实验课程名称：软件质量保证与测试B

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验项目名称 | 黑盒测试（Black-box testing） | | | 实验成绩 |  |
| 实验者姓名 |  | 专业班级 |  | 组别 | / |
| 同组者 | / | | | 实验日期 |  |
| 第一部分 实验内容概述  **一、实验目的**   1. 掌握黑盒测试分析和生成测试用例的方法； 2. 掌握边界值分析方法产生测试集。   **二、实验内容**  **2.1** **问题描述：三角形问题**  在一个程序中，依次输入三个整数 a，b，c 作为三角形的三条边。三边不同的关系，输出三角形的类型：Equilateral（等边的）、Isosceles（等腰的）、Scalene（三边不等的三角形)或者 Not A Triangle（非三角形）。这里暂时不考虑直角三角形。  判断过程的流程图如图 1 所示。    图 1 传统三角问题的解决方法描述  图中有明显错误（14，17，19判断分支错误）：Match=1?的Y分支里a+b≤c?判断，若a+b≤c不为三角形，否则为等腰三角形。 | | | | | |

## 2.2 问题描述的改进

基于前述的三角形问题，继续增加如下限制，量化问题描述进行改进。

C1: 1 ≤ a ≤ 200

C2: 1 ≤ b ≤ 200

C3: 1 ≤ c ≤ 200 C4: a < b + c

C5: b < a + c C6: c < a + b

如果输入整数值范围超出 C1, C2, C3指定，则输出提示信息“Value of b is not in the range

of permitted values.”（以 C2 不满足为例，其它类似）。

当输入a,b,c满足 C4, C5, C6指定后，再判断三角形类型（Equilateral / Isosceles / Scalene

/ Not A Triangle）。

判断过程三角形的流程图如图 2 所示。

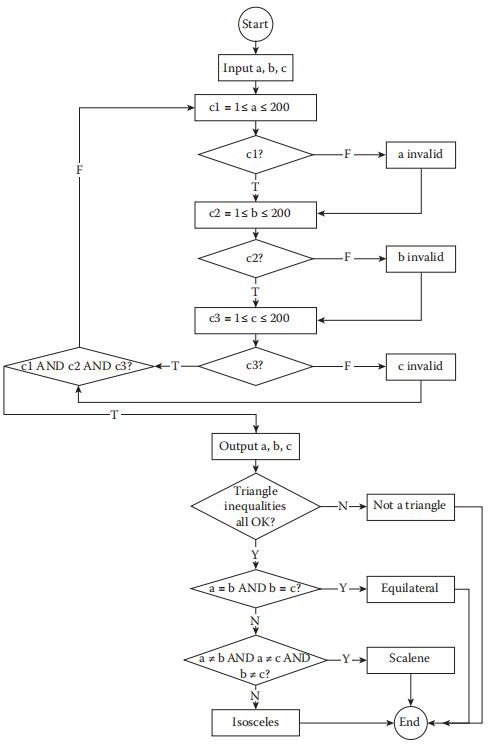


图 2 改进的三角问题解决方法描述

## 三、实验要求

* 1. **编程任务**

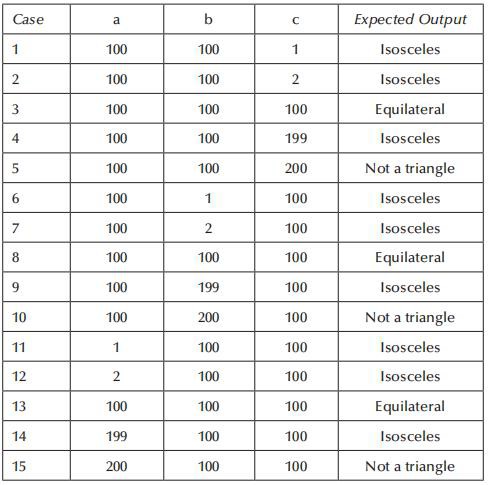
根据图 1 和图 2 描述的流程图描述，分别进行编程实现。

## 边界值分析测试实验

通过边界值分析方法生成测试集，不断地修改程序，直到所有测试用例集测试均通过。 测试用例集的生成分别基于普通边界值（normal boundary value）测试和最坏情形边界值

