OO第四次作业——ALS电梯调度说明文档

注： 本作业为A级要求

# 输入规范

输入应首先遵循基本法~

用户输入应为一系列请求序列。

请尽量单条敲入/粘贴指令，程序能够接受多条指令同时输入，但是结果可能会不正确。如果执意多条指令复制，每个请求应独占一行，且并发输入个数不应超过100个，否则程序可能将会不能正确运行。由于IDE卡死所导致的程序未响应与本程序无关。（实际上程序能够承受上千条指令的同时输入，但是为了求稳只好加以限制，求谅解）

楼层请求的输入格式为：(FR, n, UP/DOWN)

对于楼层请求来说，

如果括号中出现了除{'F','R','U','P','D','O','W','N','0'~'9', ',' }以外的任何非空字符，该指令视为无效指令。

当n=1且方向为DOWN或者n=20且方向为UP时，该指令视为无效指令。

当n不是1~20之间的整数时，该指令视为无效指令

电梯请求的输入格式为：(ER, #Ele,n)

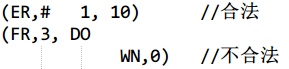
对于电梯请内求来说：

如果括号中出现了除{'E','R','0'~'9',‘#’ ','}以外的任何非空字符，该指令视为无效指令。

当n不是1~10之间的整数时，该指令视为无效指令

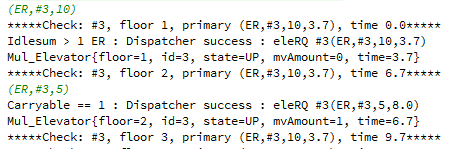
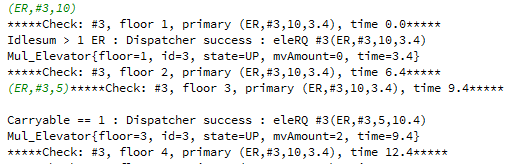
当Ele不是1~3中的整数时，该指令视为无效指令

