

实验三报告

一、 单元测试

1. 测试用例（一个方法一个测试用例）

用例编号	1			
待测试类及方法	类名: EliteEnemy 方法: shoot			
测试类及方法	类名: EliteEnemyTest 方法: shoot			
前提条件（如有）	创建一个 EliteEnemy 类的对象			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断精英敌机是否按照指定要求进行射击产生子弹。	用 EliteEnemy 类中的方法 shoot 返回子弹集合，理论上，这个子弹集合有一个子弹，用 assertEquals 方法，判断集合是否为空，再用子弹的 getLocationX，getLocationY, getSpeedY, getSpeedX 方法返回子弹的速度和产生位置，判断是否符合要求。	false	false	通过
		EliteEnemy 的 x 轴位置	enemy.getLocationX()	
		EliteEnemy 的 Y 轴位置+2	enemy.getLocationY()+ 1*2	
		Elite 的 Y 轴速度+5	enemy.getSpeedY()+ 1*5	
		0	0	

用例编号	2			
待测试类及方法	类名: EliteEnemy 方法: getProp			
测试类及方法	类名: EliteEnemyTest 方法: getProp			
前提条件 (如有)	创建一个 EliteEnemy 类的对象			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断精英敌机坠毁时是否按照指定要求产生道具。	用 EliteEnemy 类中的方法 getProp 返回道具, 用 assertEquals 方法, 判断 否 为 空或是三种道具中的一种, 再用道具的 getLocationX, getLocationY, getSpeedY, getSpeedX 方法返回道具的速度和产生位置, 判断是否符合要求。	True	True	通过
		enemy.getLocationX	enemy.getLocationX	
		enemy.getLocationY	enemy.getLocationY	
		0	0	
		0	0	

用例编号	3			
待测试类及方法	类名: HeroAircraft 方法: decreaseHp			
测试类及方法	类名: HeroAircraftTest 方法: decreaseHp			
前提条件 (如有)	创建一个 HeroAircraft 类的对象 heroAircraft			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断英雄机是否能正确减少血量。	设以一个 100 以内的随机数 power, 用 HeroAircraft 中的 decreaseHp 方法传入 power 参数, 判断是否于与 100-power 相对。之后用 HeroAircraft 中的 decreaseHp 方法传入 100, 观察边界值。	100-Power	100-Power	通过
		0	0	

用例编号	4			
待测试类及方法	类名: HeroAircraft 方法: shoot			
测试类及方法	类名: HeroAircraftTest 方法: shoot			
前提条件 (如有)	创建一个 HeroAircraft 类的对象 heroAircraft			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断英雄机是否按照指定要求进行射击产生子弹。	用 HeroAircraft 类中的方法 shoot 返回子弹集合, 这个子弹集合有一个子弹, 用 assertEquals 方法, 判断集合是否为空, 再通过子弹的 getLocationX, getLocationY, getSpeedY, getSpeedX 方法返回子弹的速度和产生位置, 判断是否符合要求。	False heroAircraft.getLocationX() heroAircraft.getLocationY()+(-1)*2 heroAircraft.getSpeedY()+(-1)*5 0	False heroAircraft.getLocationX() heroAircraft.getLocationY()+(-1)*2 heroAircraft.getSpeedY()+(-1)*5 0	通过

用例编号	5			
待测试类及方法	类名: MobEnemy 方法: shoot			
测试类及方法	类名: MobEnemyTest 方法: shoot			
前提条件 (如有)	创建一个 MobEnemy 类的对象			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断普通敌机是否按照指定要求不产生子弹。	用 MOBEnemy 类中的方法 shoot 返回子弹集合, 用 assertEquals 方法, 判断集合是否为空。	true	true	通过

用例编号	6			
待测试类及方法	类名：MobEnemy 方法：getProp			
测试类及方法	类名：MobEnemyTest 方法：getProp			
前提条件（如有）	创建一个 MobEnemy 类的对象			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断普通敌机坠毁时是否按照指定要求进行射击产生道具。	用 EliteEnemy 类中的方法 getProp 返回道具，用 assertEquals 方法，判断是否为空。	True	True	通过

