

**<<多电梯调度软件>>**

**测试计划书**



北京航空航天大学

2016-12

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| V1.0 | 12/20 | 岳明哲 | 岳明哲 | 完成部分工作，其余任务成员没有及时完成，下一版补充 |
| V2.0 | 12/21 | 岳明哲 | 岳明哲 | 完成测试内容 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 项目组成员信息 | | | |
| **小组名称** | **软工实践第27小组** | | |
| **学号** | **姓名** | **主要工作内容** | **贡献比例** |
| 14061173 | 岳明哲 | 编制、审核、提交 | 50% |
| 14061153 | 张东鑫 | 编制 | 25% |
| 14061172 | 李康宇 | 编制 | 25% |

目 录

[1引言 1](#_Toc307901232)

[1.1编写目的 1](#_Toc307901233)

[1.2背景 1](#_Toc307901234)

[1.3定义 1](#_Toc307901235)

[1.4参考资料 1](#_Toc307901236)

[2计划 1](#_Toc307901237)

[2.1软件说明 1](#_Toc307901238)

[2.2测试内容 2](#_Toc307901239)

[2.3测试1（标识符） 2](#_Toc307901240)

[2.3.1进度安排 2](#_Toc307901241)

[2.3.2条件 2](#_Toc307901242)

[2.3.3测试资料 2](#_Toc307901243)

[2.3.4测试培训 2](#_Toc307901244)

[2.4测试2（标识符） 3](#_Toc307901245)

[3测试设计说明 3](#_Toc307901246)

[3.1测试1（标识符） 3](#_Toc307901247)

[3.1.1控制 3](#_Toc307901248)

[3.1.2输入 3](#_Toc307901249)

[3.1.3输出 3](#_Toc307901250)

[3.1.4过程 3](#_Toc307901251)

[3.2测试2（标识符） 3](#_Toc307901252)

[4评价准则 4](#_Toc307901253)

[4.1范围 4](#_Toc307901254)

[4.2数据整理 4](#_Toc307901255)

[4.3尺度 4](#_Toc307901256)

**软件测试计划书**

# 1引言

## 1.1编写目的

本测试计划对多部电梯调度软件的测试工作做出了整体的规划，用于指导后续测试工作的开展。详尽描述多部电梯调度软件的测试方法，测试用例的设计，记录测试的执行结果以及对测试结果的评估。为发现软件存在的漏洞提供方法，为改进软件提供依据。

本软件计划书适用于所有的软件设计人员和测试人员。

## 1.2背景

说明：

1. 文档标示

文档标识号：A2016-00-03-00

文档标题：多电梯调度软件软件测试计划数

项目/产品中文全称：多部电梯集中调度系统

项目/产品英文全称： Multiple-elevator Dispatching System

项目/产品英文缩写：MEDS\_1.0

版本号：V1.0

1. 该开发项目经过了制定软件开发计划，确定系统需求，完成系统设计的过程。在进行此次测试计划之前要完成上述的工作，其中系统设计工作可以和测试工作同步进行，每实现一个模块的设计可以进行相应的模块测试。

