

# 00 第二次作业要求

2018

## 1. 作业目标

通过设计一个简单的**单部电梯**运行控制系统，要求设计者以面向对象程序的设计方式来实现电梯控制系统的具体功能。

## 2. 作业内容和成果物

### 2.1 业务背景

#### 1) 电梯定义

电梯是服务于规定楼层的固定式升降设备。垂直升降电梯具有一个轿厢，运行在至少两列垂直的或倾斜角小于  $15^\circ$  的刚性导轨之间。台阶式自动扶梯和滚梯不在本次作业考虑范围之内。

#### 2) 电梯的简单使用方法（不考虑紧急情况和复杂情况）：

载人电梯都是微机控制的智能化、自动化设备，不需要专门的人员来操作驾驶，普通乘客只要按下列程序乘坐和操作电梯即可。

- i) 在乘梯楼层电梯入口处，根据自己上行或下行的需要，按向上或向下的箭头按钮，只要按钮上的灯亮，就说明你的呼叫已被记录，只要等待电梯到来即可。
- ii) 电梯到达目标楼层停下开门后，先让轿厢内人员走出电梯，然后呼梯者再进入电梯轿厢。进入轿厢后，根据你需要到达的楼层，按下轿厢内操纵盘上相应的数字按钮。同样，只要该按钮灯亮，则说明你的目标楼层已被记录；此时不用进行其他任何操作（在到达目标楼层之前，反复按该目标楼层的按钮为无效操作），一旦电梯门关闭之后，电梯就会开始运动，只要等电梯到达你的目标层停靠开门即可。
- iii) 电梯行驶到你的目标层后会自动开门，此时按顺序走出电梯即结束了一次乘梯过程。

## 2.2 作业电梯系统基本描述

- 1) 本作业的电梯为一个运行在 **10 层** 楼的电梯，电梯可在每层经停。楼层计数采用中国式楼层计数，即 1 层显示为 1，2 层为 2，依次类推，直至顶层显示为 10。
- 2) 每个楼层都有一组电梯运行请求按钮（下面简称为楼层按钮），即**上行请求按钮**和**下行请求按钮**。如果乘客按上行请求按钮时，则表示想搭乘电梯去往上面的楼层；反之，则表示想搭乘电梯去下面的楼层。**规定最底层（1 层）只有上行请求按钮，最顶层（10 层）只有下行请求按钮，中间楼层同时有上行和下行两个方向请求按钮。**
- 3) 一个电梯箱体（剩余文档简称电梯）内有对应每个楼层的目的地请求按钮（下面简称目标楼层），一般显示相应楼层的层数。

## 2.3 电梯基本运行规则

- 1) **程序运行开始或重置时设置电梯停靠在一层；**
- 2) 几个名词解释：
  - a) 电梯运行状态：从电梯启动时刻（此时速度 $>0$ ），到电梯运动停止时（此时速度刚刚为 0）的运行状态。（启动时刻，运行停止时刻]，期间运行速度始终大于 0。
  - b) 电梯开关门状态：电梯静止时，从门打开时刻（门开始动作），到门完全关闭时刻（门刚刚停止动作）时的状态。（门打开时刻，门关闭时刻]
  - c) 电梯停留状态：电梯停在某层，且门长时间处于关闭状态。（门关闭时刻，门准备打开时刻或电梯准备启动时刻]
- 3) 一个楼层按钮**同一时刻**只能发出**一个上行或下行请求**。电梯未到本楼层的时候，某个请求按钮变亮后，再按不会产生实际效果，但是发出上行请求后可以再发出下行请求，反之亦可，这视为两个不同的请求，执行完一个后另一个仍需执行。在电梯到达某楼层处于开关门状态时，该楼层的多个同向请求只认为是一个请求。当电梯关门动作完成后（含门完全静止的那一刻），可以再产生新的上下行请求；
- 4) 电梯内的一个目标楼层按钮只能发出对应目标楼层的请求，一旦发出某个目标楼层请求后，在电梯到达该楼层并完成关门动作前（**包括关门完毕时刻**），

目标楼层与该按钮楼层相同的多个电梯内请求被认为是一个请求。当电梯关门结束后，可以再发出任意目标楼层请求。

- 5) 所有请求按照请求发出的时间顺序被电梯系统管理和调度，按照时间上先来先服务的策略（First Arrived First Served, FAFS）进行调度。
- 6) 如果电梯同时收到了电梯内请求和楼层请求时，则按照输入时的请求排列顺序执行。
- 7) 本次作业的电梯系统采用傻瓜式调度策略：控制系统不断扫描请求队列，按照 FAFS 策略取出待响应请求，只有当该请求被执行完毕之后，才会尝试调度下一个请求。**请注意此处不要按照常见的电梯运行调度策略进行优化设计，不要合并“顺路”的请求。**如从 2 层去 8 层期间，未到 6 层时有 6 层的上行请求，应先处理完 2 层到 8 层的请求，再处理 6 层的请求；或者 2 层到 8 层时，中间又发出了去 6 层的请求，同样先执行完到 8 层的动作，再处理去 6 层的请求。