用例编号	7			
待测试类及方法	类名：BaseBullet 方法：forward			
测试类及方法	类名：BaseBulletTest 方法：forward			
前提条件（如有）	创建一个 BaseBullet 类的对象 bullet			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断子弹是否能够按照要求移动。	调用一次 BaseBullet 中的方法 forward，用 assertEquals 方法判断子弹位置与预期位置是否相同。	20	20	通过

用例编号	8			
待测试类及方法	类名：BloodProp 方法：effect			
测试类及方法	类名：BloodPropTest 方法：effect			
前提条件（如有）	创建一个 BloodProp 类的对象，HeroAircraft 类的对象 heroAircraft			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断回血道具能否给英雄机加血。	用 BloodProp 类中的方法 effect，输入参数 heroAircraft，用 assertEquals 方法，判断英雄机此时的血量与预期血量是否相等。	140	140	通过

用例编号	9			
待测试类及方法	类名: BloodProp 方法: crash			
测试类及方法	类名: BloodPropTest 方法: crash			
前提条件（如有）	创建一个 BloodProp 类的对象，创建一个 HeroAircraft 类的对象 heroAircraft			
用例描述	测试步骤	期望结果	实际输出	测试结果
判断回血道具与英雄机能否碰撞正常。	设定道具和英雄机初始的位置是不会发生碰撞，道具移动一次后会发生碰撞。判断此时是否发生碰撞。之后调用道具的 forward 方法，道具移动之后判断此时是否会发生碰撞。	false True	false True	通过

2. JUnit 单元测试结果

EliteEnemyTest:

```

40
41 @RepeatedTest(10)
42 @DisplayName("Test EliteEnemyshoot method")
43 void shoot() {
44     System.out.println("****Begin to test shoot method ****");
45     List<BaseBullet> bullet = enemy.shoot();
46     assertEquals(bullet.isEmpty(), actual: false);
47     for(int size = bullet.size(), i=0; i<size; i++) {
48         assertEquals(bullet.get(i).getLocationX(), enemy.getLocationX());
49         assertEquals(bullet.get(i).getLocationY(), actual: enemy.getLocationY()+ 1*2);
50         assertEquals(bullet.get(i).getSpeedY(), actual: enemy.getSpeedY()+1*5);
51         assertEquals(bullet.get(i).getSpeedX(), actual: 0);
52     }

