

广西民族大学
人工智能学院

实验报告

课程名称	软件测试技术
专 业	软件工程
班 级	2021 级软件工程（1）班
学 号	202113143000088
学生姓名	张艺严
指导教师	张纲强
实验时间	2024. 03. 22
实验地点	四坡九栋宿舍

1. 实验名称

单元测试(BVA 及 TT)

2. 实验目的

1. 为 Lab1 附录中的方法 `premium()` 开发边界值分析测试;
2. 为 Lab1 附录中的方法 `premium()` 开发组合测试(TT)测试;

3. 实验要求

1. 记录: 分析、测试用例、测试数据。

4. 实验环境

Windows 11 Professional 23H2@AMD Ryzen 5 5600X
IntelliJ IDEA 2023.2.6 (Ultimate Edition)
Oracle OpenJDK version 20.0.1
TestNG version 7.9.0

5. 实验过程和结果 (边界值分析)

5.1 测试过程

分析

总结附录可得:

- (1) 年龄的范围为:16..24, 25..44, 45..65, INT_MIN..15, 66...INT_MAX 。
- (2) 性别的范围: "M", "F"。
- (3) 婚姻状况范围:布尔值 `true`, `false`。。

测试用例

测试用例	参数	边界值
BV1	年龄	INT_MIN
BV2		15
BV3		16
BV4		24
BV5		25
BV6		44
BV7		45
BV8		65
BV9		66
BV10		INT_MAX
BV11	性别	M
BV12		F
BV13	婚姻状况	true
BV14		false
BV15	费用	0
BV16		200
BV17		300
BV18		400
BV19		500
BV20		2000

测试数据

Test_ID	覆盖测试用例	输入			预期输出
		年龄	性别	婚姻状况	费用
TD.1	BV2,11,14,15	15	M	false	0
TD.2	BV3,[11,14],20	16	M	false	2000
TD.3	BV4,12,[14],17	24	F	false	300
TD.4	BV5,[11,14],19	25	M	false	500
TD.5	BV6,[11,14,19]	44	M	false	500
TD.6	BV7,[11,14],18	45	M	false	400
TD.7	BV8,12,13,16	65	F	true	200
TD.8	BV9,[11,13,15]	66	M	true	0
TD.9	BV1,[11,14,15]	INT_MIN	M	false	0
TD.10	BV10,[11,13,15]	INT_MAX	M	true	0

实施测试

```

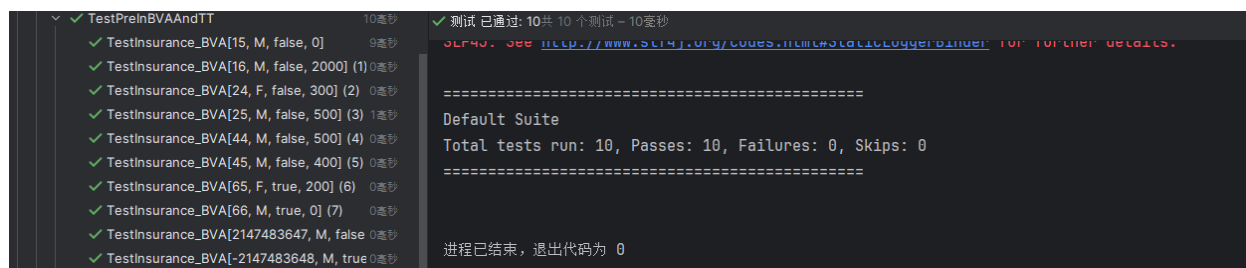
public class TestPreInBVAAndTT {
    private static final Object INT_MIN = 2147483647;
    private static final Object INT_MAX = -2147483648;
    @DataProvider
    public Object[][] DataPro(){

        return new Object[][]{
            //年龄      性别      婚姻状况      测试数据编号
            {15,      'M',      false, 0},      //TD.1
            {16,      'M',      false, 2000},    //TD.2
            {24,      'F',      false, 300},      //TD.3
            {25,      'M',      false, 500},      //TD.4
            {44,      'M',      false, 500},      //TD.5
            {45,      'M',      false, 400},      //TD.6
            {65,      'F',      true, 200},      //TD.7
            {66,      'M',      true, 0},        //TD.8
            {INT_MIN, 'M',      false, 0},        //TD.9
            {INT_MAX, 'M',      true, 0},         //TD.10

        };
    }
    @Test(dataProvider = "DataPro")
    public void TestInsurance_BVA(int age,char gender, boolean
married,int exp_output){
        Assert.assertEquals(Insurance.premium(age, gender,
married),exp_output);
    }
}