## 3. 作业内容和成果物

### 3.1 作业内容

实现一个符合 2.2 节和 2.3 节所描述的电梯运行调度 java 程序。

### 3.2 提交内容

- 1) java 语言程序（java 程序文件）；
- 2) 程序说明文档（Readme），内容包括：
  - a) 电梯调度策略和程序功能说明；
  - b) 程序运行所需环境和运行指令规范；
  - c) 程序的输入说明，包括标准输入格式、输入限制和遇见输入错误时的响应信息；
  - d) 程序计算结果的输出规格，以及可预见的运行错误响应信息；
- 3) 程序中若干类的说明文档（Readme，word 文件），详见 4.4 节。

## 4. 作业要求和限制

### 4.1 输入规范

用户输入为**按照请求产生时间排序**的请求序列（注意：可以输入时间相同的两个请求，排在前面请求被优先执行），序列通过字符串表示；

请求分为两类：一类是楼层请求，一类是电梯内请求。

楼层请求格式为：(FR, m, UP/DOWN, T)，其中 FR 为楼层请求标识，m 为发出请求的楼层号，UP 为向上请求，DOWN 为向下请求，T 为发出时刻。（注释：相当于请求者在楼道里的某楼层按“上行”或“下行”键）

电梯内请求格式为：(ER, n, T)，其中 ER 为电梯内请求标识，n 为请求前往的目标楼层号，T 为发出时刻。（注释：相当于人在电梯里按一个目标楼层号）

所有的逗号应采用 ASCII 字符集中的逗号“,”，而不是中文字符逗号“，”。

请求之间必须通过换行进行分隔，两条请求之间不允许有空行。一条请求的内部元素之间可以有空格，要求程序能够自动过滤。

- 运行规则：

T 为请求产生的相对时刻，第一个请求的 T 值必须设置为 0，否则视为一个 crash 错误！设电梯运行一个楼层距离的时间消耗为 0.5s；达到楼层后一次开关门动作时间消耗为 1.0s。

合法的请求产生时刻为非负整数（要求最大为 4 字节的非负整数，格式不得为-0），n, m 为 1~10 之间（包含）的正整数。

不正确的标识符，不正确的方向，不正确的数字范围，多余的其他非允许字符，均认定为不合法输入，即无效输入。

特别地，对于 FR 标识符，1 楼的 DOWN 和 10 楼的 UP 也认为是无效输入。

在一行内只能输入一条请求，一行内输入多条请求将导致 0J 无法识别！输入全部请求后，必须再键入 RUN 表示输入结束，回车后程序开始执行调度。

本程序中的 T 和 t 均为模拟值，即不要按照 T 和 t 的值作为真实时间来控制运行输出。

参考输入样例（例 1）：

(FR,3,DOWN,0)

(FR,1,UP,1)

(ER,1,2)

(ER,6,4)

RUN

本次作业要求一次性输入所有请求，然后执行程序进行电梯调度并输出结果。  
**标准输入的请求是按照时间排序的，如果遇到一个乱序的请求，即请求产生时间小于前面一个请求产生时间，则该请求直接被丢掉，继续处理下一个请求。**

**要求程序能够忽略相同的请求**，包括产生时刻相同的相同请求和产生时刻不同但是实质上相同的请求。详情见 2.3.3 及 2.3.4 中的规定。

例 2:

(FR,3,DOWN,0)

(FR,3,DOWN,1)

RUN

根据前面的运行规则，例 2 中的第二条请求发出时第一条请求还没有执行结束，相当于楼梯按钮仍处于“按下状态”（即电梯尚未到达 3 层，而请求者多次按了“下行按钮”），所以第二条请求与第一条请求实质上相同。

例 3:

(FR,3,DOWN,0)

(FR,3,DOWN,1000)

RUN

第二条请求执行时，按照运行规则此时电梯已到 3 层，该请求相当于是同层请求（即电梯停在本层），应执行一次开关门动作。对于电梯内请求的类似情况同理。

本次作业不要求楼层请求和电梯请求的顺序符合真实情况（所谓真实情况，即实际情况中电梯中如果没有乘客，不会产生电梯内请求，但本次作业做简化考虑，任何顺序的楼层和电梯内请求都可被接受）

**本次作业不允许使用文件作为输入。**

除**黑体**和**红色字**表明的强制规定外，对于更多的细节的输入规范，如与文档冲突，请在 readme 说明文档（PDF 文件）中说明，若没有说明且与文档的冲突，测试者有理由质疑。

## 4.2 输入方式

本程序运行**要求为控制台或命令行输入**，具体输入方式由程序设计者决定，但是要求在**说明文档**中加以明确说明。

### 4.3 输出规范

程序的输出为**按照时间排序**的电梯运行状态描述，包括以下内容：电梯停靠的楼层、停靠前的运动方向及停靠时刻（即电梯刚到达目标楼层由运动转为静止状态，尚未执行开关门的时刻）：