```

运行: EliteEnemyTest x

测试已通过: 30 共 30 个测试 - 507毫秒

测试结果 507毫秒

EliteEnemyTest 507毫秒

Test ElitegetProp metho 492毫秒

Test EliteEnemyshoot me 15毫秒

**** Executed before each test method in this class ****

****Begin to test getProp method ****

**** Executed After each test method in this class ****

**** Executed before each test method in this class ****

****Begin to test getProp method ****

**** Executed After each test method in this class ****

**** Executed before each test method in this class ****

****Begin to test getProp method ****

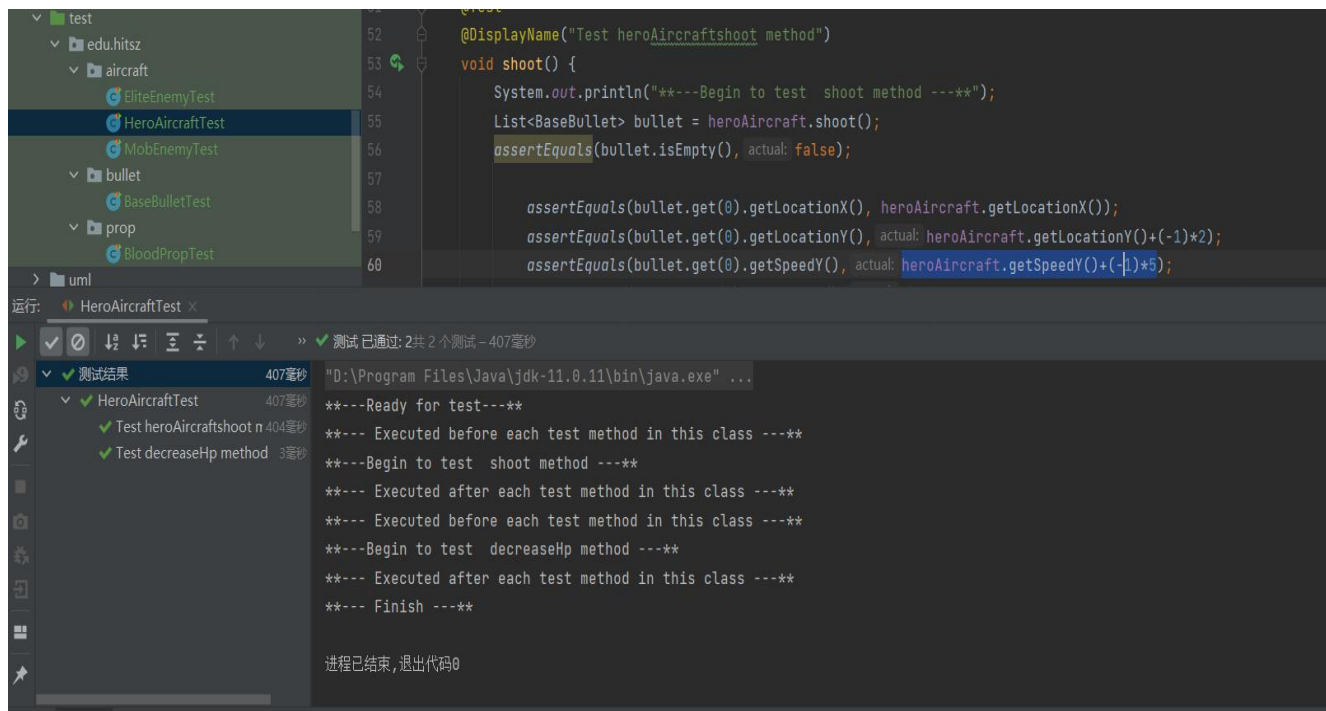
**** Executed After each test method in this class ****

