实验三报告

*采用黑盒测试和白盒测试的常用方法，为飞机大战系统设计测试用例。在系统中选择英雄机、敌机、子弹和道具类的方法（包含其父类方法）作为单元测试的对象，为每个测试对象编写单元测试代码。要求至少选择3个类，每个类至少测试2个方法，并截图JUnit单元测试的结果。*

1. 测试用例*（一个方法一个测试用例）*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | S1 | | | |
| 待测试类及方法 | HealingSupply.getSpeedY | | | |
| 测试类及方法 | HealingSupplyTest.getSpeedY | | | |
| 前提条件（如有） | 创建一个随机坐标, Y轴速度为5的Healing道具 | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 一个随机坐标, Y轴速度为5的Healing道具 | 1. HealingSupply初始化, 指定随机坐标, 指定Y轴速度为5 2. 调用getSpeedY方法 3. 判断getSpeedY方法得到的Y轴速度是不是5 | getSpeedY方法返回5 | 5 | Pass |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | S2 | | | |
| 待测试类及方法 | HealingSupply.getHealingAmount | | | |
| 测试类及方法 | HealingSupplyTest.getHealingAmount | | | |
| 前提条件（如有） | 创建一个Healing道具 | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 一个随机坐标, Y轴速度为5,HealingAmount=500的Healing道具 | 1. HealingSupply初始化,设置坐标速度等 2. 调用getHealingAmount方法 3. 判断getHealingAmount方法得到的值是不是500 | getHealingAmount方法返回的值是500 | 500 | Pass |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | E1 | | | |
| 待测试类及方法 | EliteEnemy.forward | | | |
| 测试类及方法 | EliteEnemyTest.forward | | | |
| 前提条件（如有） | 创建一个EliteEnemy | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 一个随机坐标的eliteEnemy实例, 以及修改其Y坐标超出屏幕范围 | 1. EliteEnemy实例化为eliteEnemy 2. 调用setLocation方法, 将eliteEnemy的Y坐标设置为edu.hitsz.application.Main.WINDOW\_HEIGHT + 1 3. 调用方法forward 4. 判断eliteEnemy的isValid是否是False | eliteEnemy的isValid是False | False | Pass |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | E2 | | | |
| 待测试类及方法 | EliteEnemy.getScore\_award | | | |
| 测试类及方法 | EliteEnemyTest.getScore\_award | | | |
| 前提条件（如有） | 创建一个EliteEnemy | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 一个随机坐标的eliteEnemy实例, 其award是50 | 1. EliteEnemy实例化为eliteEnemy 2. 调用getScore\_award方法 3. 判断getScore\_award方法返回的是不是50 | getScore\_award方法返回的值是50 | 50 | Pass |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | H1 | | | |
| 待测试类及方法 | HeroAircraft.getHeroAircraftInstance | | | |
| 测试类及方法 | HeroAircraftTest.getHeroAircraftInstance | | | |
| 前提条件（如有） | 调用getHeroAircraftInstance来创建一个HeroAircraft类型的heroAircraft | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 对一个通过调用getHeroAircraftInstance创建出的个HeroAircraft类型的heroAircraft进行判断 | 1. 调用getHeroAircraftInstance来创建一个HeroAircraft类型的instance 2. 判断instance是否是NotNull 3. 判断heroAircraft是否和instance是同一个对象 | Instance NotNull, 并且heroAircraft和instance是同一个对象 | True  True | Pass |

6.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例编号 | H2 | | | |
| 待测试类及方法 | HeroAircraft.decreaseHp | | | |
| 测试类及方法 | HeroAircraftTest.decreaseHp | | | |
| 前提条件（如有） | 一个已经存在的HeroAircraft类型的heroAircraft | | | |
| 用例描述 | 测试步骤 | 期望结果 | 实际输出 | 测试结果 |
| 两组decrease和expected, 分别满足decreaseHp中的两组分支条件 | 1. 赋值decrease1=30,decrease2=-300 2. 先对heroAircraft调用decreaseHp(decrease1) 3. 判断此时英雄机的hp是否等于原先hp-decrease1 4. 再对heroAircraft调用decreaseHp(decrease2) 5. 判断此时英雄机的hp是否等于maxHp | 分别都等于 | True,  True | Pass |

1. JUnit单元测试结果

*请截图JUnit每个测试类（包含多个方法）的运行结果。*



 