格式为：(n, UP/DOWN, t)

**注意：一行只有一个输出结果，若一行内有多个输出结果，则 0J 无法识别！**

其中 n 为楼层号，UP/DOWN 为电梯运行方向（只显示 UP 或 DOWN 之一）；t 为相对于第一个请求发生的时间。对于 t 的规格有如下要求：**t 是保留小数点后一位有效位的浮点数**，如果 t=3，则要求输出为 3.0；如果 t=0，则要求输出 0.0。不允许出现-0.0 的格式。**否则会导致 0J 误判！**。

有同层请求时（即电梯停在某层，此时有目标为该层的请求），则输出为：(n, STILL, t)，此处 t 应考虑电梯执行一次开关门动作的时间。

**输出必须使用 system.out 标准输出，否则会导致 0J 误判！**

例：

输入为(ER,1,0) 时，应输出(1,STILL,1.0)。解释：电梯停在 1 层，此时在电梯内发出去 1 层的请求，则输出开关门一次后的时刻。

输入为(ER,3,0) (ER,3,5) 时，应输出：

(3,UP,1.0)

(3,STILL,6.0)

**输出格式要求采用 UTF-8 标准。**

**对于本程序不能识别的输入内容，必须按照以下规格进行输出，否则按照错误结果扣分！**

ERROR

*#所对应的错误输入内容*

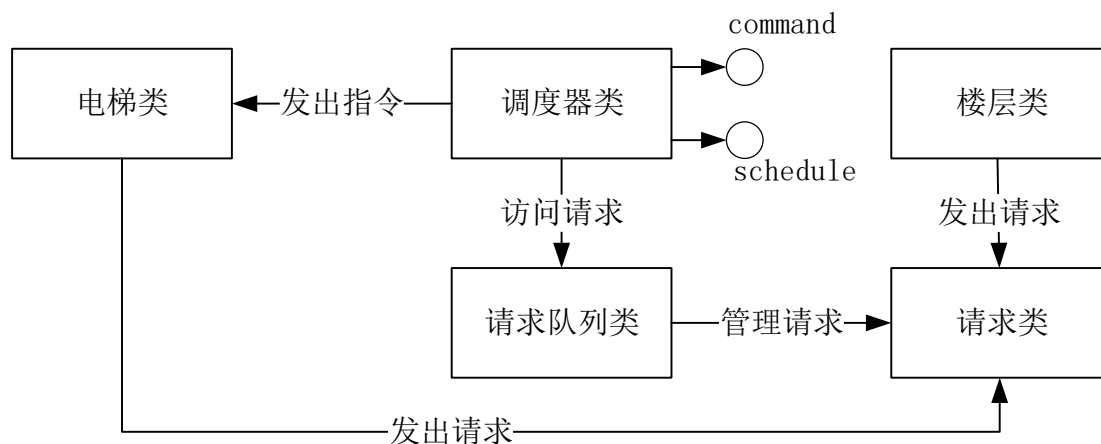
相关的输出规范要在**说明文档**（README）中加以说明。

其他未规定的地方可由编程者自行决定。但是需要在 readme 中说明

### 4.4 设计要求

**本次作业不能使用 lamda 表达式编程！**

必须要实现电梯、楼层、请求队列、调度器、请求这五个类，且类中不允许出现 **Public 属性**。其中楼层类可能相对简单一些，不做具体要求，但是其他四个类的具体实现必须单独编写说明文件，内容包括类属性和方法的简要说明。



几个类之间的协作关系参考图

## 5. 其它说明事项

### 5.1 设计建议

- (1) 从输入读取请求
- (2) 构造请求对象
- (3) 加入到队列中
- (4) 启动调度
- (5) 记录电梯对象对请求的响应

### 5.2 错误处理

处理原则：

- 1) 如果发现输入请求序列不满足时间排序要求，则输出**无效输入提示**（输出格式为**#开头的提示内容**），并忽略不满足要求的请求，继续处理下一个输入请求。
- 2) 遇到无效请求（包括格式或内容不符合要求的），**需要给出无效输入提示**后，继续处理下一个输入请求直至结束。
- 3) **格式有误或数据无效**(如楼层超过 10)的请求将被直接从输入请求序列中去掉，并给出无效输入提示，但不影响继续对其他有效请求的调度处理。

4) 任何情况下，程序都**不应 crash**，要正常结束（`exitcode=0`）。

### 5.3 Tips

如何设计 3 个对象：电梯箱体、楼层、调度器，它们的定义是什么，有什么（属性）；它们能做什么，在什么情况下做（方法）。同理，分析另外 2 个内部管理类。

## 6. 其他规定

- 1) 文档中**粗体**字体部分为强制要求。
- 2) 无效作业，以下四种情况视为无效作业。
  - （1）程序不能编译和运行；
  - （2）未使用 **Java** 语言；
  - （3）所编制的程序不是本次作业的内容。
  - （4）无法通过任何一个可以输出正常结果的公共测试案例；