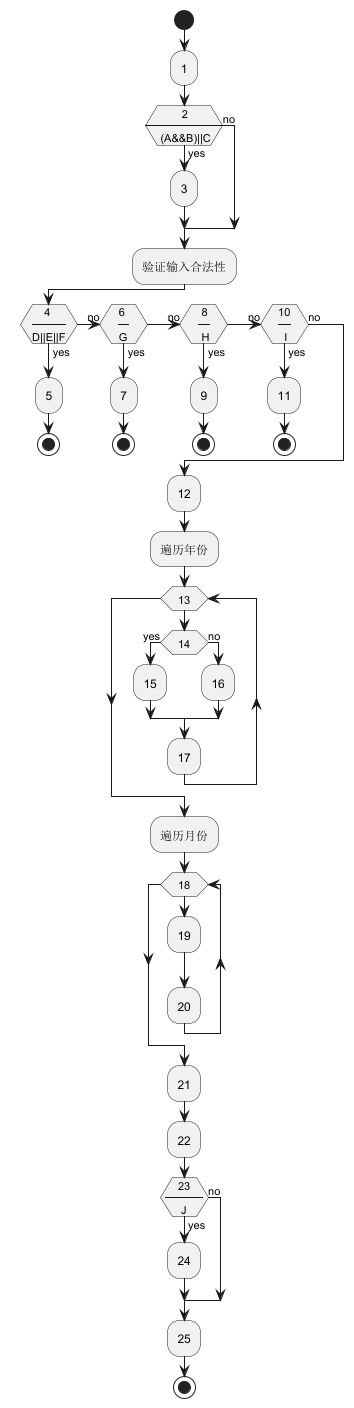
实验一 白盒测试实验一

1. 实验内容
2. 给出 getWeekday()的程序流程图，这是进行基于控制流动态白盒测试的基础。

注意：

1. 这是一个参考示例，答案并不固定。同学们可以自由绘制自己的流程图。如果想要借鉴该示例，请补全图中的各语句。
2. 为便于作业的批改以及执行路径的书写，请同学们在流程图中对各语句和条件进行标记，或在答案中体现。



**（二）分别以语句覆盖和判定覆盖方法设计测试用例，并写出每个测试用例的执行路径。**

**注意：循环语句统计时可简化为：执行 N>=1 次视为同一条路径（内部语句默认全部执行），执行 0 次（即跳过）视为另一条路径；**

**若测试用例过多，难以写全，每种方法可只写8条。**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 执行路径 |
| 001 | 语句覆盖 | 2021/2/14 | 7 | 7 | 1-2-3-4-6-8-10-12-13-13-18-18-21-22-23-24-25 |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 判定取值 | | | | | | 执行路径 |
| 2 | 4 | 6 | 8 | 10 | 23 |
| 001 | 判定覆盖 | 2021/2/14 | 7 | 7 | T | F | F | F | F | T | 1-2-3-4-6-8-10-12-13-13-18-18-21-22-23-24-25 |
| 002 | 判定覆盖 | 1/1/1 | 1 | 1 | F | F | F | F | F | F | 1-2-4-6-8-10-12-13-18-21-22-23-25 |

**（三）分别以条件覆盖、判定条件覆盖和条件组合覆盖方法设计测试用例，并写出每个测试用例的执行路径。**

**注意：循环语句内部的条件暂不考虑；**

**因条件数较多，为便于书写，可只写出为真的条件；**

**若测试用例过多，难以写全，每种方法可只写8条。**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 条件取值（为真） | 执行路径 |
| 001 | 条件覆盖 | 1/1/7 | 7 | 7 | J | 1-2-4-6-8-10-12-13-18-21-22-23-24-25 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 条件取值（为真） | 判定取值 | | | | | | 执行路径 |
| 2 | 4 | 6 | 8 | 10 | 23 |
| 001 | 判定条件覆盖 |  |  |  |  |  |  |  |  |  |  |  |

