**1B008 Homework 3**

**Junit测试自己写的BlackBoxDemo**

**Due on April 29th, 8:30, 2019**

（课前拷贝给老师）

作业描述:

**依据以下步骤生成测试用例设计：**

1. 输入范围(从命令行输入)检查：
   1. 整数类型检测，
   2. 下界小于等于上界；
   3. 不正确时提示重新输入；
2. 应用等价类划分技术，对于输入范围进行划分：返回一个有效区间，两个无效区间
3. 应用边界值分析技术，对于各区间去边界值，并返回边界值。注意返回的边界值里没有重复值
4. 应用随机数生成器对每个区间生成指定数量(从命令行输入)的随机值。注意返回的值里没有重复值
5. 主程序：调用功能1至4，生成一组测试用例
6. 用Junit测试第二次代码，做单元测试，写相关测试报告
7. 作业名称：姓名+学号+第3次作业

**Junit单元测试测试用例报告**

**测试时间：2019年4月21日18:27:15**

**测试人：吴宇**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **编号** | **m** | **n** | **v** | **num** | **测试说明** | **测试显示** | **正常预期** |
| 1 | 10 | 30 | 15 | 5 | 正常情况 | 4绿全过 | 4绿全过 |
| 2 | 10 | 5 | 8 | 5 | 下界>上界 | 4绿全过 | 4绿全过 |
| 3 | a | b | 5 | 5 | 非法字符输入 | 代码报错，4红 | 编译无法进行 |
| 4 | -10 | 30 | 20 | 50 | 随机数大于区间绝对值 | 4绿全过 | 4绿全过 |
| 5 | -10.1 | 30 | 20 | 5 | 有小数 | 代码报错，4红 | 编译无法进行 |
| 6 | -10 | 30 | -11 | 5 | 任意数超下区间 | 4绿全过 | 4绿全过 |
| 7 | -10 | 30 | -10 | 5 | 任意数为下区间 | 4绿全过 | 4绿全过 |
| 8 | -10 | 30 | 31 | 5 | 任意数超上区间 | 4绿全过 | 4绿全过 |
| 9 | -10 | 30 | 29 | 5 | 任意数为上区间 | 4绿全过 | 4绿全过 |

**说明：m为区间下界，n为区间上界，需要判断两个数值都是整型值，且下界小于等于上界方可符合要求，把整型区间划分为一个有效区间，两个无效区间；v为任意数，需要落在[m,n]区间方可符合要求；产生随机数个数，一定要小于等于区间绝对值，另外要保证数值不能重复，在无效区间也需要产生同样数目且不重复的随机数。**

**第一次测试**

**//m=10,n=30,v=15,num=5**

您输入的任意数15不在：[10,30]之间

有效类的区间边界值为: 下界--->10, 11, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界9, 上界右边界31

您输入数值有效！满足区间绝对值！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

有效数为：

8 10 12 4 13

左边无效成绩为：

-62309835 -1376146946 -222133444 -1763604031 -169189767

右边无效成绩为：

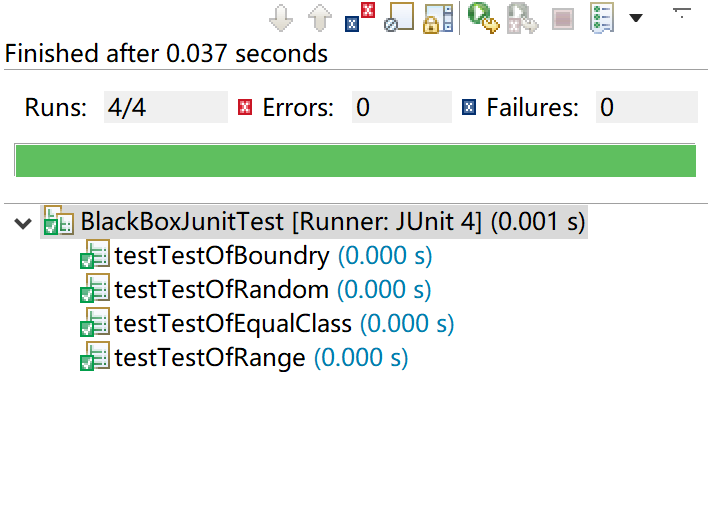
696651445 67492121 1770306800 426973612 2082062254 左边无效类为：[-2147483648,10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,10)

右边无效类为：(30,2147483647]