## 1.3定义

MEDS：多部电梯集中调度系统（Multiple-elevator Dispatching System）

## 1.4参考资料

A2010-00-01-00.SRS-系统需求规格说明书

第27组多电梯调度软件项目计划书v3.0

第27组多电梯调度软件需求系统需求规格说明书v1.0

A2016-01-02-01. 多电梯调度软件系统设计文档

# 2计划

## 2.1软件说明

1）用户功能

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 输入 | 输出 | 说明 |
| 电梯外部按键（上） | 触发按键 | 触发成功/触发失败 | 若触发成功则会获得触发按钮所在的楼层和前往的方向（上）。否则触发失败 |
| 电梯外部按键（下） | 触发按键 | 触发成功/触发失败 | 若触发成功则会获得触发按钮所在的楼层和前往的方向（下）。否则触发失败 |
| 电梯内部选择楼层 | 触发按键 | 触发成功/触发失败 | 若触发成功则会获得触发按钮所要前往的楼层。否则触发失败 |
| 控制开门 | 触发按键 | 触发成功/触发失败 | 若触发成功则会收到控制电梯门打开的信号。否则触发失败 |
| 控制关门 | 触发按键 | 触发成功/触发失败 | 若触发成功则会收到控制电梯门关闭的信号。否则触发失败 |
| 确定操作 | 触发按键 | 触发成功/触发失败 | 若触发成功则会获得确认的信息。否则触发失败 |
| 报警 | 触发按键 | 触发成功/触发失败 | 若触发成功则会获得报警信息。否则触发失败 |

2）控制功能

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 输入 | 输出 | 说明 |
| 暂停电梯运行 | 点击按钮 | 操作成功/失败 | 无 |
| 启动电梯 | 点击按钮 | 操作成功/失败 | 无 |

3）调度功能

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 输入 | 输出 | 说明 |
| 控制电梯方向向上 | 单个请求 | 电梯运行方向 | 根据当前电梯的位置和要去的楼层确定方向 |
| 控制电梯方向向下 | 单个请求 | 电梯运行方向 | 根据当前电梯的位置和要去的楼层确定方向 |
| 选择最佳电梯调度方案 | 请求队列 | 调用电梯的顺序 | 根据多个请求和不同电梯所在的位置确定电梯的调用顺序 |

## 2.2测试内容

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称标识符 | 起始时间 | 终止时间 | 时长 | 测试内容 | 测试目的 |
| 模块功能测试 | 2016.12.01 | 2016.12.2 | 2天 | 分模块测试软件单元的组合能否正常工作 | 测试单个模块的正确性 |
| 软件集成测试 | 2016.12.05 | 2016.12.05 | 1天 | 测试构成系统的所有模块组合 | 测试软件整体功能的正确性 |
| 性能测试 | 2016.12.11 | 2016.12.11 | 1天 | 测试软件的运行情况 | 保障软件在不同的条件下能够正常运行 |
| 运行环境测试 | 2016.12.11 | 2016.12.11 | 1天 | 在不同的环境下运行本软件 | 确保软件的兼容性能在不同平台上正确执行 |

表2-2-1测试内容表

## 2.3测试1（模块功能测试）

参与这项测试的人员为岳明哲。被测试的模块如下：

|  |  |
| --- | --- |
| 模块标号 | 名称 |
| 1 | 多线程电梯模块ElevatorThread |
| 2 | GUI界面中的按钮类IdentifiedButton |
| 3 | 调度监视模块 |
| 4 | 总体GUI界面模块 |

表2-3-1测试模块表

### 2.3.1进度安排

计划测试时间为2016年12月1日-2016年12月2日，时长两天，工作内容包括熟悉代码运行环境。准备输入数据，确定测试方法，汇总BUG信息。

### 2.3.2条件

陈述本项测试工作对资源的要求，包括：

1. 要求客户计算机内存4G以上，硬盘空间500M以上
2. 需要java运行环境，jre版本在1.8以上
3. 要求安装java开发工具如Eclipse
4. 测试使用插件：Junit单元测试
5. 测试设备：windows操作系统电脑一台
6. 测试人员：1人

### 2.3.3测试资料

列出本项测试所需的资料，如：

1. A2010-00-01-00.SRS-系统需求规格说明书

第27组多电梯调度软件项目计划书v3.0

第27组多电梯调度软件需求系统需求规格说明书v1.0

-01-02-01. 多电梯调度软件系统设计文档

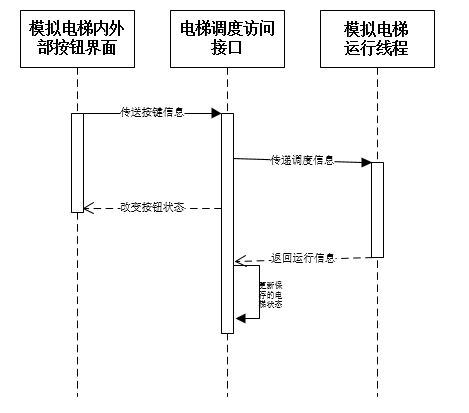
1. 测试的输入和输出举例：对于测试电梯调度，通过Junit将请求信息作为输入加入到调度模块中，预期结果：得出调度最佳电梯

