**软件测试报告**

# 引言

本测试报告为文件复制加密系统的测试报告，目的在于总结测试阶段的测试情况以及分析测试结果，描述系统是否符合需求，作为测试质量参考文档提供给用户、测试人员、开发人员、项目管理者、其他质量管理人员和需要阅读本报告的高级管理人员阅读。

## 目的

为了尽可能的找出软件的不足，提高软件的质量，促进软件的成功验收，专门制定了本大纲。其主要目的在于为所要进行的测试工作制定各种必要的准则和规范，以及在有关方面协议的基础上对测试工作进行合理组织与管理。

## 术语

本大纲所提及的术语，其定义遵照GB/T 11457标准。

## 参照标准

* GB/T 11457—1995

软件工程术语

* GB 8566—1995；

信息技术软件生存期过程

* OGB 8567—1988\*

计算机软件产品开发文件编制指南

* GB 9385\*

计算机软件需求说明编制指南

* GB 9386—1988\*

计算机软件测试文件编制指南

* GB/T 12504—1990

计算机软件质量保证计划规范

* OGB/T 12505—1990

计算机软件配置管理计划规范

* OGB/T 14079—1993

软件维护指南

* OGB/T 14394—1993

计算机软件可靠性和可维护性管理

* GB/T 16680一1996

软件文档管理指南

* 开发者企业规范

软件开发者有关软件工程的规范

* 其它文件

例如：合同书等，法律文件中的有关规定。

说明：（1）应该遵循自顶而下、就严不就宽的原则，除非合同书等法律文件中另有规定。

（2）标记（\*）号的标准为推荐标准。

# 测试日期安排

测试时间为10月1日至11月1日。

# 测试小组及成员

测试小组成员是张天茗、彭易涛、陈元起。

# 测试具体内容

测试内容包括：合法性检查、文档检查、软件一致性检查、软件系统测试与测试结果评审等几项工作。

## 合法性检查

检查开发者在开发本软件时，使用的开发工具是否合法。对在编程中使用的一些非本单位自己开发的，也不是由开发工具提供的控件、组件、函数库等，检查其是否有合法的发布许可。

## 软件文档检查

### 必须提供检查的文档

* 项目实施计划；
* 系统设计方案；
* 软件需求规格说明书(STP)；
* 软件测试计划(STP)(含测试用例)；
* 软件测试报告(STR)；
* 源程序(SCL)(不可修改的电子文档)；
* 项目实施计划(PIP)；
* 项目开发总结(PDS)；

### 其他可能需要检查的文档

无

### 文档质量的度量准则

文档是软件的重要组成都分，是软件生存周期各个不同阶段的产品描述。文档质量的度量准则就是要评审各阶段文档的合适性。主要有以下六条：

* 完备性

开发方按照GB 8567(计算机软件产品开发文件编制指南)的规定编制相应的文档，保证了在开发阶段结束时其文档是齐全的。

* 正确性

在软件开发各个阶段所编写的文档的内容，必须真实的反映阶段的工作且与该阶段的需求相一致。

* 简明性

在软件开发各个阶段所编写的各种文档的语言表达应该清晰、准确简练，适合各种文档的特定读者。

* 可追踪性

在软件开发各个阶段所编写的各种文档应该具有良好的可追踪性。文档的可追踪性包括横向可追踪性和纵向可追踪性两个方面。前者是指在不同的文档的相关内容之间相互检索的难易程序；后者是指确定同一文档某一内容在本文档范围中检索的难易程度。