**说明：完全符合要求，全绿！**



**第二次测试**

**//m=10,n=5,v=8,num=5,**

有效类的区间边界值为: 下界5, 6, 上界9, 10

无效类的区间边界值(1)为：下界左4,上界右11

左无效类的边界值(2)为：2147483647<---下界溢出, -2147483648, -2147483647

右无效类的边界值(3)为：2147483646, 2147483647, 上界溢出--->-2147483648

您输入数值无效！越界！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

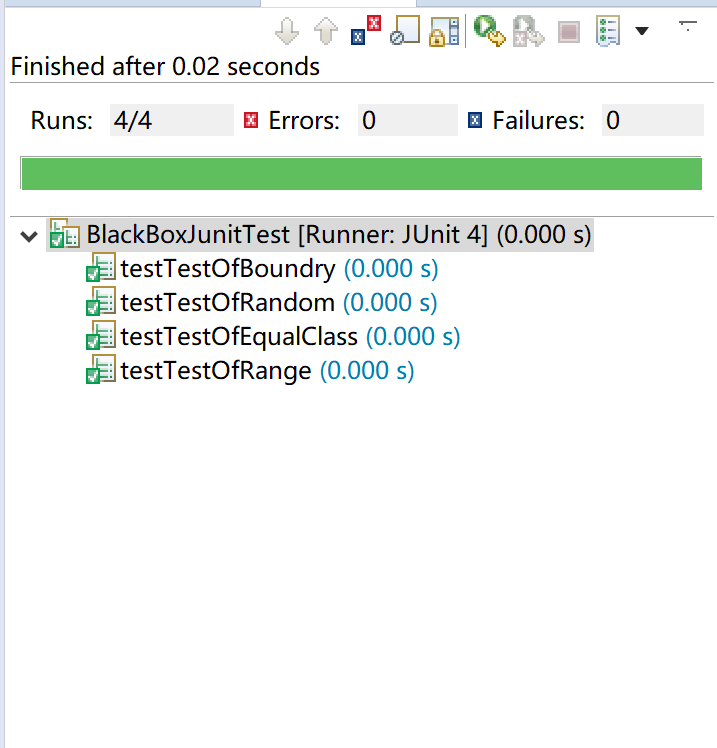
左边无效类为：[-2147483648,5)

右边无效类为：(10,2147483647]

左边无效类为：[-2147483648,5)

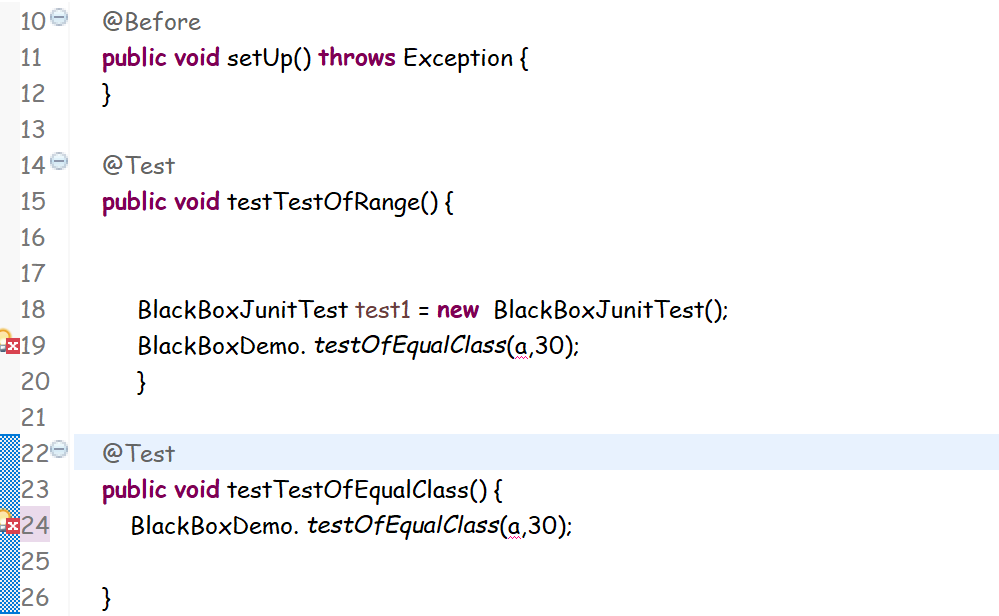
右边无效类为：(10,2147483647]

