查是否是磁盘空间不足、路径无效或权限不够

<http://blog.csdn.net/danming60520/article/details/7743776?locationNum=14&fps>

=1

## 二 . 实验步骤

请在这里简要写下你的实验过程。

只实现前 5 点基本要求和加分要求第一点：

1.人物出场只实现简单的效果：粒子系统

```
// 人物出场特效 ——> 粒子系统 太阳粒子
CCParticleSun *particle = CCParticleSun::create();
particle->setPosition(Vec2(xpos, 180));
particle->setDuration(1);
this->addChild(particle, 2);
```

2.为玩家和箱子设置刚体属性, 重点是掩码和材料的设置。

3. 箱子下落过程过程能穿透玩家, 不会发生碰撞。

玩家的掩码为 5, 下落的箱子掩码为 2;

设置禁止旋转防止箱子和玩家在移动的过程中发生碰撞而导致刚体旋转, 也可防止玩家掉进

海里；

箱子重量和摩擦力系数都设置到非常大，防止玩家推动箱子，可能设的这些值在后面的举起箱子会导致一些 bug，但是举起箱子没有实现不考虑这个

```
// 设置角色刚体属性
auto physicsBody = PhysicsBody::createBox(player1->getContentSize(), PhysicsMaterial(20.0f, 0.0f, 1.0f));
physicsBody->setDynamic(true);
physicsBody->setCategoryBitmask(0x0005); // 0101
physicsBody->setCollisionBitmask(0x0005);
physicsBody->setContactTestBitmask(0x0005);
physicsBody->setRotationEnable(false); // 禁止旋转
player1->setPhysicsBody(physicsBody);
```

```
// 为箱子设置刚体属性
auto physicsBody = PhysicsBody::createBox(box->getContentSize(), PhysicsMaterial(100.0f, 0.0f, 100000.0f));
physicsBody->setDynamic(true);
physicsBody->setCategoryBitmask(0x0002); // 0010
physicsBody->setCollisionBitmask(0x0002);
physicsBody->setContactTestBitmask(0x0002);
physicsBody->setRotationEnable(false);
physicsBody->setMass(100000000);
box->setPhysicsBody(physicsBody);
box->setTag(20);
```

#### 4.箱子碰到船或者碰到其他箱子之后，能和玩家发生碰撞。

根据掩码判断碰撞的物体是否为下落的箱子，并把该箱子掩码改为 3；

```
// 箱子碰到船或者碰到其他箱子之后改变掩码，可以与玩家发生碰撞
// Todo
bool FriendShip::onConcaetBegin(PhysicsContact & contact) {
    auto nodeA = contact.getShapeA()->getBody()->getNode();
    auto nodeB = contact.getShapeB()->getBody()->getNode();

    if (nodeA && nodeB) {
        if (nodeA->getPhysicsBody()->getCategoryBitmask() == 0x0002) {
            nodeA->getPhysicsBody()->setCategoryBitmask(0x0003); // 0011
            nodeA->getPhysicsBody()->setCollisionBitmask(0x0003);
            nodeA->getPhysicsBody()->setContactTestBitmask(0x0003);
        }
        else if (nodeB->getPhysicsBody()->getCategoryBitmask() == 0x0002) {
            nodeB->getPhysicsBody()->setCategoryBitmask(0x0003); // 0011
            nodeB->getPhysicsBody()->setCollisionBitmask(0x0003);
            nodeB->getPhysicsBody()->setContactTestBitmask(0x0003);
        }
    }
}
```

#### 5.控制玩家左右移动

```

use cocos2d::EventKeyboard::KeyCode::KEY_RIGHT_ARROW.
// 左右移动
if (code == EventKeyboard::KeyCode::KEY_LEFT_ARROW) {
    IsPlayer1Left = true;
    if (LastPlayer1Press == 'D') {
        LastPlayer1Press = 'A';
        player1->setFlippedX(true); // 人物动画左右对称
    }
}
else {
    IsPlayer1Right = true;
    if (LastPlayer1Press == 'A') {
        LastPlayer1Press = 'D';
        player1->setFlippedX(false);
    }
}
break;