**** Executed before each test method in this class ****

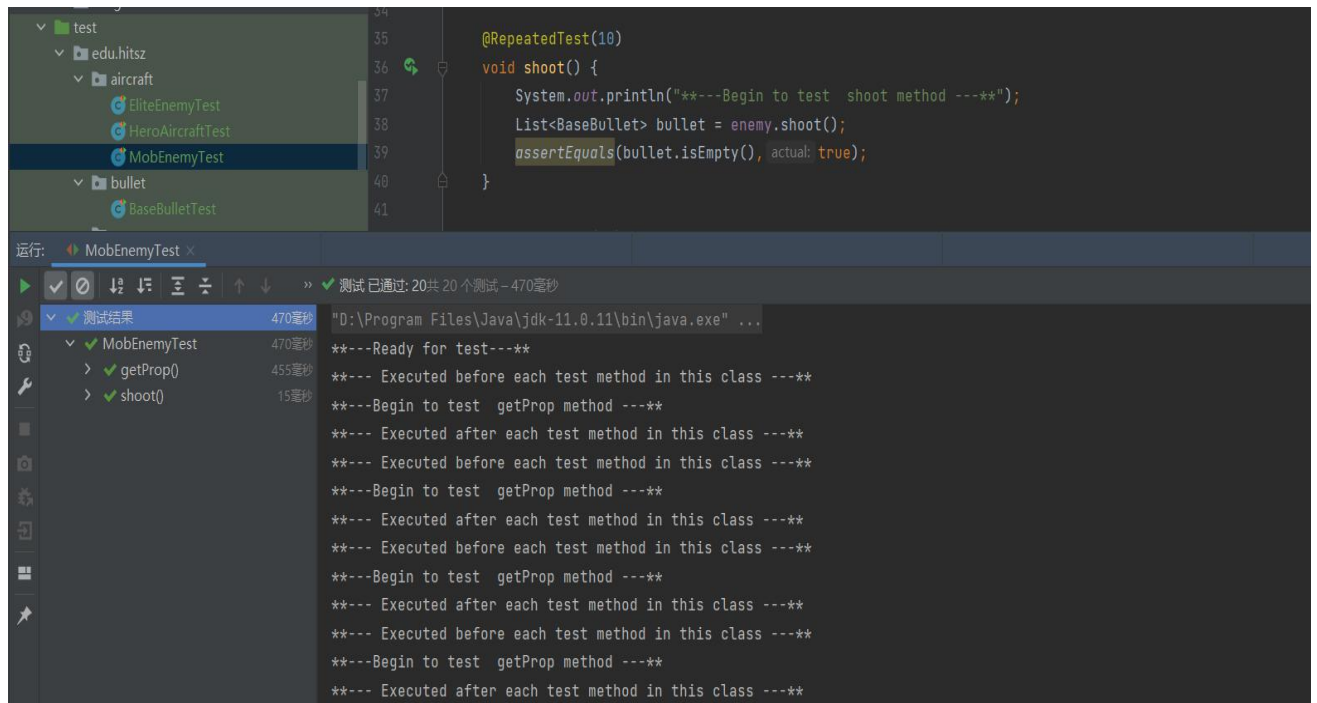
****Begin to test getProp method ****

**** Executed After each test method in this class ****

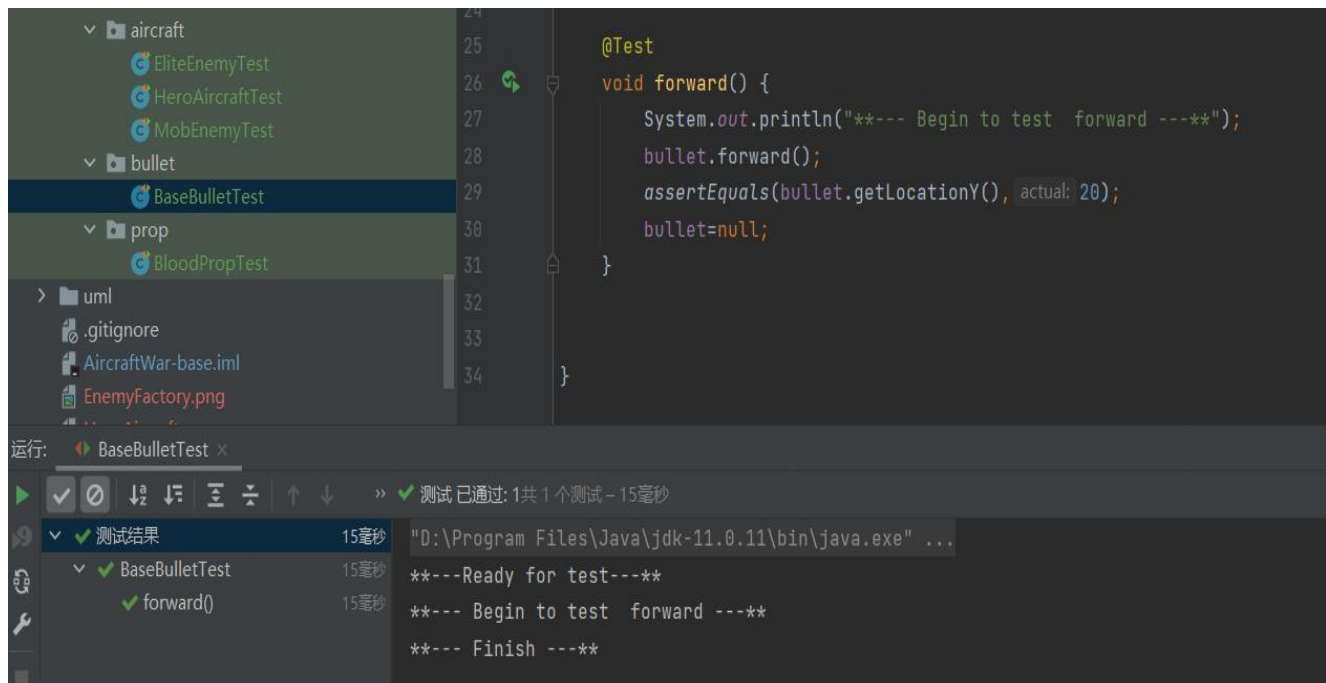
HeroAircraftTest:



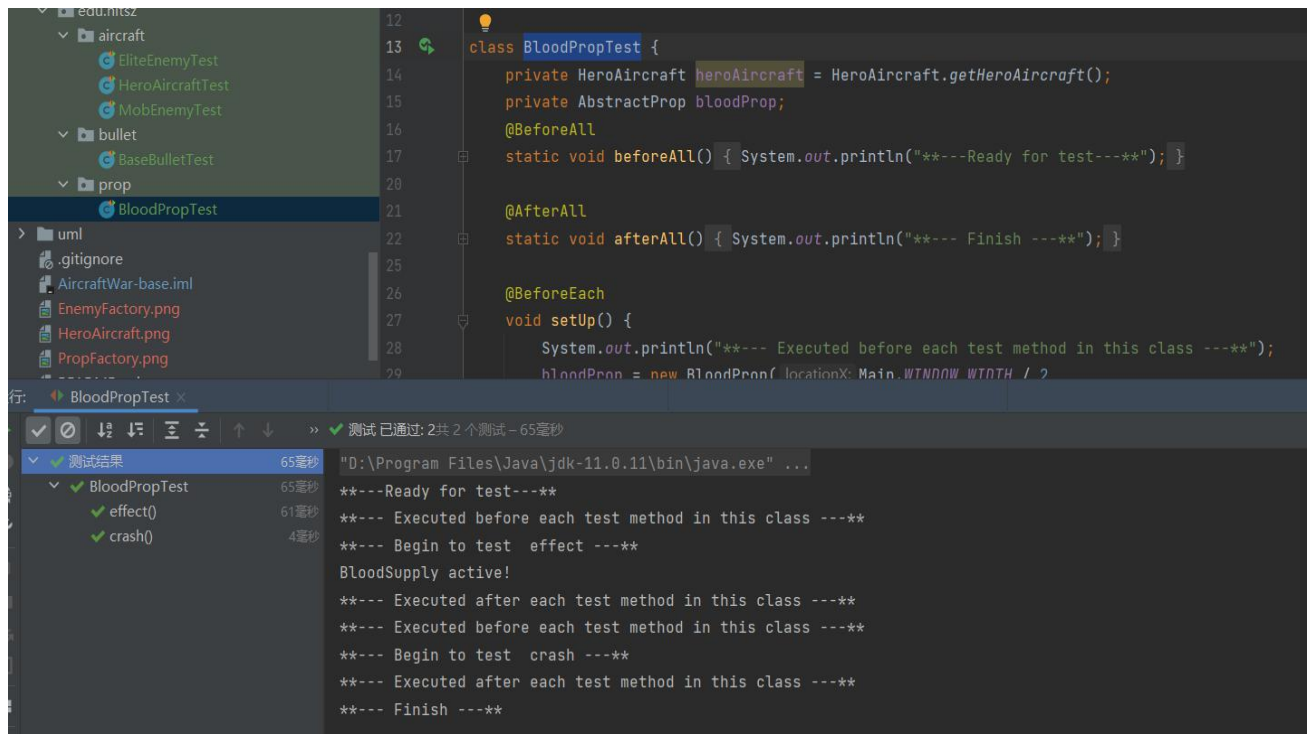
MobEnemyTest:



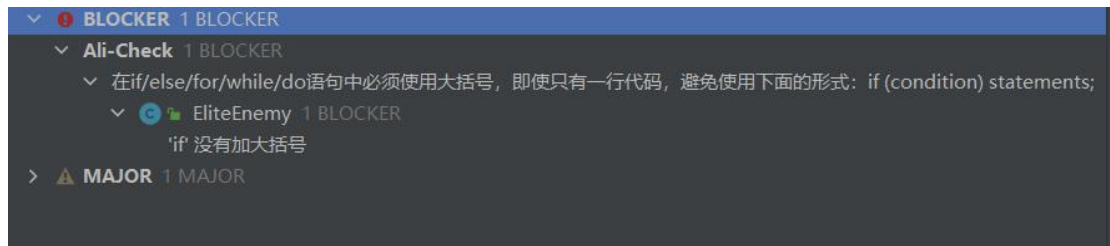
BaseBulletTest:



BloodPropTest:



二、编码规约



案例 1：用阿里编码规约插件扫描 EliteEnemy 类，if 后只有一行代码，没有加大括号，有较大风险。

修改：加上大括号。



案例 2：部分子类的方法，没有加@Override 注解

修改：补上注解。