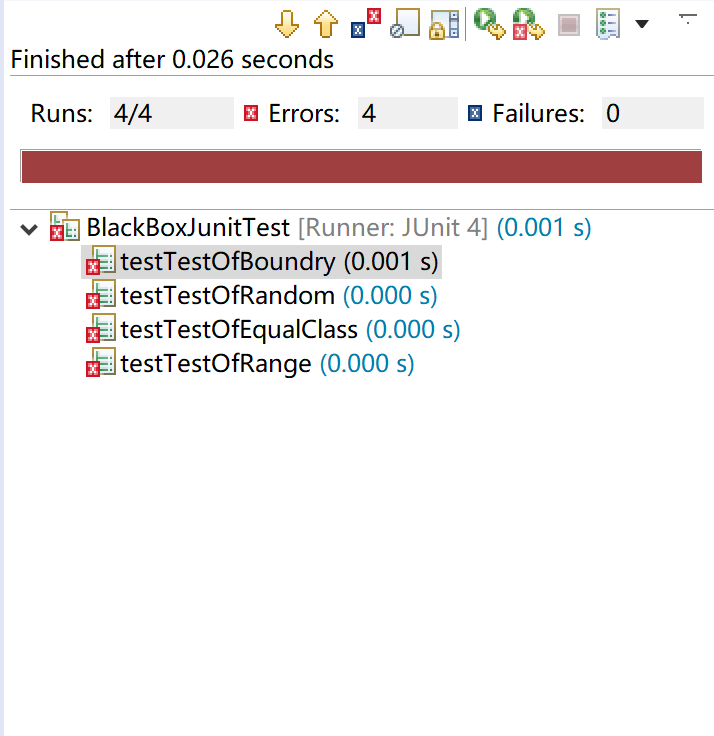
**说明：因为在代码中有纠正功能，即当n>m发生时，发生区间下界大于上界的exception，程序捕获异常，并自动逆转n和m，从而不需要用户再次输入数字，所以全绿！**



**第三次测试**

**//m=a,n=30,v=5,num=5,**





**说明：有字母，所以全红报错！**

**第四次测试**

**//m=-10,n=30,v=20,num=50,**

您输入的任意数20在：[-10,30]之间

有效类的区间边界值为: 下界--->-10, -9, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界-11, 上界右边界31

您输入数值无效！越界！

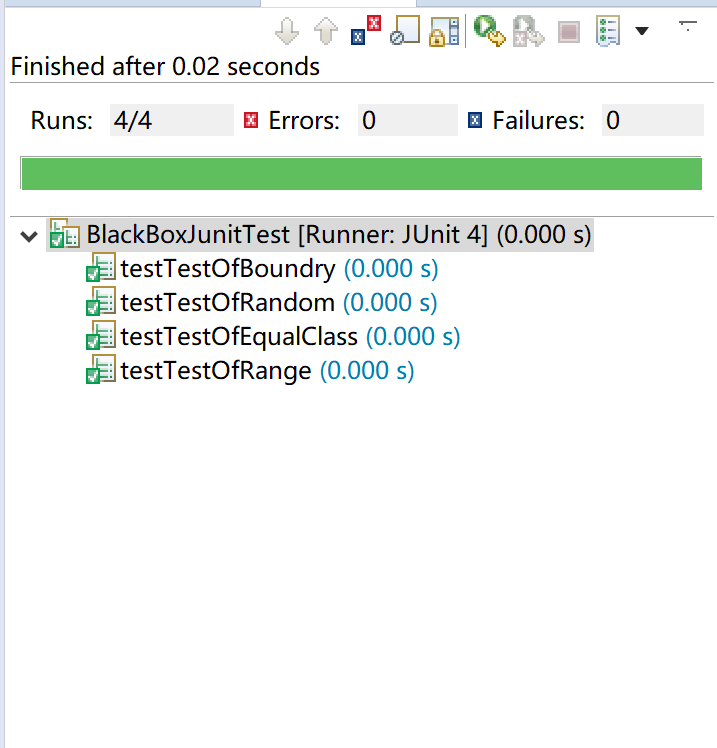
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

