

操作系统课设一

电梯调度

洪嘉勇

1452822

目录

项目要求概述

3

项目目的 3

开发环境 3

基本需求 3

调度算法概述

4

乘客行为概述 4

电梯行为概述 4

调度 5

类概述

8

ELEVATOR 9

线程概述

12

资源 12

任务 12

项目要求概述

1. 项目目的

- 通过控制电梯调度，实现操作系统调度过程
- 学习特定环境下多线程编程的方法
- 学习调度算法

2. 开发环境

语言：java

系统平台：全平台（具备 java 环境）

IDE：IntelliJ IDEA

产品呈现模式：jar 包

执行环境要求：安装 java

Win：安装 java 配置环境变量后双击

Linux / Mac：命令行：

java -jar 电梯.jar

3. 基本需求

- 模拟 20 层楼中 5 架电梯的调度
- 电梯具有最基本的按键
- 可显示电梯的当前状态

调度算法概述

1. 乘客行为概述

- 乘客可以在 **20** 层楼的任何一层楼按当前楼层的上或者下的按键对电梯提出需求
- 乘客可以按动电梯中的楼层选择按钮来对指定电梯前去哪里，由于 **ui** 的设计问题，这一功能被要求在按动请求按钮时一并完成
- 乘客可以在电梯中按动紧急按钮迫使当前电梯停止运作

2. 电梯行为概述

- 电梯初始状态均为静止，且停泊在第一层
- 电梯通过反复自检自身的状态变量来变更自己的行为
- 行进中的电梯每到一个楼层都自检下客队列，判断当前楼层是否需要开门下客
- 行进中的电梯每到一个楼层都要检查当前楼层乘客等待队列是否有符合当前方向的乘客，判断当前楼层是否要载客，如果在该楼层电梯中没有了乘客且没有应答其他请求，则载上当前楼层人数较多方向的乘客继续行进