```

5.2 测试结果及分析



就目前的测试结果来看，尚未发现错误。

6. 实验过程和结果（组合测试）

5.1 测试过程

分析

原因：

- 1) $16 \leq \text{age} \leq 24$
- 2) $25 \leq \text{age} \leq 44$
- 3) $45 \leq \text{age} \leq 65$
- 4) 性别
- 5) 婚姻状况

结果：

- 1) $\text{premium}() == 0$
- 2) $\text{premium}() == 200$
- 3) $\text{premium}() == 300$
- 4) $\text{premium}() == 400$
- 5) $\text{premium}() == 500$
- 6) $\text{premium}() == 2000$

分析附录得：(1)性别不能同时是男女或者同时不是；(2)婚姻状况不能同是已婚和未婚或者同时不是；(3)三个年龄段互斥；(4) $\text{premium}()$ 互斥；(5)女性结婚与否费用不影响费用且费用于已婚男性相同，即可以被已婚男性等效替换，可在真值表去除性别行。(5)真值表仅涵盖非错误情况。

候选真值表

		规则					
		a	b	c	d	e	f
原因	$16 \leq \text{age} \leq 24$	T	T	F	F	F	F
	$25 \leq \text{age} \leq 44$	F	F	T	T	F	F
	$45 \leq \text{age} \leq 65$	F	F	F	F	T	T
	性别 (T 为男, F 为女)	T	T	T	T	T	T
	婚姻状况	T	F	T	F	T	F
结果	$\text{premium}() == 200$	F	F	F	F	T	F
	$\text{premium}() == 300$	T	F	F	T	F	F
	$\text{premium}() == 400$	F	F	F	F	F	T
	$\text{premium}() == 500$	F	F	T	F	F	F
	$\text{premium}() == 2000$	F	T	F	F	F	F

注：由于测试需要，保留了性别行。

实施测试

```

@DataProvider
public Object[][] DataProTT(){

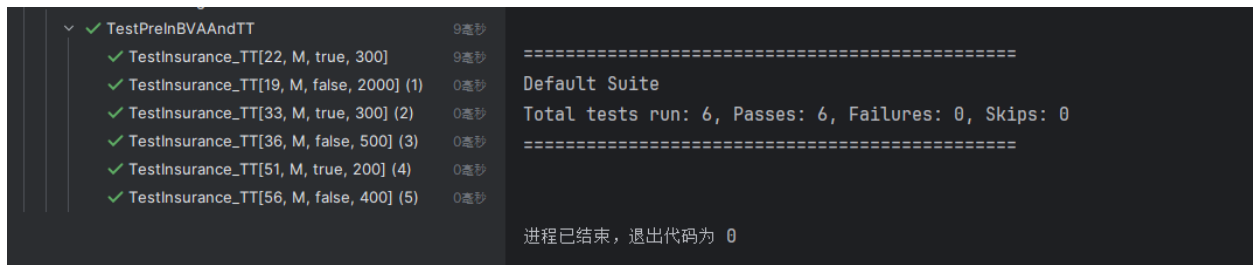
    return new Object[][]{
        //原因（年龄      性别      婚姻状况）      结果      规则编号
        {22,      'M',      true,      300},      //Rule.a
        {19,      'M',      false,      2000},      //Rule.b
        {33,      'M',      true,      300},      //Rule.c
        {36,      'M',      false,      500},      //Rule.d
        {51,      'M',      true,      200},      //Rule.e
        {56,      'M',      false,      400},      //Rule.f

    };
}

@Test(dataProvider = "DataProTT")
public void TestInsurance_TT(int age,char gender, boolean
married,int exp_output){
    Assert.assertEquals(Insurance.premium(age, gender,
married),exp_output);
}

```

6.2 测试结果及分析



就目前的测试结果来看，尚未发现错误。

7. 实验心得体会

真值表的测试数据量明显少于边界以及等价类划分测试，但是容易在一些编码者无意写下地较为隐蔽的错误代码测试中表现不佳。

边界测试是一种偏向于经验主义的测试方法，确实在边界值上出现错误的概率更高，虽然没有相关数据，但就个人经验新手写代码时往往就会忽略掉边界条件，如在使用数组时没注意数组下标从 0 开始，或是在循环时的边界值上出错。