<公历转农历工具>

测试评估报告

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <日/月/年> | <x.x> | <详细信息> | <姓名> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1. 简介 4](#_Toc40637166)

[1.1 目的 4](#_Toc40637167)

[1.2 范围 4](#_Toc40637168)

[1.3 定义、首字母缩写词和缩略语 4](#_Toc40637169)

[1.4 参考资料 4](#_Toc40637170)

[1.5 概述 4](#_Toc40637171)

[2. 测试结果摘要 4](#_Toc40637172)

[3. 测试环境 4](#_Toc40637173)

[4. 测试结果及分析 4](#_Toc40637174)

[4.1 覆盖率及缺陷分布 4](#_Toc40637175)

[4.2 缺陷严重程度 5](#_Toc40637176)

[5. 缺陷清单 5](#_Toc40637177)

[6. 测试结论与建议 5](#_Toc40637178)

测试评估报告

# 简介

## 目的

该“测试计划”文档有助于实现以下目标:

（1) 为软件可靠性与安全性的评估提供依据

（2) 提供系统化、规范化、工程化、实用化的测试技术规范，尽早发现故障

（3) 检查软件对误操作的处理能力

（4) 验证软件是否满足软件设计所规定的技术要求

## 范围

测试工作包括：

（1) 白盒测试

## 定义、首字母缩写词和缩略语

## 参考资料

《测试计划》

## 概述

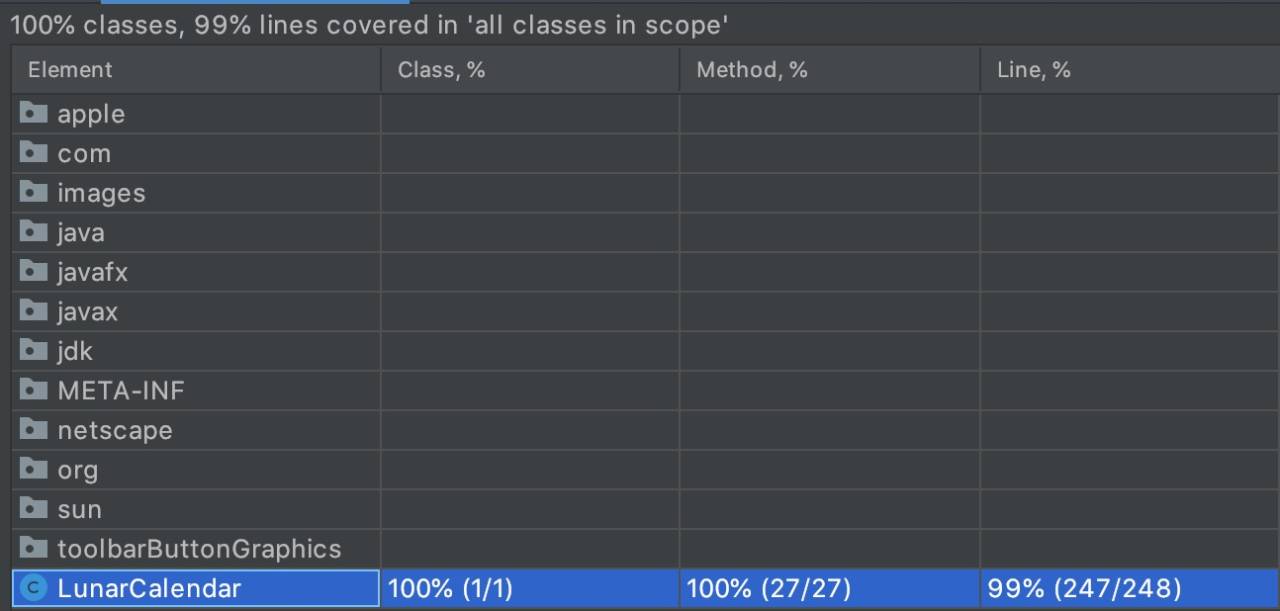
本测试文档以测试为核心，从测试概要开始，接着列出测试的环境，然后较详细的列出了测试的结果，之后罗列出测试中发现的缺陷，最后得出了本次测试的结论与建议。