### 2.3.4测试培训

项目组成员已具备所需技术

## 2.4测试2（软件集成测试）

参与这项测试的人员为李康宇。被测试部位包括对这个软件的黑盒测试。



**图一 控制电梯状态序列图**

### 2.3.1进度安排

计划测试时间为2016年12月5日-2016年12月5日，时长一天，工作内容包括熟悉代码运行环境。准备输入数据，确定测试方法，汇总BUG信息。

### 2.3.2条件

陈述本项测试工作对资源的要求，包括：

1. 要求客户计算机内存4G以上，硬盘空间500M以上
2. 需要java运行环境，jre版本在1.8以上
3. 要求安装java开发工具如Eclipse
4. 测试使用插件：Junit单元测试
5. 测试设备：windows操作系统电脑一台
6. 测试人员：1人

### 2.3.3测试资料

列出本项测试所需的资料，如：

1. A2010-00-01-00.SRS-系统需求规格说明书

第27组多电梯调度软件项目计划书v3.0

第27组多电梯调度软件需求系统需求规格说明书v1.0

-01-02-01. 多电梯调度软件系统设计文档

1. 对于集成测试流程根据使用流程进行测试。

1）用户请求电梯，根据模拟按钮发出请求到调度器模块，调度器模块根据当前多部电梯的状态选择最近且顺路的电梯运送乘客。

2）控制停止电梯，通过主界面能够停止电梯的运行

3）在电梯内用户能够通过报警键发出报警

### 2.3.4测试培训

项目组成员已具备所需技术

## 2.5测试3（软件性能测试）

参与这项测试的人员为张东鑫。被测试部位包括对这个软件的黑盒测试。通过正常输入和压力测试，测试系统的正确性和反应时间。

### 2.3.1进度安排

计划测试时间为2016年12月11日-2016年12月11日，时长一天，工作内容包括熟悉代码运行环境。准备输入数据，确定测试方法，汇总BUG信息。

### 2.3.2条件

陈述本项测试工作对资源的要求，包括：

1. 要求客户计算机内存4G以上，硬盘空间500M以上
2. 需要java运行环境，jre版本在1.8以上
3. 要求安装java开发工具如Eclipse
4. 测试使用插件：Junit单元测试
5. 测试设备：windows操作系统电脑一台
6. 测试人员：1人

### 2.3.3测试资料

列出本项测试所需的资料，如：

1. A2010-00-01-00.SRS-系统需求规格说明书

第27组多电梯调度软件项目计划书v3.0

第27组多电梯调度软件需求系统需求规格说明书v1.0

-01-02-01. 多电梯调度软件系统设计文档

1. 对于性能测试流程根据使用流程进行测试（同集成测试），在此基础上进行压力测试，测试系统的反应时间和此时的正确率。

### 2.3.4测试培训

项目组成员已具备所需技术

## 2.4测试4（软件环境测试）

参与这项测试的人员为张东鑫。被测试部位包括对这个软件的黑盒测试。

### 2.3.1进度安排

计划测试时间为2016年12月11日-2016年12月11日，时长一天，工作内容包括熟悉代码运行环境。准备输入数据，确定测试方法，汇总BUG信息。

### 2.3.2条件

陈述本项测试工作对资源的要求，包括：

1. 要求客户计算机内存4G以上，硬盘空间500M以上
2. 需要java运行环境，jre版本在1.8以上
3. 要求安装java开发工具如Eclipse
4. 测试使用插件：Junit单元测试
5. 测试设备：windows操作系统电脑一台
6. 测试人员：1人