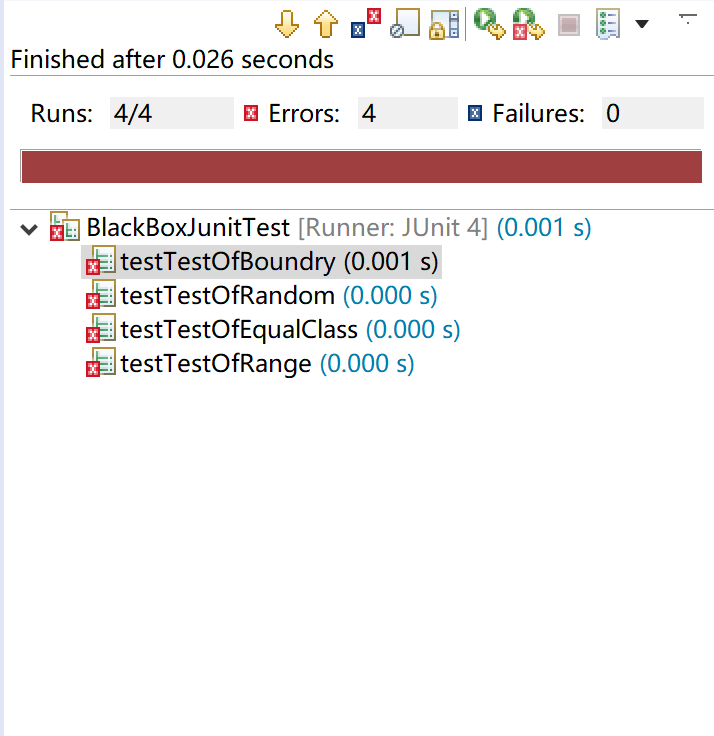


**说明：console提示随机数溢出报错信息，除了无法产生随机数，其他功能代码块正常！所以全绿！**

**第五次测试**

**//m=-10.1,n=30,v=20,num=5,**





**说明：有浮点型数值，所以全红！**

**第六次测试**

**//m=-10.1,n=30,v=-11,num=5,**

您输入的任意数-11不在：[-10,30]之间

有效类的区间边界值为: 下界--->-10, -9, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界-11, 上界右边界31

您输入数值有效！满足区间绝对值！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

有效数为：

8 26 37 31 39

左边无效成绩为：

-250675949 -1800253244 -1023242879 -1284700972 -370495501

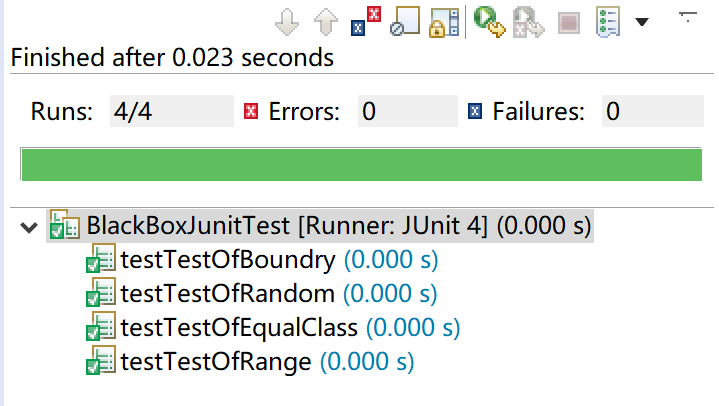
右边无效成绩为：

782570280 1373719203 629047197 68724203 548737325 左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]



**说明：正确判断-11超出下区间值，全绿！**

**第七次测试**

**//m=-10,n=30,v=-10,num=5,**

您输入的任意数-10在：[-10,30]之间

有效类的区间边界值为: 下界--->-10, -9, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界-11, 上界右边界31

您输入数值有效！满足区间绝对值！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

有效数为：

4 33 18 40 1

左边无效成绩为：

-1352175356 -1343156217 -1004733433 -713256532 -1129819335

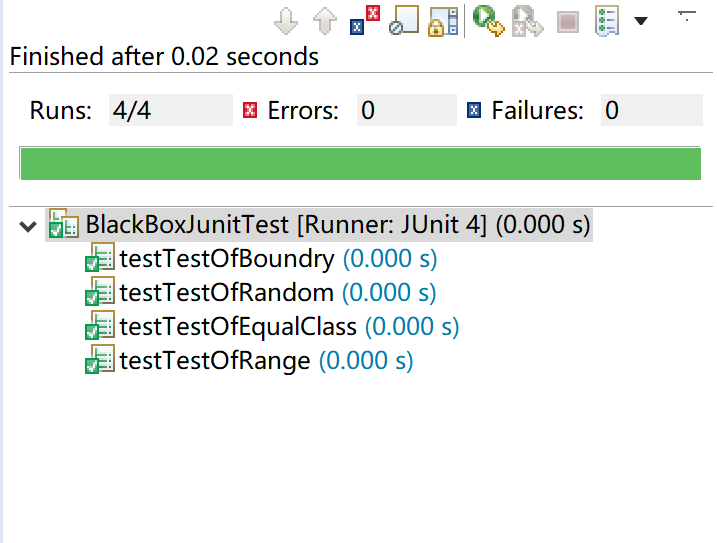
右边无效成绩为：

639422484 379907426 371311636 1516922160 803528920 左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]



