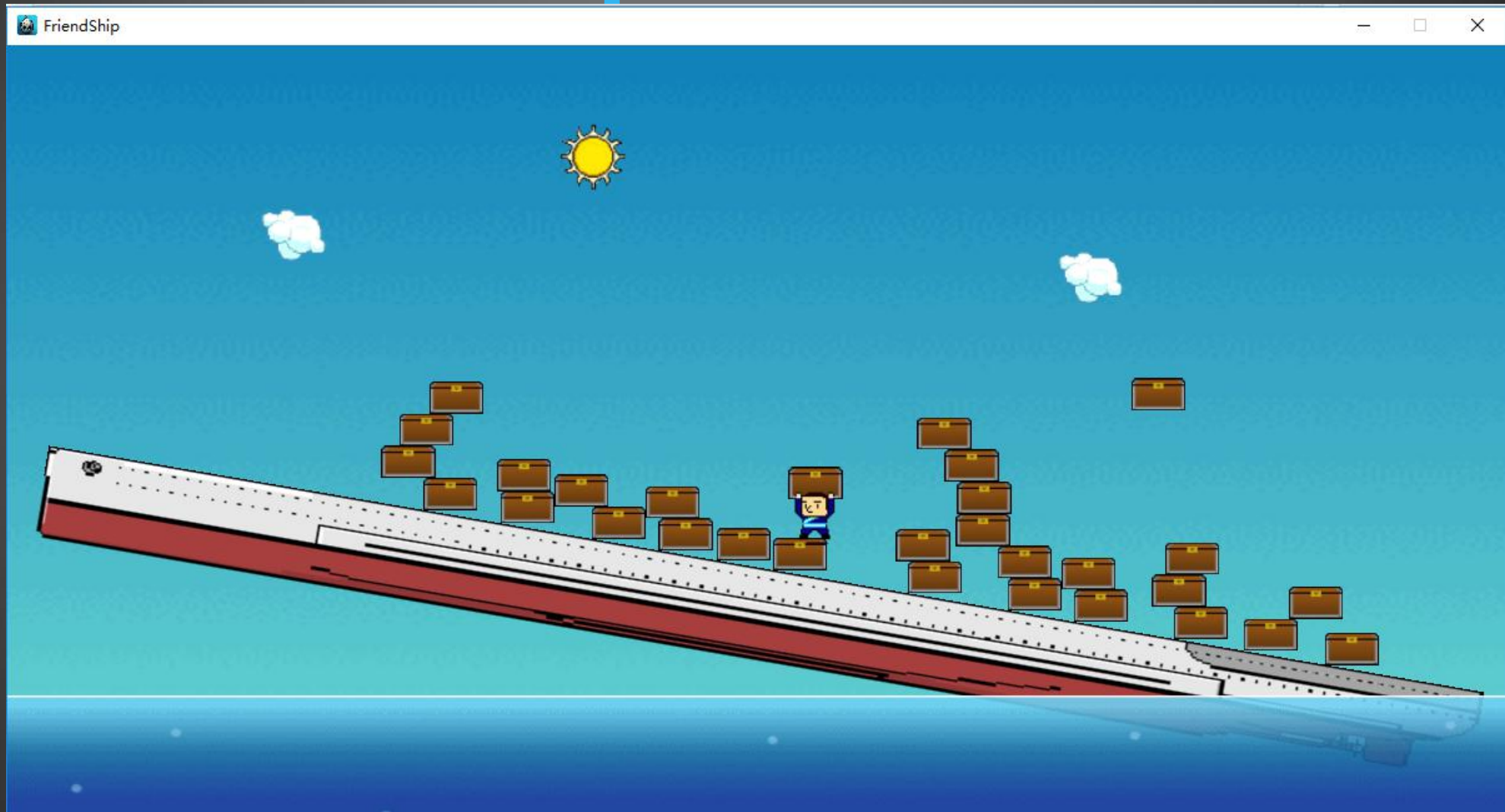


# Hw14 物理引擎与粒子系统

黄建武



# hw: Friendship



# hw: Friendship

基本要求：难度逐渐加大

1. 至少使用一种粒子系统。(如人物出场或箱子碰撞)
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。
4. 箱子碰到船或者碰到其他箱子之后, 能和玩家发生碰撞。
5. 控制玩家左右移动。
6. 使用关节举起和扔下箱子。(固定距离关节或其他)