### 2.3.3测试资料

列出本项测试所需的资料，如：

1. 1A2010-00-01-00.SRS-系统需求规格说明书

第27组多电梯调度软件项目计划书v3.0

第27组多电梯调度软件需求系统需求规格说明书v1.0

-01-02-01. 多电梯调度软件系统设计文档

### 2.3.4测试培训

项目组成员已具备所需技术

# 3测试设计说明

## 3.1测试1（模块功能测试）

第一项测试通过白盒测试，对模块进行覆盖性测试，保障模块基本功能的正确性和完备性。提高程序的鲁棒性。

### 3.1.1控制

本次测试输入通过半自动引入和手动输入结合的方式进行。其中将部分测试样例写入Junit的测试代码中进行覆盖性测试。

### 3.1.2输入

说明本项测试中所使用的输入数据及选择这些输入数据的策略。

1）多线程电梯模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 被测模块 | 输入 | 采取策略 |
| 1.01 | 多线程电梯 | 初始请求 | 传入请求电梯状态随之改变 |
| 1.02 | 多线程电梯 | 一条指令，其中指令的目标楼层比当前楼层低 | 对电梯根据目标楼层和当前楼层对状态的改变进行覆盖性测试 |
| 1.03 | 多线程电梯 | 一条指令，其中指令的目标楼层比当前楼层高 | 对电梯根据目标楼层和当前楼层对状态的改变进行覆盖性测试 |
| 1.04 | 多线程电梯 | 调用线程控制的方法 | 关闭电梯 |
| 1.05 | 多线程电梯 | 调用Opendoor方法 | 控制台是否正确输出提醒 |
| 1.06 | 多线程电梯 | 调用closedoor方法 | 控制台是否正确输出提醒 |

表3-1-2-1多线程电梯模块输入表

2）电梯GUI场景模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 被测模块 | 输入 | 采取策略 |
| 2.01 | 电梯GUI场景模块 | 无 | 直接运行代码通过观察界面是否完整检测基本绘制功能 |
| 2.02 | 电梯外部GUI场景模块 | 点击按钮 | 点击全部按钮覆盖性检测是否能够得到正确的所在楼层和前往的方向 |
| 2.03 | 电梯内部GUI场景模块 | 点击按钮 | 点击全部按钮覆盖性检测是否能够得到正确的所在楼层和前往的方向 |

表3-1-2-2电梯GUI场景模块输入表

3）调度器模块

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 被测模块 | 输入 | 采取策略 |
| 3.01 | 调度器模块 | 测试电梯当前静止或者是与请求同方向运行的请求 | 覆盖其中一种情况 |
| 3.01 | 调度器模块 | 一个请求 | 此请求同时有两部电梯可以满足，但是从中根据距离选取最近的电梯运送。 |
| 3.02 | 调度器模块 | 一条或多条请求  要求：当前其中一条或多条请求无法被执行 | 覆盖需要把请求放入队列等待能够执行的电梯执行的情况 |
| 3.03 | 调度器模块 | 一条请求  要求：距离两部静止电梯距离相等的请求 | 覆盖随机选择部分 |
| 3.04 | 调度器模块 | 一条请求  要求：对于两部电梯满足距离两部电梯相等，且其中一部静止 | 选择运行中的电梯 |

表3-1-2-3调度器模块输入表

### 3.1.3输出

说明预期的输出数据，如测试结果及可能产生的中间结果或运行信息。

|  |  |
| --- | --- |
| 测试用例(编号) | 输出 |
| 1.01 | 电梯状态改变 |
| 1.02 | 电梯状态变为下 |
| 1.03 | 电梯状态变为上 |
| 1.04 | 电梯状态变为stop |
| 1.05 | 控制台输出opendoor且图形化界面会发生相应改变 |
| 1.06 | 控制台输出closedoor且图形化界面会发生相应改变 |
| 2.01 | 正确绘制界面 |
| 2.02 | 能够得到按钮代表的正确含义 |
| 2.03 | 能够得到按钮代表的正确含义 |
| 3.01 | 请求加入到该电梯的运行方向队列中，在图形化界面完成单部电梯的运行 |
| 3.01 | 选取正确的电梯 |
| 3.02 | 选取正确的电梯 |
| 3.03 | 选取正确的电梯 |
| 3.04 | 选取正确的电梯 |