# 测试结果摘要

本次测试覆盖率到达了99%，测试用例数共有149个，通过测试149个，未通过测试0个，通过率为100%。（参数化测试的每一个参数都算作一个测试用例）

# 测试环境

# 测试结果及分析



## 覆盖率及缺陷分布

**表1 覆盖率及缺陷分布**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **测试内容** | | **测试**  **用例数** | **缺陷数** | **缺陷率** | **备注** |
| **方法** | **LunarCalendar.addDays** | 17 | 0 | 0% |  |
| **LunarCalendar.isLeapYear** | 6 | 0 | 0% |  |
| **LunarCalendar.getDays** | 4 | 0 | 0% |  |
| **LunarCalendar.formatYear** | 5 | 0 | 0% |  |
| **LunarCalendar.formatMonth** | 14 | 0 | 0% |  |
| **LunarCalendar.formatDay** | 12 | 0 | 0% |  |
| **LunarCalendar.cast** | 3 | 0 | 0% |  |
| **LunarCalendar.addZero** | 4 | 0 | 0% |  |
| **LunarCalendar.getDate** | 12 | 0 | 0% |  |
| **LunarCalendar.** **judge** | 45 | 0 | 0% |  |
| **LunarCalendar.read** | 6 | 0 | 0% |  |
|  | **总计** | 149 | 0 | 0% |  |

## 缺陷严重程度

**表2 缺陷严重程度**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **严重程度** | **致命** | **严重** | **一般** | **细微** | **总数** |
| **缺陷个数** | 0 | 0 | 0 | 0 | 0 |
| **占缺陷百分比** | 0 | 0 | 0 | 0 | 0 |

# 缺陷清单

**表3 缺陷列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序**  **号** | **缺陷**  **编号** | **严重程度** | **方法** | **缺陷描述** |
| 1 |  |  |  |  |

表中相关项说明：

1. 缺陷编号：为每个缺陷赋予一个唯一的编号，可以通过此编号对缺陷进行跟踪。例如：Bug001。
2. 缺陷严重程度：缺陷可以根据严重程度分为以下几种情况。
3. 致命（fatal）：致命的错误，测试执行直接导致系统死机、蓝屏、挂起、或是程序非法退出；系统的主要功能或需求没有实现。
4. 严重（critical）：严重错误，系统的次要功能点或需求点没有实现；数据丢失或损坏。执行软件主要功能的测试用例导致系统出错，程序无法正常继续执行；程序执行过于缓慢或是占用过大的系统资源。
5. 一般（major）：不太严重的错误，这样的缺陷虽然不影响系统的基本使用，但没有很好地实现功能，没有达到预期的效果。如次要功能丧失，界面错误，打印内容、格式错误，提示信息不太正确，或用户界面太差，简单的输入限制未放在前台进行控制，删除操作未给出提示，操作时间长等。
6. 细微（minor）：一些小问题，对功能几乎没有影响，产品及属性仍可使用。如软件的实际执行过程与需求有较小的差异；程序的提示信息描述容易使用户产生混淆；有个别错别字、 文字排列不整齐；辅助说明描述不清楚，显示格式不规范，长时间操作未给用户进度提示，提示窗口文字未采用行业术语，可输入区域和只读区域没有明显的区分标志，系统处理未优化等。
7. 方法：所测试并出现该缺陷的方法名称。
8. 缺陷描述：对缺陷的详细描述。描述缺陷的重现步骤和问题所在，描述需清晰明了，使研发人员可以按照描述将缺陷重现出来。必要时可以放上相应的截图，以便于研发人员理解问题所在。

# 测试结论与建议

## 结论

在测试过程中，公历转农历工具的各个方法都没有表现出缺陷来，因此软件整体可能出现错误的概率是比较小的，该软件的条件处理设计的非常好。

## 建议

在本次测试过程中，我们由于技术限制无法对IOException类进行单元测试，对于这方面还需要加强学习。