# hw: Friendship

加分要求:

1. 实现人物跳跃
2. 举着箱子的情况下跑动和跳跃
3. 轮船倾斜以及翻船



# intersectsRect 碰撞检测

```
void update(float dt) {  
    auto a = sprite1->getBoundingBox();  
    auto b = sprite2->getBoundingBox();  
  
    if (a.intersectsRect(b)) {  
        // boom  
    }  
}
```





# 创建物理场景

```
PhysicsWorld* m_world;

void FriendShip::setPhysicsWorld(PhysicsWorld* world) { m_world = world; }

Scene* FriendShip::createScene() {
    auto scene = Scene::createWithPhysics();
    scene->getPhysicsWorld()->setGravity(Vec2(0, -300.0f));
    auto layer = FriendShip::create();
    layer->setPhysicsWorld(scene->getPhysicsWorld());
    scene->addChild(layer);
    return scene;
}
```

默认的物理场景重力为Vec2(0, -98.0f), `_autostep = True`



# autostep 自动步进

刷新物理世界的过程也被称为步进(steppping)。默认情况下，物理世界会不停地进行自动刷新。这被称为“自动步进(autostepping)”。

Windows下自动步进可能会出现很尴尬的情况, 刚体满天飞, 不受控制。



# autostep 自动步进



<http://img.blog.csdn.net/20151116110620023?watermark/2/text/aHR0cDovL2Jsbn2cuY3Nkbi5uZXQv/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/Center>



COCOS2D X



# 关闭自动步进

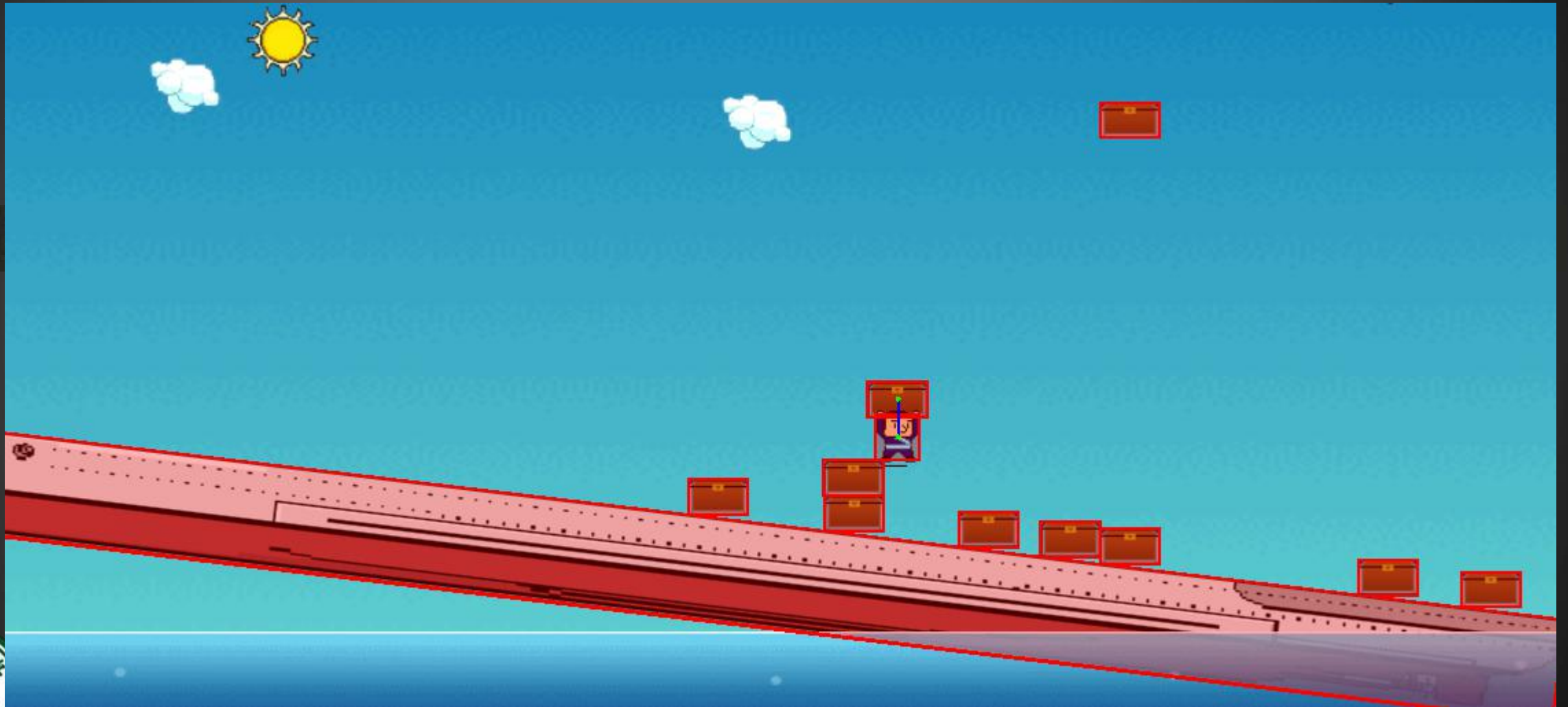
```
scene->getPhysicsWorld()->setAutoStep(false);
```