表3-1-3-1输出表

## 3.1.4过程

设计正确的测试样例，在运行中通过工具观察代码覆盖率，观察多线程相关的部分的正确性。

## 3.2测试2（集成测试）

第二项测试通过黑盒测试，对程序的各子功能进行集成测试，保障程序基本功能的正确性和完备性。提高程序的鲁棒性。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 被测项 | 被测子项 | 测试用例编号 | 需求回溯 |
| 2.01 | 启动关闭系统 | 启动系统 | 2.01.1 | 系统正常运行需求 |
| 关闭系统 | 2.01.2 |
| 2.02 | 启动关闭电梯 | 启动电梯线程 | 2.02.1 | 电梯控制需求 |
| 关闭电梯线程 | 2.02.2 |
| 2.03 | 控制电梯 | 楼层上升指令以及捎带情况 | 2.03.1.1 | 电梯控制需求 |
| 2.03.1.2 |
| 2.03.1.3 |
| 2.03.1.4 |
| 2.03.1.5 |
| 楼层下降指令以及捎带情况 | 2.03.2.1 |
| 2.03.2.2 |
| 2.03.2.3 |
| 2.03.2.4 |
| 2.03.2.5 |
| 电梯上升指令以及捎带情况 | 2.03.3.1 |
| 2.03.3.2 |
| 2.03.3.3 |
| 2.03.3.4 |
| 2.03.3.5 |
| 电梯下降指令以及捎带情况 | 2.03.4.1 |
| 2.03.4.2 |
| 2.03.4.3 |
| 2.03.4.4 |
| 2.03.4.5 |
| 2.04 | 电梯状态查询 | 主面板电梯状态 | 2.04.1 | 电梯状态查询需求 |
| 电梯面板电梯状态 | 2.04.2 |
| 2.04.3 |

表3-2-1-1测试内容表

### 3.2.1控制

本次测试输入通过手动输入的方式进行。在电梯UI界面进行操作，人工核对结果正误。

### 3.2.2输入

|  |  |
| --- | --- |
| 测试用例编号 | 输入 |
| 2.01.1 | 启动系统 |
| 2.01.2 | 关闭系统 |
| 2.02.1 | 随机启动一个电梯线程 |
| 2.02.2 | 随机关闭一个电梯线程 |
| 2.03.1.1 | 启动1个电梯线程，随机发出一条楼层上升指令 |
| 2.03.1.2 | 启动2个电梯线程，随机发出一条楼层上升指令 |
| 2.03.1.3 | 启动3个电梯线程，随机发出一条楼层上升指令 |
| 2.03.1.4 | 启动4个电梯线程，随机发出一条楼层上升指令 |
| 2.03.1.5 | 启动5个电梯线程，随机发出一条楼层上升指令 |
| 2.03.2.1 | 启动1个电梯线程，随机发出一条楼层下降指令 |
| 2.03.2.2 | 启动2个电梯线程，随机发出一条楼层下降指令 |
| 2.03.2.3 | 启动3个电梯线程，随机发出一条楼层下降指令 |
| 2.03.2.4 | 启动4个电梯线程，随机发出一条楼层下降指令 |
| 2.03.2.5 | 启动5个电梯线程，随机发出一条楼层下降指令 |
| 2.03.3.1 | 启动1个电梯线程，随机发出一条电梯上升指令 |
| 2.03.3.2 | 启动2个电梯线程，随机发出一条电梯上升指令 |
| 2.03.3.3 | 启动3个电梯线程，随机发出一条电梯上升指令 |
| 2.03.3.4 | 启动4个电梯线程，随机发出一条电梯上升指令 |
| 2.03.3.5 | 启动5个电梯线程，随机发出一条电梯上升指令 |
| 2.03.4.1 | 启动1个电梯线程，随机发出一条电梯下降指令 |
| 2.03.4.2 | 启动2个电梯线程，随机发出一条电梯下降指令 |
| 2.03.4.3 | 启动3个电梯线程，随机发出一条电梯下降指令 |
| 2.03.4.4 | 启动4个电梯线程，随机发出一条电梯下降指令 |
| 2.03.4.5 | 启动5个电梯线程，随机发出一条电梯下降指令 |
| 2.04.1 | - |
| 2.04.2 | 随机选取正在运行的一个电梯开启电梯面板 |
| 2.04.3 | 关闭电梯面板 |

