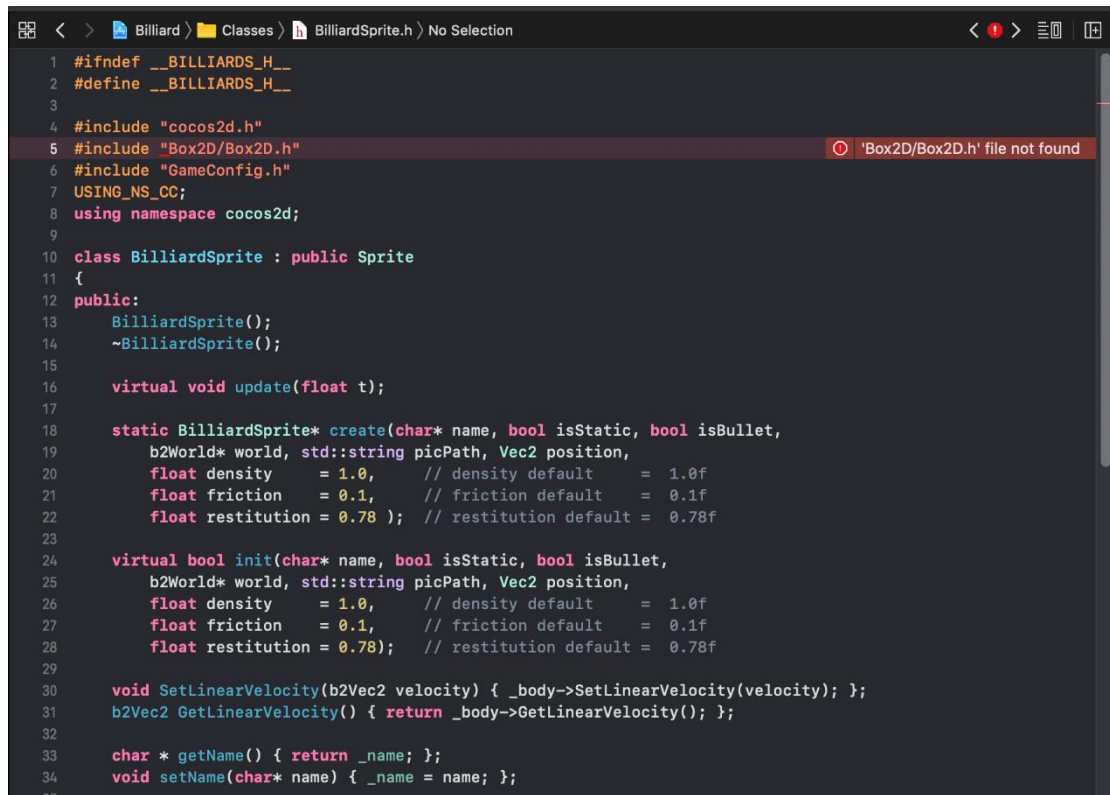


## 《计算机游戏设计》实验 4 常见问题

### 问题 1:

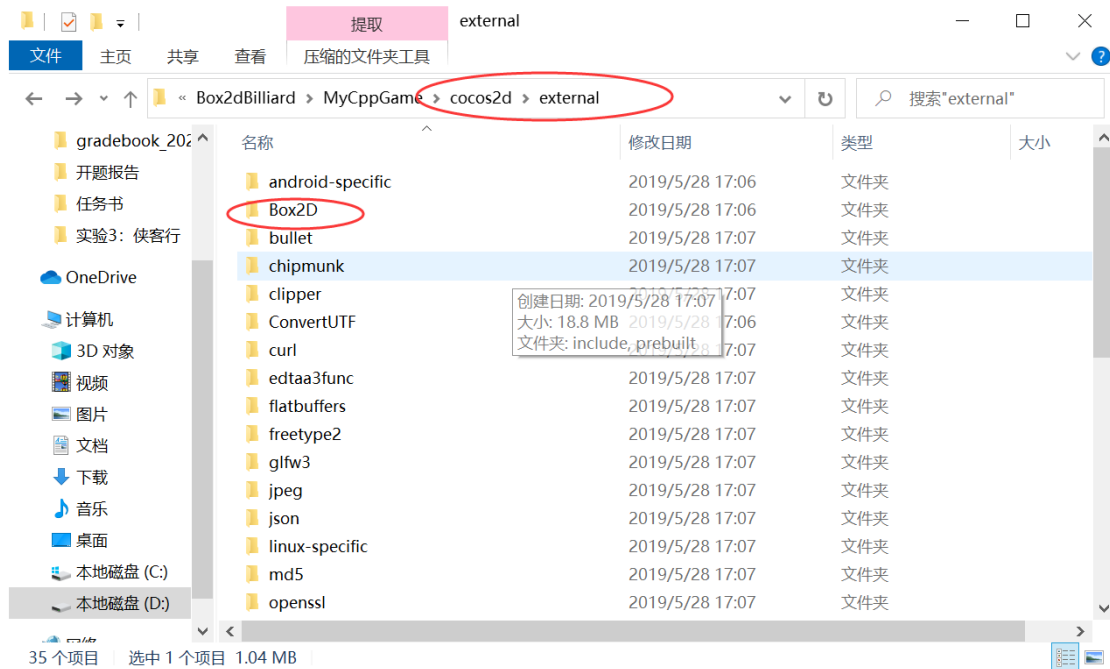
找不到 Box2D/Box2D.h 文件，cocos2d-x 版本为 3.17 及以上（如下图所示）。



```
1 #ifndef __BILLIARDS_H__
2 #define __BILLIARDS_H__
3
4 #include "cocos2d.h"
5 #include "Box2D/Box2D.h"
6 #include "GameConfig.h"
7 USING_NS_CC;
8 using namespace cocos2d;
9
10 class BilliardSprite : public Sprite
11 {
12 public:
13     BilliardSprite();
14     ~BilliardSprite();
15
16     virtual void update(float t);
17
18     static BilliardSprite* create(char* name, bool isStatic, bool isBullet,
19     b2World* world, std::string picPath, Vec2 position,
20     float density = 1.0, // density default = 1.0f
21     float friction = 0.1, // friction default = 0.1f
22     float restitution = 0.78 ); // restitution default = 0.78f
23
24     virtual bool init(char* name, bool isStatic, bool isBullet,
25     b2World* world, std::string picPath, Vec2 position,
26     float density = 1.0, // density default = 1.0f
27     float friction = 0.1, // friction default = 0.1f
28     float restitution = 0.78 ); // restitution default = 0.78f
29
30     void SetLinearVelocity(b2Vec2 velocity) { _body->SetLinearVelocity(velocity); };
31     b2Vec2 GetLinearVelocity() { return _body->GetLinearVelocity(); };
32
33     char * getName() { return _name; };
34     void setName(char* name) { _name = name; };
35 }
```