```

在 update 中判断，先播放动画再移动：

```

// 玩家1是否在举着箱子的情况下 左右移动
// 播放动画
if (IsPlayer1Left || IsPlayer1Right) {
    if (player1->getSpriteFrame() == frame1) {
        auto animation = RepeatForever::create(Animate::create(AnimationCache::getInstance()->getAnimation("player1RunWithoutBoxAnimation")));
        animation->setTag(11);
        player1->runAction(animation);
    }
    else if (player1->getSpriteFrame() == IdleWithBox1) {
        auto animation = RepeatForever::create(Animate::create(AnimationCache::getInstance()->getAnimation("player1RunWithBoxAnimation")));
        animation->setTag(11);
        player1->runAction(animation);
    }
}
// 移动动作
if (IsPlayer1Left && player1->getPosition().x - 3 > origin.x) {
    auto moveBy = MoveBy::create(0.01f, Vec2(-1, 0));
    player1->runAction(moveBy);
}
else if (IsPlayer1Right && player1->getPosition().x + 3 < origin.x + visibleSize.width) {
    auto moveBy = MoveBy::create(0.01f, Vec2(1, 0));
    player1->runAction(moveBy);
}
}

```

松开左右键，静止动画：

```

// 停止动画和运动
player1->stopAllActions();
if (IsPlayer1Hold == true) { // 精灵举着箱子
}
else { // 精灵不举着箱子
    player1->setSpriteFrame(frame1);
    if (LastPlayer1Press == 'A')
        player1->setFlippedX(true);
}
break;

```

加分要求：

1. 实现人物跳跃

```
case cocos2d::EventKeyboard::KeyCode::KEY_UP_ARROW:
    // 跳
    // 判断玩家处于速度为0的状态
    if (player1->getPhysicsBody()->getVelocity().getDistance(Vec2(0, 0)) - 0 < 0.01) {
        IsPlayer1Jump = true;
    }
    break;
```

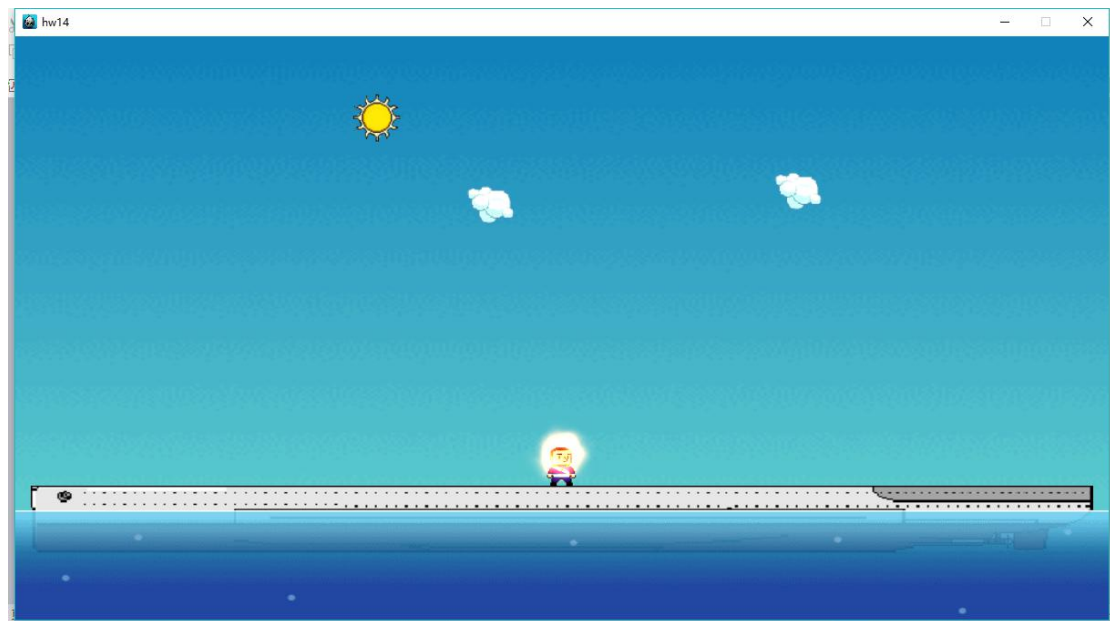
update 中设置一个线性速度：

```
if (IsPlayer1Jump) {
    player1->getPhysicsBody()->setVelocity(Vec2(0, player1->getPosition().y + 50));
    IsPlayer1Jump = false;
}
```