**（四）给出对程序中循环的测试用例，并给出测试用例设计的理由。**

**提示：可以根据循环执行的次数来设计**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 执行路径 |
| 001 |  |  |  |  |  |

**（五）如果要进一步用路径覆盖准则来测试 getWeekday()，请基于程序流程图计算其中可能的路径共有多少条？是否包含不可达路径？**

**（六）给出 MC/DC(修订的判定条件覆盖)方法对下列 2 处语句的测试用例**

**注意：这两处语句二选一即可，只用写一个**

**1、语句1**

1. **int**[] MonthDays = { 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
2. **if** (((year % 4 == 0) && (year % 100 != 0)) || year % 400 == 0) {
3. MonthDays[1] = 29;
4. }

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 条件1 | 条件2 | 条件3 | 判定 |
|  |  |  |  |  |  |  |  |  |

**2、语句2**

1. **if** (((i % 4 == 0) && (i % 100 != 0)) || i % 400 == 0) {
2. totalDays += 366;
3. } **else** {
4. totalDays += 365;
5. }

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 执行条件 | 输入  year/month/day | 期望输出 | 实际输出 | 条件1 | 条件2 | 条件3 | 判定 |
|  |  |  |  |  |  |  |  |  |

1. **实验思考**

**（1）在存在逻辑短路的情况下，请分析条件覆盖、判定条件覆盖、条件组合覆盖、 MC/DC 覆盖的联系和区别。**

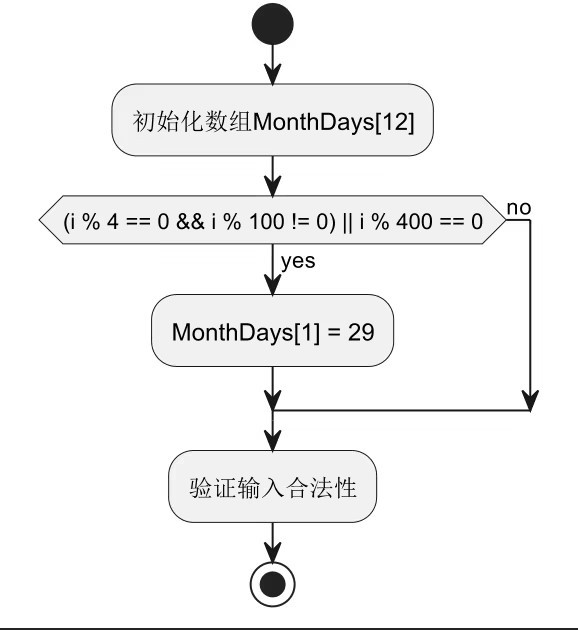
**（2）在白盒测试实践中，你觉得哪些步骤是关键的？并谈谈你所体会到的白盒测试过程中的难点。**

**（3）通过测试，是否发现程序中存在的缺陷？可以如何改进？**

**问题解答**

* 1. Q:在利用plantuml画出来的程序流程图if的结束位置会有空框。如下左图

A：因为判断语句的no分支没有进行操作，所以会产生一个空的方框，对整体流程没有影响。如果想要改成下右这张图的形式，可以考虑在代码开头增加一行代码skinparam ConditionEndStyle hline，改成水平线样式，具体可查阅指南6.24.2



* 1. Q:plantuml代码中的“:”后面是什么意思，说是写要执行的活动，但是指导书里面的图把注释也写在了 “:”的后面，然后下面接着写真正做的操作（感觉有点重复？）

A:这个冒号就是既可以写注释，又可以写活动的。因为有时候流程图可能比较复杂，这个就是起到一个解释后面流程的作用。指导书里面只是一个简单的示例，不一定按照这个写，如果觉得麻烦的话，可以删掉的。

* 1. Q:怎么在plantuml里面添加示例中所示上下部分的序号

A:可以直接用换行符\n手动换行，比如“6\n \nG”

* 1. Q:流程图的示意图里面，圆角矩形是不是用得不太对，好像有特定含义

A:示例流程图是用工具自动生成的，可能确实和一般的流程图不太一样。只要程序流程没有问题，方框

的画法不太重要，可以自由发挥的。