表3-2-2-1集成测试输入表

### 3.2.3输出

|  |  |
| --- | --- |
| 测试用例编号 | 输出 |
| 2.01.1 | 系统启动 |
| 2.01.2 | 系统关闭 |
| 2.02.1 | 该电梯线程启动 |
| 2.02.2 | 该电梯线程关闭 |
| 2.03.1.1 | 电梯响应楼层上升指令 |
| 2.03.1.2 | 电梯经过调度，响应楼层上升指令 |
| 2.03.1.3 | 电梯经过调度，响应楼层上升指令 |
| 2.03.1.4 | 电梯经过调度，响应楼层上升指令 |
| 2.03.1.5 | 电梯经过调度，响应楼层上升指令 |
| 2.03.2.1 | 电梯响应楼层下降指令 |
| 2.03.2.2 | 电梯经过调度，响应楼层下降指令 |
| 2.03.2.3 | 电梯经过调度，响应楼层下降指令 |
| 2.03.2.4 | 电梯经过调度，响应楼层下降指令 |
| 2.03.2.5 | 电梯经过调度，响应楼层下降指令 |
| 2.03.3.1 | 电梯响应电梯上升指令 |
| 2.03.3.2 | 电梯经过调度，响应电梯上升指令 |
| 2.03.3.3 | 电梯经过调度，响应电梯上升指令 |
| 2.03.3.4 | 电梯经过调度，响应电梯上升指令 |
| 2.03.3.5 | 电梯经过调度，响应电梯上升指令 |
| 2.03.4.1 | 电梯响应电梯下降指令 |
| 2.03.4.2 | 电梯经过调度，响应电梯下降指令 |
| 2.03.4.3 | 电梯经过调度，响应电梯下降指令 |
| 2.03.4.4 | 电梯经过调度，响应电梯下降指令 |
| 2.03.4.5 | 电梯经过调度，响应电梯下降指令 |
| 2.04.1 | 主面板电梯状态随电梯运行而实时改变 |
| 2.04.2 | 电梯面板展示电梯当前内部信息 |
| 2.04.3 | 电梯面板关闭 |

表3-2-3-1集成测试输出表

### 3.2.4 过程

包括测试样例的准备，测试环境的初始化，对测试输入输出的记录以及分析，以及测试过程中意外情况的记录。

## 3.3 测试3（性能测试）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 被测项 | 被测子项 | 测试用例编号 | 需求回溯 |
| 3.01.01 | 控制电梯 | 操作电梯升降楼层 | 3.01.01.1 | 电梯控制需求 |
| 3.01.01.2 |
| 3.01.01.3 |
| 3.01.01.4 |
| 3.02.01 | 压力测试 | 测试长时间工作软件稳定性 | 3.02.01.1 | 软件稳定性需求 |
| 3.03.01 | 负载测试 | 测试不同数量请求下软件的性能 | 3.03.01.1 | 软件性能需求 |

3-3-0性能测试内容表

### 3.3.1控制

1. 本测试通过在模拟测试窗口输入相应的请求，观察电梯响应时间，程序的运行状态来完成测试。
2. 操作顺序为先测试电梯的基本升降功能，再测试软件在长时间运行状态下的稳定性，最后测试不同负载下软件对请求的响应情况。
3. 结果记录方式为表格文件。