（worst-case boundary value）测试两种情况进行。例如，现已给出如下普通边界值测试用例表 1。

表 1 普通边界值测试用例



继续设计最坏情形用例边界值，并将两种测试集实现，用于程序的测试。

**四、实验原理**

## 边界值分析测试

边界值分析测试分为四类：普通边界值（normal boundary value）测试，最坏情形边界值（worst-case boundary value）测试，健壮性边界值（robust boundary value）测试，健壮性最坏情形边界值（robust worst-case boundary value）测试。在二维测试用例空间中，四种类型的测试用例分布如下图 3 至图 6 所示。

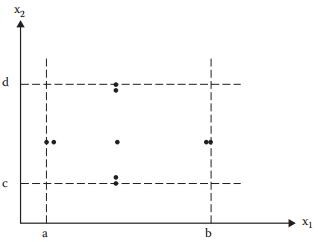


图 3 普通边界值测试

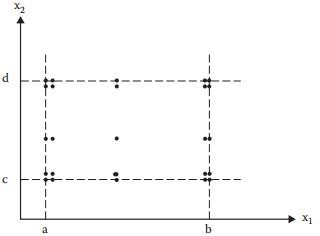


图 4 最坏情形边界值测试

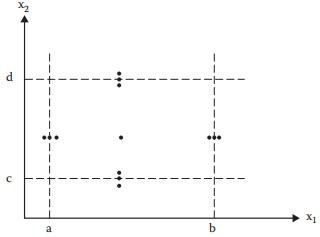


图 5 健壮性边界值测试

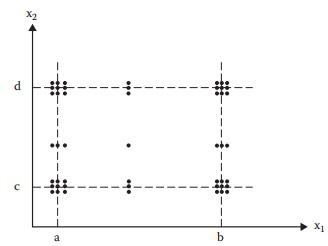
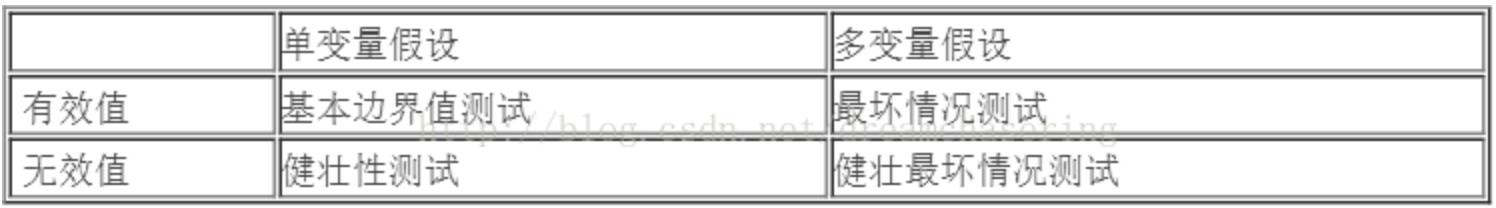


图 6 健壮性最坏情形边界值测试

对于这四种测试，它们的关系可以用下表 4 来描述。

表 4 四种边界分析测试的关系



健壮性是指在异常情况下，软件还能正常运行的能力。

单缺陷假设指“失效极少是由两个或两个以上的缺陷同时发生引起的”。在边界值分析 中，单缺陷假设即选取测试用例时仅仅使得一个变量取极值，其他变量均取正常值。

多缺陷假设，则是指“失效是由两个或两个以上缺陷同时作用引起的”，要求在选取测试用例时同时让多个变量取极值。

**五、主要仪器设备及依赖**

1. PC 机；
2. 编程语言：Java（jdk-14）；
3. 集成开发环境：IDEA；
4. 绘图软件：Visio 2019。