原因：课本代码是 cocos3.15 版本，而 3.17 以上版本的 Box2D 目录结构不一样，因而出现引用错误。

解决方案：把 3.15 版本的 Box2D 文件夹复制到项目中的相应目录下，覆盖（如下图所示）。



## 问题 2

groundBody->SetUserData("boundary")报错（如下图所示）。

```
61 // ground
62 b2BodyDef groundBodyDef;
63
64 // left down
65 groundBodyDef.position.Set(0, 0);
66
67 // create ground
68 b2Body* groundBody = _world->CreateBody(&groundBodyDef);
69 groundBody->SetUserData("boundary"); // Cannot initialize a parameter of type 'void **' with an lvalue of type 'const char [9]'
70
71 // shape
72 b2EdgeShape groundBox;
73
74 // down
75 groundBox.Set(b2Vec2(0,0),b2Vec2(s.width/PTM_RATIO,0));
76 groundBody->CreateFixture(&groundBox,0);
77
```

解决方案：

加上强制类型转换。

```
291 // create ground
292 b2Body* groundBody = _world->CreateBody(&groundBodyDef);
293 groundBody->SetUserData((void *) "boundary");
294
295 // shape
296 b2EdgeShape groundBox;
297
```

### 问题 3:

添加球洞后，出现空指针错误，调用堆栈看到是在 BilliardSprite 的 init 函数中出错。

```
        bodyDef.bullet = true;
    }
    bodyDef.position.Set(position.x / PTM_RATIO, position.y / PTM_RATIO);
    // 创建body
    _body = world->CreateBody(&bodyDef);
    _body->SetUserData(this);

    b2CircleShape circle;
    circle.m_radius = (_sprite->getContentSize().width / 2) / PTM_RATIO;
    if (isStatic)
    {
```

解决方案:

在场景初始化中，把球洞初始化放在 this->initPhysics();后面

```
this->addChild(boundary_left);

// boundary right
auto boundary_right = Sprite::create("Chapter07/slider/bg-left.png");
boundary_right->setPosition(Vec2(visibleSize.width - 2, visibleSize.height/2));
this->addChild(boundary_right, 2);

// init aim rect
_aimRect = Rect(0, 0, visibleSize.width - 60, visibleSize.height);

// init physics world
this->initPhysics(); // 这句话先执行了，才能创建球洞和球

// 白球
_mainBilliards = BilliardSprite::create("main", false, true, _world, "Chapter07/Ball/8
this->addChild(_mainBilliards, 10, MAIN_BILLIARDS_TAG);

for (int i=0; i<5; i++)
{
    auto ballSprite = BilliardSprite::create("ball", false, false, _world, ball[i], ball
    this->addChild(ballSprite);
}

// add touch listener
auto listener = EventListenerTouchOneByOne::create();
listener->onTouchBegan = CC_CALLBACK_2(PhysicsBox2dScene::onTouchBegan, this);
```

### 问题 4

教材源代码运行后，球撞到边界会报错。

原因：源代码中球碰撞的时候，会调用 BilliardSprite 的 getUserData ()。但如果碰撞一方是边界，此时就会出现空指针和内存访问错误（如下图所示）。

```
27 BilliardSprite* A = static_cast<BilliardSprite*>(contact->GetFixtureA()->GetBody()->GetUserData());
28 BilliardSprite* B = static_cast<BilliardSprite*>(contact->GetFixtureB()->GetBody()->GetUserData());
```

解决方案：

用 char\* 得到碰撞的 a 和 b，只要有一个是边界，就 return 掉，不响应此次碰撞。

```
14 void GameContactListener::BeginContact(b2Contact*contact)
15 {
16     char *a = (char*)contact->GetFixtureA()->GetBody()->GetUserData();
17     char *b = (char*)contact->GetFixtureB()->GetBody()->GetUserData();
18
19     if (strcmp(a, "boundary") == 0 || strcmp(b, "boundary") == 0)
20         return;
21
22     CCLOG("%s", static_cast<BilliardSprite*>(contact->GetFixtureA()->GetBody()->GetUserData()->getName()));
23     CCLOG("%s", static_cast<BilliardSprite*>(contact->GetFixtureB()->GetBody()->GetUserData()->getName()));
24
25 }
```