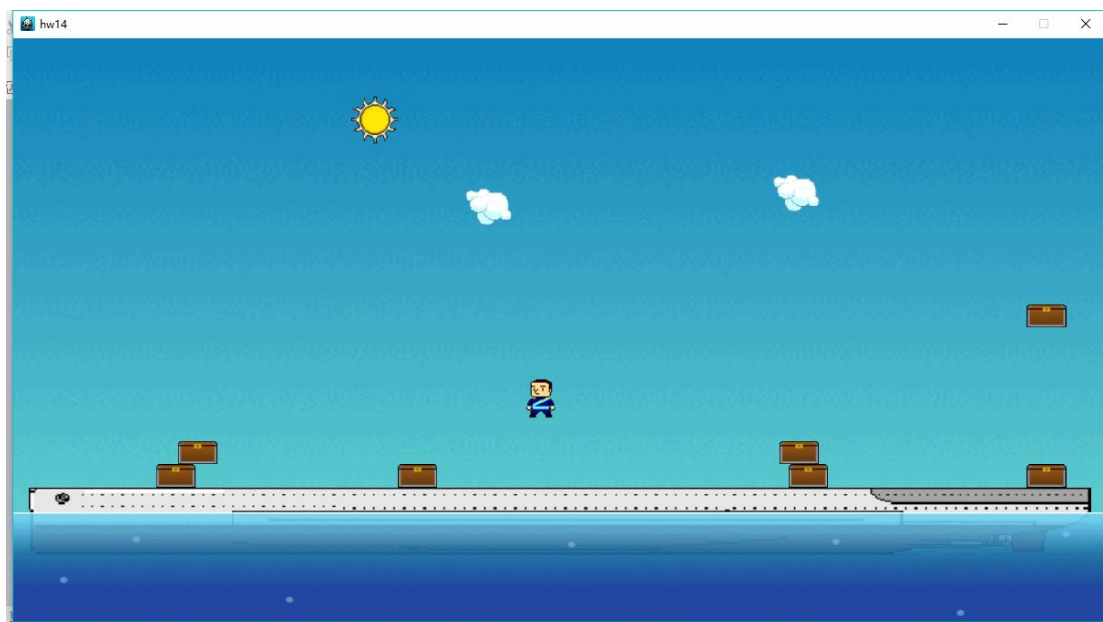
### 三．实验结果截图

请在这里把实验所得的运行结果截图。

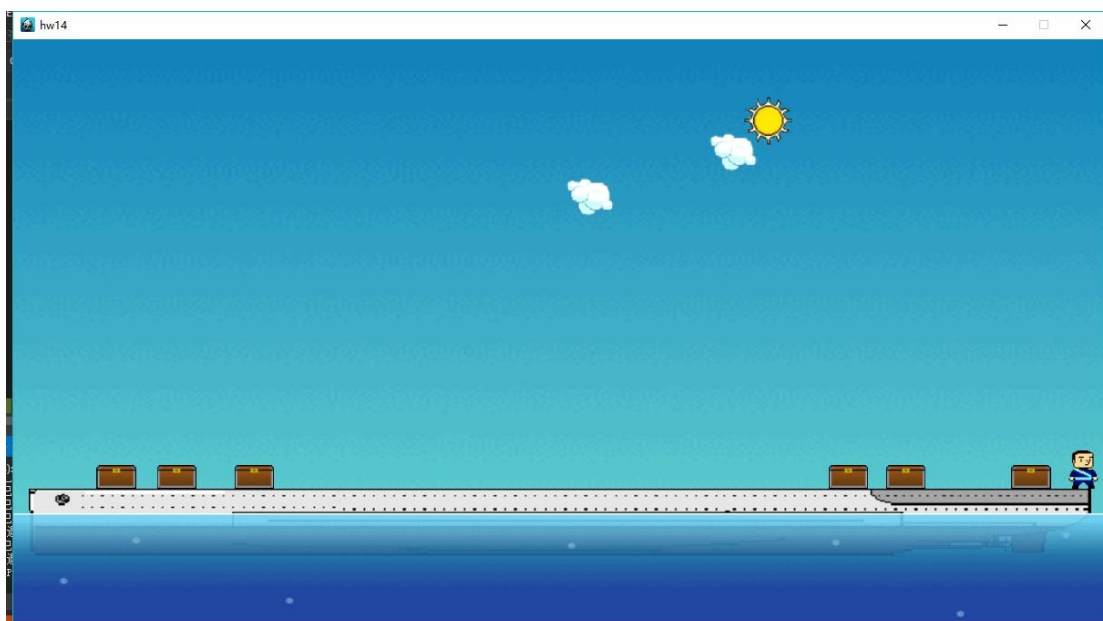
人物出场，出现特技效果：太阳粒子



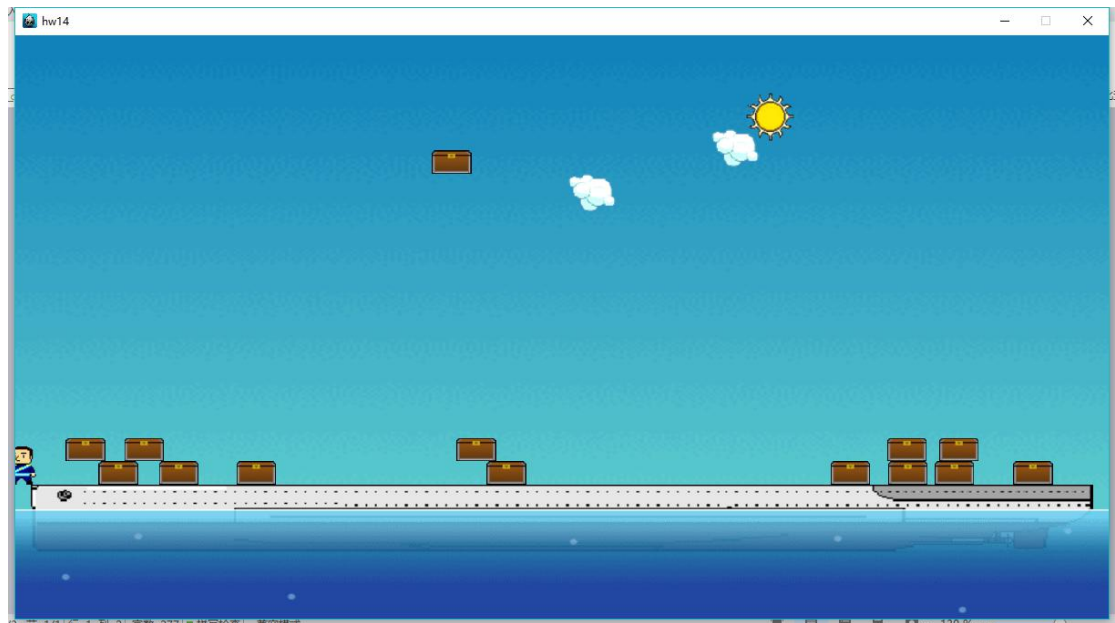
人物可以左右移动且向上跳跃：



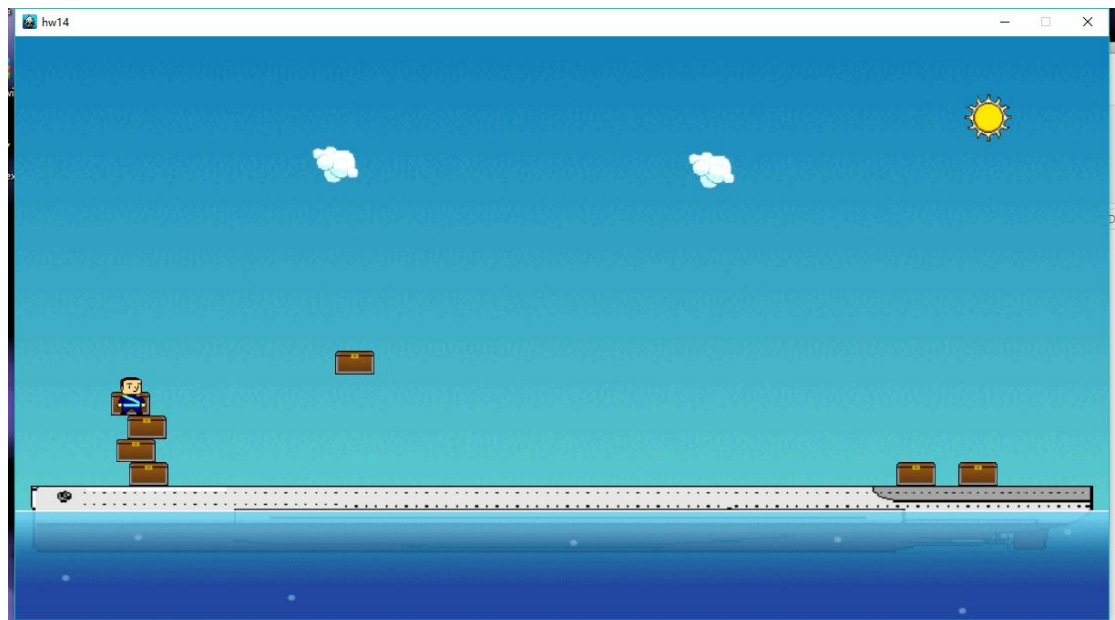
人物走到船的左右边不会掉海里：



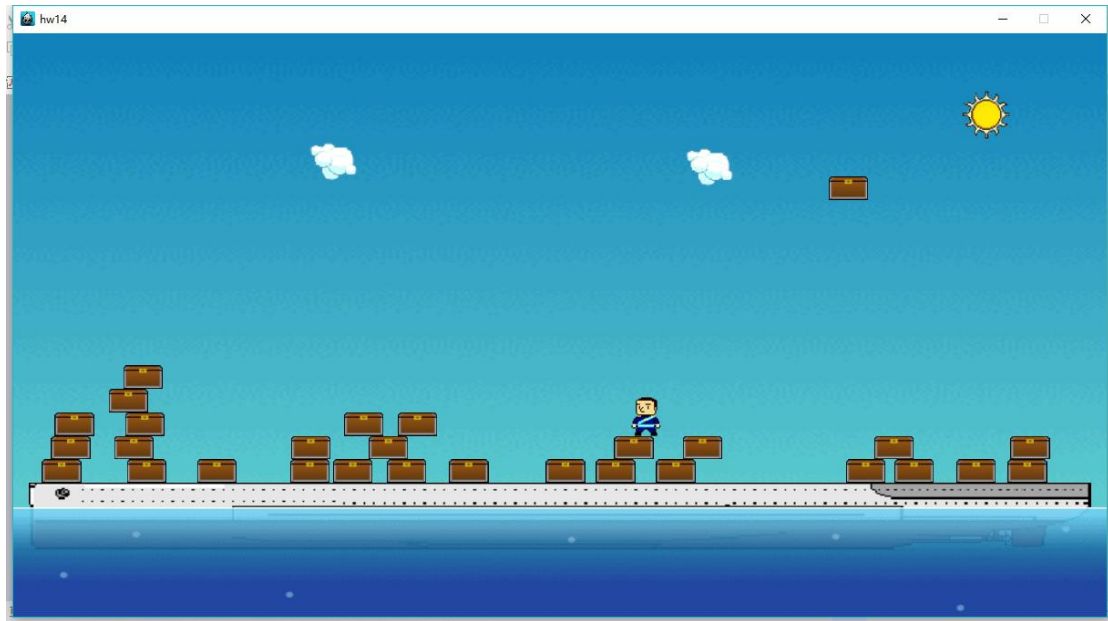




箱子刚落下，可以穿透人：



箱子掩码改变后，人不能穿透箱子：