**说明：正确判断-10恰好是下区间值，全绿！**

**第八次测试**

**//m=-10,n=30,v=31,num=5,**

您输入的任意数31不在：[-10,30]之间

有效类的区间边界值为: 下界--->-10, -9, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界-11, 上界右边界31

您输入数值有效！满足区间绝对值！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

有效数为：

1 36 18 32 28

左边无效成绩为：

-2063219189 -718838159 -909628818 -738224928 -635830171

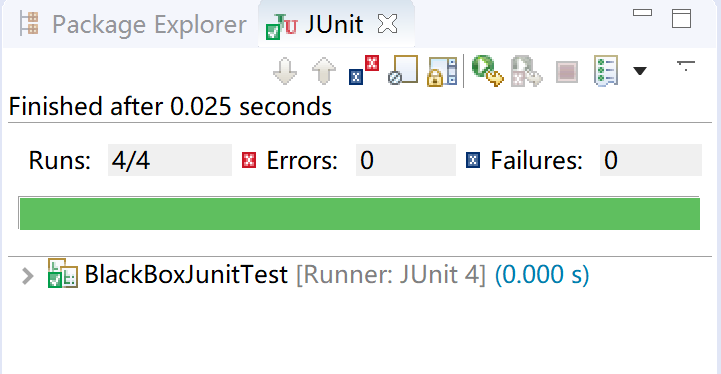
右边无效成绩为：

627335235 1339310895 727350625 965576634 679421438 左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]



**说明：正确判断31超出上区间值，全绿！**

**第九次测试**

**//m=-10,n=30,v=29,num=5,**

您输入的任意数29在：[-10,30]之间

有效类的区间边界值为: 下界--->-10, -9, 29, 30<---上界

无效类的区间边界值(1)为：下界左边界-11, 上界右边界31

您输入数值有效！满足区间绝对值！

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

有效数为：

24 12 21 16 32

左边无效成绩为：

-9749228 -1069015266 -359060282 -1477559121 -496471105

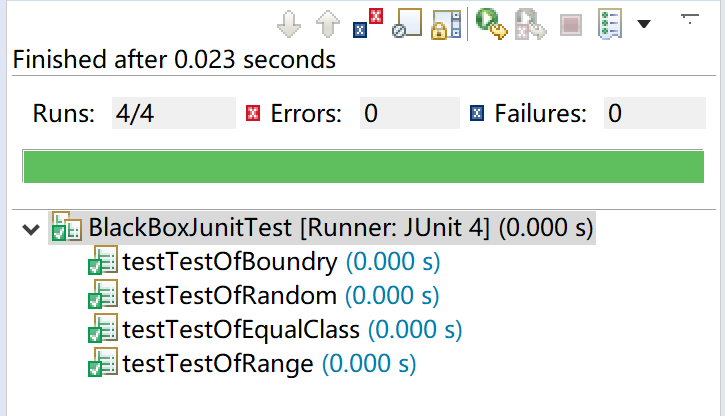
右边无效成绩为：

950190989 1504311396 288673405 943301343 247008840 左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]

左边无效类为：[-2147483648,-10)

右边无效类为：(30,2147483647]



**说明：正确判断30恰好是上区间值，全绿！**

**Junit某次测试代码：**

**import** **static** org.junit.Assert.\*;

**import** org.junit.Before;

**import** org.junit.Test;

/\*\*

\* Created by wuyupku on 2019年4月21日16:12:11

\*/

**public** **class** BlackBoxJunitTest {

@Before

**public** **void** setUp() **throws** Exception {

}

@Test

**public** **void** testTestOfRange() {

BlackBoxJunitTest test1 = **new** BlackBoxJunitTest();

BlackBoxDemo. *testOfEqualClass*(10,30);

}

@Test

**public** **void** testTestOfEqualClass() {

BlackBoxDemo. *testOfEqualClass*(10,30);

}

@Test

**public** **void** testTestOfBoundry() {

BlackBoxDemo.*testOfBoundry*(10,30,15);

}

@Test

**public** **void** testTestOfRandom() {

BlackBoxDemo. *testOfRandom*(10,30,5);

}

}