* 自说明性

在软件开发各个阶段所编写的各种文档应该具有较好的自说明性。文档的自说明性是指在软件开发各个阶段中，不同文档能够独立表达，该软件在其相应阶段的阶段成果的能力。

* 规范性

在软件开发各个阶段所编写的各种文档应该具有良好的规范性。文档的规范性是指文档的封面、大纲、术语的含义以及图示符号等符合有关规范的规定。

对本软件相关文档进行检查度量，文档完备性、正确性、简明性、可追踪性、自说明性以及规范性良好。

## 软件代码测试

### 源代码一般性检查

仅对系统关键模块的源代码进行抽查，检查模块代码编写的规范性，批注的准确性，是否存在潜在性错误，以及代码的可维护性。

* 命名规范检查

检查源代码中的变量、函数、对象、过程等的命名符合约定规范。

* 注释检查

检查程序中的注释是否规范，注释量达到约定要求30%。

* 数据类型检查

路径检查表：

|  |  |  |
| --- | --- | --- |
| 检查项 | | 结论 |
| 数据种类问题 | 种类的数据类型有错误吗 | 无 |
| 存在不同的数据类型赋值吗？ | 有 |
| 存在不同的数据类型种类的比较吗？ | 无 |
| 变量值问题 | 变量的初始化或缺省值有错误吗？ | 无 |
| 变量发生上溢或下溢吗？ | 发生 |
| 变量的精度不够吗？ | 够 |
| 逻辑判断问题 | 由于精度原因以致比较无效吗？ | 无 |
| 表达式中的优先级有误吗？ | 有 |
| 逻辑判断结果颠倒吗 | 有 |
| 循环问题 | 循环停止条件不正确吗？ | 正确 |
| 无法正常停止（死循环）吗？ | 是 |
| 错误地更正循环变量吗？ | 否 |
| 存在误差累积吗？ | 无 |
| 内存问题 | 内存没有被正确的初始化却被使用吗 | 无 |
| 内存被释放后却连续被使用吗？ | 无 |
| 内存泄露吗？ | 不泄露 |
| 内存越界吗？ | 未越界 |
| 出现指针越界吗？ | 未出现 |
| 文件I/O问题 | 对不存在的也许错误的文件进行操作吗？ | 无 |
| 文件以不正确的方式打开吗？ | 无 |
| 文件结束判断不正确吗？ | 无 |
| 没有正确地关闭文件吗？ | 无 |

* 限制性检查

对一些程序中使用到的、具有使用限制的命令、事件、方法、过程、函数、对象、控件等进行检查。检查在长时间运行时，基本不会达到软件限制条件。

### 软件一致性检查

* 编译检查

要求提交的源代码在其规定的编译环境中，能够重新编译无错误，并且能够完成相应的功能。

* 安装／卸载检查

在新系统上用交付的软件安装盘重新安装各个模块，并且通过运行这些软件模块，能否完成相应的功能。在安装后立即卸载所安装的模块，能够做到彻底卸载。

* 运行模块检查

将新安装的软件模块与现场运行模块用软件工具抽样比较，确认交付的软件安装盘与现场运行软件一致。

抽查数处现场运行模块用软件工具比较，确认现场运行软件一致。

## 软件系统测试

### 界面(外观)测试

本系统采用图形界面，软件界面较为简单。

界面如下：



### 可用性测试

本次测试备份模块一共发生2个Bug，这2个Bug已被开发人员全部修复，现已处于关闭状态。

Bug汇总：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 致命 | 严重 | 一般 | 轻微 | 建议 | 总数 |
| 总数 | 0 | 1 | 0 | 1 | 0 | 2 |
| 已关闭 | 0 | 1 | 0 | 1 | 0 | 2 |
| 遗留Bug数 | 0 | 0 | 0 | 0 | 0 | 0 |

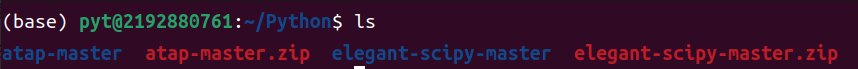
### 功能测试

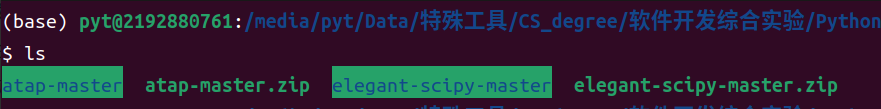
总的功能黑盒测试：

|  |  |  |
| --- | --- | --- |
| 测试模块 | 测试项目 | 测试结果 |
| 文件类型支持 | 普通文件 | 测试通过 |
| 目录文件 | 测试通过 |
| 块设备文件 | 测试通过 |
| 字符设备文件 | 测试通过 |
| 套接字文件 | 测试通过 |
| 管道文件 | 测试通过 |
| 链接文件 | 测试通过 |
| 备份功能 | 打包压缩备份 | 测试通过 |
| 加密功能 | 加密文件 | 测试通过 |
| 恢复功能 | 所选时间节点对应恢复版本 | 测试通过 |

下图为本系统的运行截图。

图 １: 压缩后文件

图 ３: 备份后

图 ２: 备份前

### 强壮性(恢复)测试

容错能力/恢复能力测试用例：

|  |  |  |
| --- | --- | --- |
| 异常输入动作 | 容错能力/恢复能力 | 造成的危害、损失 |
| 输入未知格式文件 | 警告，重启时正常 | 为造成危害 |
| 正在上传时，关闭窗口 | 重启时，系统正常 | 未造成危害 |
| 备份还原时，关闭窗口 | 备份文件未有损害 | 未造成危害 |
| 正在使用系统时，关闭电源 | 重启时系统正常工作 | 未造成危害 |