### 3.3.2 输入

|  |  |
| --- | --- |
| 测试用例编号 | 输入 |
| 3.01.01.1 | 楼层发出请求使电梯向上 |
| 3.01.01.2 | 楼层发出请求使电梯向下 |
| 3.01.01.3 | 电梯内发出大于电梯当前楼层的请求 |
| 3.01.01.4 | 电梯内发出小于电梯当前楼层的请求 |
| 3.02.01.1 | 长时间运行软件（期间不断发出请求） |
| 3.03.01.1 | 在较短时间内发出一定量的请求（多次测试，每次测试请求数量递增） |

表3-3-2-1 性能测试输入

### 3.3.3 输出

|  |  |
| --- | --- |
| **测试用例（编号）** | **输出** |
| 3.01.01.1 | 响应时间 |
| 3.01.01.2 | 响应时间 |
| 3.01.01.3 | 响应时间 |
| 3.01.01.4 | 响应时间 |
| 3.02.01.1 | 软件运行状态 |
| 3.03.01.1 | 请求完成情况，响应时间 |

表3-3-3-1 性能测试输出

### 3.3.4 过程

1) 测试的准备：准备测试样例，人员分工

2) 测试初始化：开启模拟测试窗口

3) 测试中间步聚：电梯控制功能测试，压力测试和负载测试。

首先测试电梯基本功能的响应时间，压力测试：测试电梯调度系统在长时间运行状态下的稳定性。负载测试：测试软件对大量请求的处理能力。

1. 测试运行结束方式：所有测试样例测试完成，汇总测试结果后关闭模拟窗口。

## 3.4测试4（运行环境测试）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 被测项 | 被测子项 | 测试用例(编号) | 需求回溯 |
| 4.01 | 运行环境（操作系统）测试 | Windows7 | 5.01.1 | 不同用户操作系统兼容需求 |
| Unix | 5.01.2 |
| OS | 5.01.3 |
| Windows8 | 5.01.4 |
| Windows8.1 | 5.01.5 |
| Windows10 | 5.01.6 |

表3-4-0-1运行环境测试内容表

### 3.4.1控制

1. 本测试通过在相应的操作系统下运行程序，检测程序运行状态，功能是否正常来完成测试，每个环境测试完成后将结果进行记录。

2) 操作顺序为Windows7，Unix，OS，Windows8，Windows8.1，Windows10.

3) 结果记录方式为表格文件。

### 3.4.2 输入

|  |  |
| --- | --- |
| 测试用例(编号) | 输入 |
| 4.01.1 | Windows7操作系统 |
| 4.01.2 | Unix操作系统 |
| 4.01.3 | OS操作系统 |
| 4.01.4 | Windows8操作系统 |
| 4.01.5 | Windows8.1操作系统 |
| 4.01.6 | Windows10操作系统 |

表3-4-2-1 运行环境测试输入

### 3.4.3 输出

|  |  |
| --- | --- |
| 测试用例(编号) | 输出 |
| 4.01.1 | Windows7下能否正常工作 |
| 4.01.2 | Unix下能否正常工作 |
| 4.01.3 | OS下能否正常工作 |
| 4.01.4 | Windows8下能否正常工作 |
| 4.01.5 | Windows8.1下能否正常工作 |
| 4.01.6 | Windows10下能否正常工作 |

表3-4-3-1 运行环境测试输出

### 3.4.4 过程

1) 测试的准备：准备测试样例，人员分工，准备测试环境。

2) 测试中间步聚：分别对不同的操作系统环境进行功能测试，检测程序的运行状态，记录运行状态信息和功能是否正常。

3) 测试结束：某一操作系统下所有测试样例测试完成，该环境测试完成，所有操作系统都完成测试后，测试完成。

# 4评价准则

## 4.1范围

测试用例的范围：本系统测试的主要内容包括模块测试、集成测试、确定测试。覆盖了软件的所有功能需求，保证在各种输入情况下可以给出正确的符合预期的输出。

限制：模拟电梯和实际电梯存在一定差异，在实际情况中可能会出现部分差异。

## 4.2数据整理

根据Junit工具自动检测代码覆盖性，对于结果的正确性通过手工方式进行检查

## 4.3尺度

测试与预期完全一致，每个请求都能得到正确执行。如果出现运送错误将错误率控制在1%以内。