#### 四．实验过程遇到的问题

请在这里写下你在实验过程中遇到的问题以及解决方案。

1. ta 给的代码刚开始运行的时候会出现报错，貌似是游戏结束时的“退出”label 的锅，不太懂为什么会这样，解决方法时在退出后面补个空格就可以运行了……搞不懂是为什么

2. 运行时出现报错：

OgIPMesh fatal error LNK1201: 写入程序数据库 “d:/OgIPMesh/Debug/OgIPMesh.pdb” 时出错；  
请检查是否是磁盘空间不足、路径无效或权限不够

试了一下网上的方法不行，然后把 Debug 的这个 pdb 删除重新编译就好了

3. 出现刚体穿透的问题：



试过网上的办法不能解决，想了一些奇技淫巧，花了一个晚上捣鼓效果不够理想就放弃了

4. 固定距离关节固定了玩家和箱子后，距离并不是固定的……不知道把箱子放在玩家头顶上就能避免这个问题？最终由于时间原因，没有实现

#### 五．思考与总结

请在这里写下你本次试验的心得体会以及所思所想。

ta 大大说得对，世上无难事，只要肯放弃。于是挣扎一天半把简单要求实现出来后

果断弃疗，关节那部分本来想做好的，在玩家移动的过程中偶尔会出现 bug，但由于最近在准备另外的事情就没有完成，很惭愧。不禁感叹 ta 大大当年做这个游戏的艰辛。

最后就是好好准备期末项目（但是关节的坑我没有怎么踩肯定会吃亏。

1. 实验报告提交格式为 pdf。
2. 实验内容不允许抄袭，我们要进行代码相似度对比。如发现抄袭，按 0 分处理。