关闭自动步进后, 就需要手动步进, 设置一个调度器, 在update函数里执行

```
void FriendShip::update(float dt) {  
    m_world->step(1 / 100.0f);  
}
```



# DebugDraw



COCOS2D X

另一件很尴尬的事, `setAutoStep(false)`和`DebugDraw`好像不能同时使用, 这次作业如果关闭自动步进并且开启Debug模式会发生段错误



# Bitmask

每个刚体都有categoryBitmask, collisionBitmask, contactTestBitmask属性

两个刚体的categoryBitmask和collisionBitmask互相按位与都不为0, 会发生碰撞, 否则不发生碰撞;

两个刚体的categoryBitmask和contactTestBitmask互相按位与都不为0, 会触发碰撞检测监听器, 否则不会有反馈。

如果要在碰撞的时候都会触发监听器, 最简单的方法就是同一刚体三个Bitmask设置相同的数值。



1. 箱子下落过程过程能穿透玩家, 不会发生碰撞。
2. 箱子碰到船或者其他箱子之后, 能和玩家发生碰撞。

假设各刚体的掩码如下:

船s, 玩家p, 下落过程的箱子b1,  
接触到船或其他箱子的箱子b2

```
s & p != 0  
s & b1 != 0  
s & b2 != 0  
b1 & p = 0  
b2 & p != 0  
b1 & b2 != 0
```



COCOS2D X



船和其他刚体都会发生碰撞，所以所有刚体的掩码都不为0，船的掩码是一个和任何数按位与都不为0的数

```
auto shipbody = PhysicsBody::createBox(ship-&gtgetContentSize(), PhysicsMaterial(100.0f, 0.0f, 1.0f));
shipbody->setCategoryBitmask(0xFFFFFFFF);
shipbody->setCollisionBitmask(0xFFFFFFFF);
shipbody->setContactTestBitmask(0xFFFFFFFF);
shipbody->setDynamic(false); // 静态刚体不受重力影响，同时也无法发生倾斜了
ship->setPhysicsBody(shipbody);
```

密度100, 完全非弹性碰撞, 摩擦系数1.0





### 3. 控制玩家左右移动和跳跃

```
player1->getPhysicsBody()->setVelocity(Vec2(x, y));
```

该速度会因为摩擦阻力和重力的影响逐渐变小



COCOS2D X

## 4. 举起和放下箱子

遍历所有的箱子,找到第一个与玩家接触的箱子,对玩家和该箱子使用固定距离关节,判断接触可以简单的使用

`intersectsRect`



5 五八 白凡九百全

FriendShip

00:00

最高分: 02:56



LULUCU

22:21



## 4. 轮船倾斜

标准做法？

我不知道，如果你能谷歌到轮船倾斜的标准做法，写进实验报告，这次作业已经满分了

奇技淫巧骚操作？

有，自己想，顺便在实验报告说明，最好能让我感叹“还有这种操作???”



## 最后再说几句

这次作业有点难度(大神当我没说), 不过最大的几个坑我都已经说了

有问题多google, 英文搜索, 注意找PhysicsWord的资料, 网上很多都是针对Box2D的

多用自动补全, 对象有什么属性和方法一目了然, 然后再去查具体的使用方法

如果实在写不出来就算了, 好好准备期末项目



世上无难事  
只要肯放弃





THE END

THANKS FOR WATCHING