3. 调度

● 乘客按下请求按钮响应流程

1. 上下方向上有朝这一楼行进且该电梯的最高 / 低请求大于该楼层：

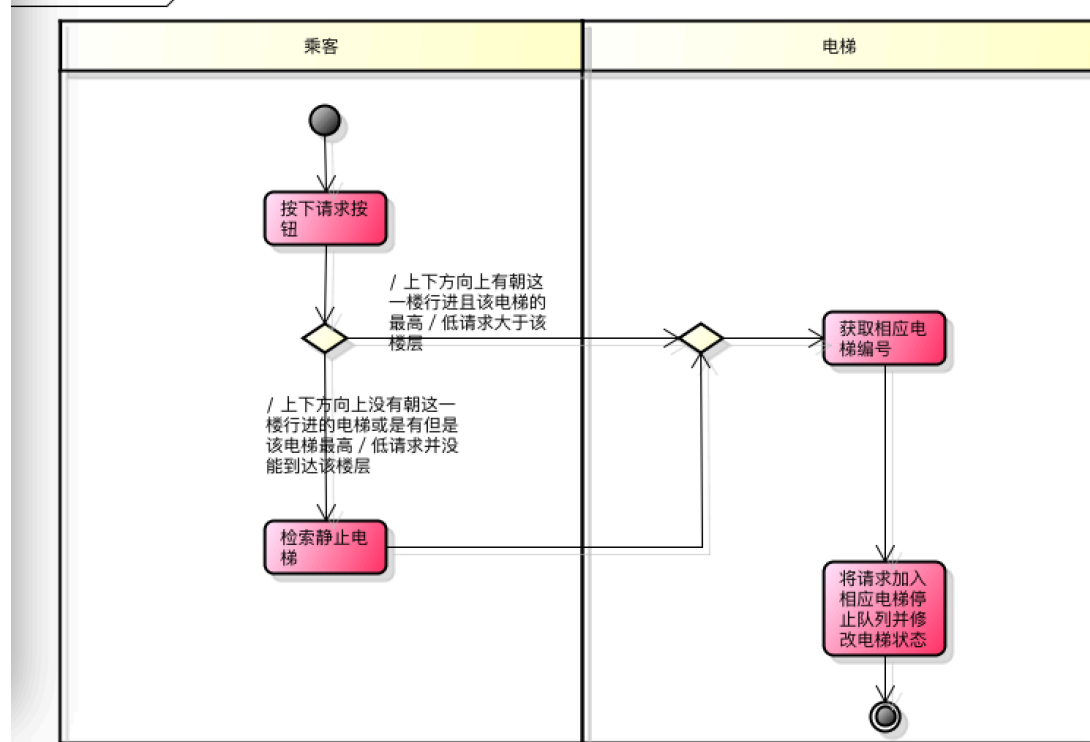
将会等待该电梯到达该楼层来载上该乘客

2. 上下方向上没有朝这一楼行进的电梯或是有但是该电梯最高 / 低请求并没能到达该楼层：

将会进行检索静止的电梯队列：

静止电梯的选择将**位置优先**，选择离该楼层最近的静止电梯来响应请求，将该电梯启动，并将在该楼层停下的指令塞入该电梯。

乘客请求响应活动图



- 行进电梯到达某一楼层执行操作流程

1. 电梯检索自身的停止队列中是否有该楼层

有该楼层：

停留并将队列中的乘客全部弹出队列，将停止队列的该楼层弹出

没有该楼层：

进行下一步

2. 电梯检索当前楼层的请求队列

一． 电梯当前停止队列已经为空：

1. 当前楼层没有请求

电梯设置自身状态变量为静止

2. 当前楼层有请求

电梯选择人多的一个方向载客，将他们弹出请求队列，并设置状态变量然后启动向该方向行进

二． 电梯当前停止队列并未空

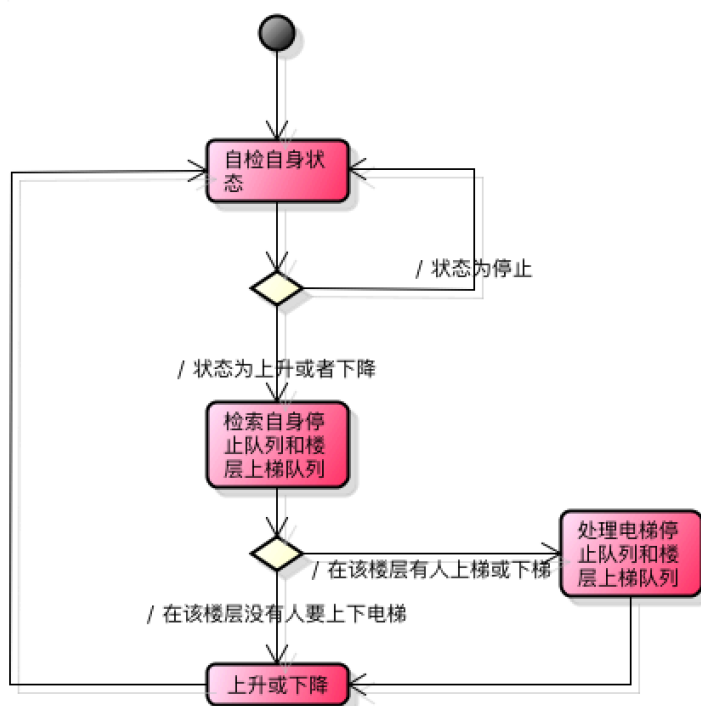
1. 当前楼层没有请求

电梯继续行进

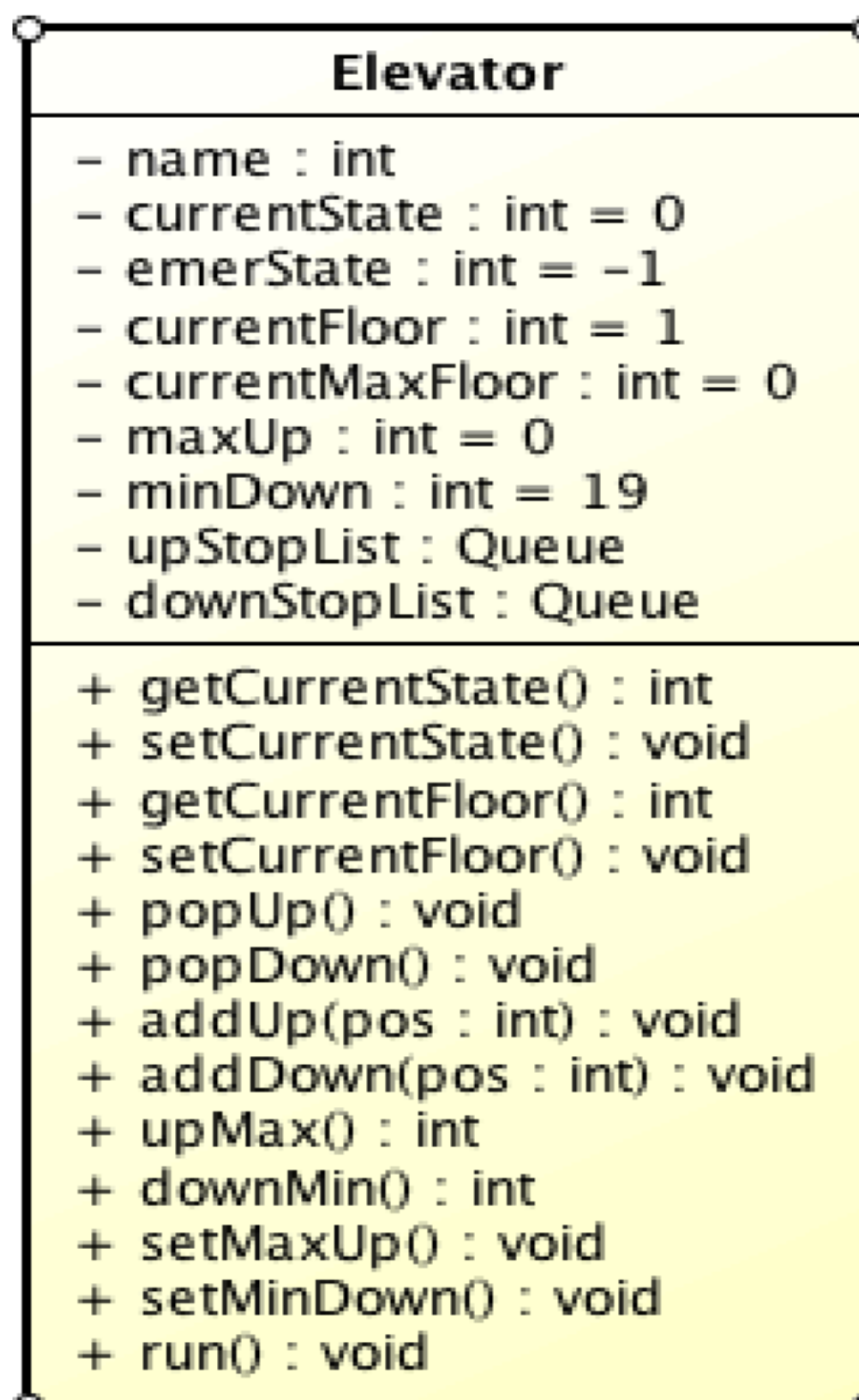
2. 当前楼层有请求

电梯载上对应方向的乘客，将这些乘客从请求队列弹出，继续行进

act 电梯处理请求活动图



类概述



Elevator

私有变量:

- **int name** 电梯名
- **int currentState** 当前状态变量
- **int emerState** 紧急状态变量
- **int currentMaxFloor** 当前可去的最高的楼层
- **int maxUp** 当前电梯要去的最高楼层
- **int minDown** 当前电梯要去的最低楼层
- **Queue<Integer> upStopList** 电梯下降停止队列
- **Queue<Integer> downStopList** 电梯上升停止队列
- **JButton buttonList** ui 中的按钮控件队列

方法:

- **int getCurrentState()** 获取 **currentState**
- **void setCurrentState()** 设置 **currentState**
- **int getCurrentFloor()** 获取 **currentFloor**
- **void setCurrentFloor()** 设置 **currentFloor**
- **void popUp()** 将 **upStopList** 的第一个元素弹出
- **void popDown()** 将 **downStopList** 的第一个元素弹出
- **void addUp(int pos)** 将位置楼层加入 **upStopList**

-
- **void addDown(int pos)** 将位置楼层加入
downStopList
 - **int upMax()** 获取 maxUp
 - **void setUpMax()** 设置 maxUp
 - **int downMin()** 获取 minDown
 - **void setDownMin()** 设置 minDown
 - **void run()** 启动电梯线程

线程概述

资源:

- 电梯

任务:

- 乘客移动