括号中可以过滤空格，但是每条请求都请在同一行中完成输入，如下图：



本程序支持即时输入，需要即时输入的指令请单起一行输入，如下图：

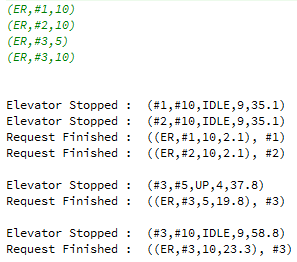
提示信息可能与您看到的不同，下面会介绍如何开启调试提示

# 输入方式

本程序Mul\_Main.java为程序入口

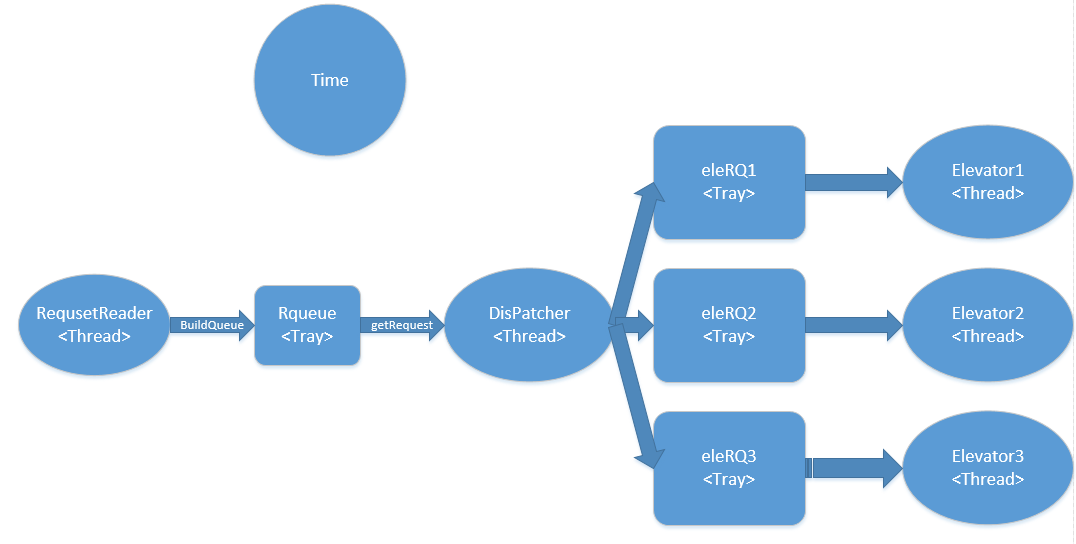
本程序采取控制台(Console)输入，控制台输出。如下图：



# 实现方法

鉴于每个人实现的方法都不尽相同，我觉得还是有必要在这里将我程序里面的细节实现简要地概括一下，以方便您对于我的程序的理解。

首先本程序的大体结构如下图：其中RequestReader、Dispatcher、elevator1~3是五个相互独立的线程，RequestReader和Dispatcher共享一个请求队列RQueue，Dispatcher和三个电梯各自共享一个电梯的请求队列。RequestReader读入指令后写入Queue，Dispatcher扫描RQueue中的请求，如果有电梯能够相应则放入对应的电梯队列，若当前不能分配则会留在RQueue中等待电梯状态的刷新。Dispatcher用来维护三个电梯的请求队列，主要是结合电梯的运动状态来给电梯分配能够相应/捎带的请求。电梯只负责完成eleRQ队列中的请求，当到达某一楼层后会将队列中到达目的地的请求删去。



# 输出格式

1. **基本格式**

电梯运动状态在每条请求完成时输出（即停靠），并连带输出完成的请求

停靠格式为Elevator Stopped :（#电梯, #楼层, 运动方向, 累计运动量，时间)

请求格式为 Request Finished : (被完成的请求, 完成该请求的#电梯)

此处的时间为电梯完成指令后（到达指定楼层），包括开关门的时间。

方向表明电梯现在相对于主请求的运动方向。

注：停靠的时间和请求产生的时间可能有细微的误差，调试的时候请忽略这一点= =

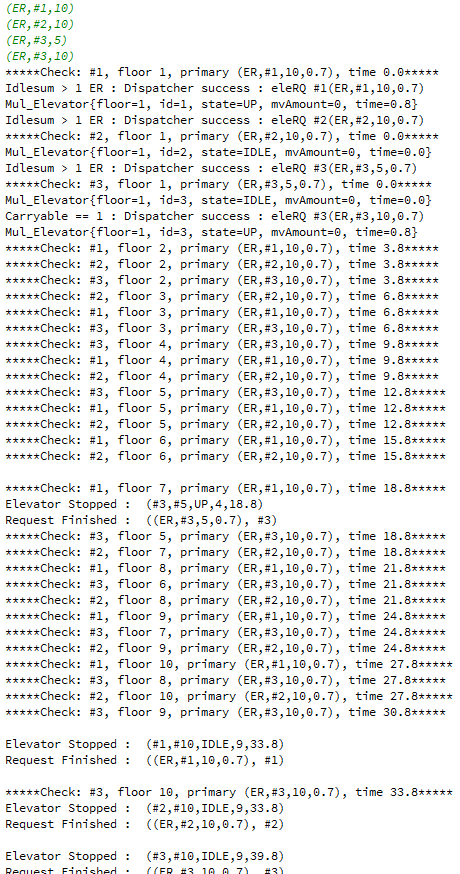
1. 打开调试输出

本着利人利己的原则，我觉得还是有必要将调试功能的开启方法写明。

本程序的调试功能默认为关闭状态，可通过如下方式打开。

修改Mul\_Elevator Line65的注释可以在每次电梯运动时输出电梯的状态

将Mul\_Dispatcher Line94 95 101 102 115 116 123 124 140 142的注释删除可以观察到程序是在何种情况下添加到哪个电梯的请求队列的。开启后效果如下：



当然也有可能因为调试信息太多找不到输出的情况，开不开还是见仁见智吧

为了和谐六系，还请少侠手下留情



# 其他说明

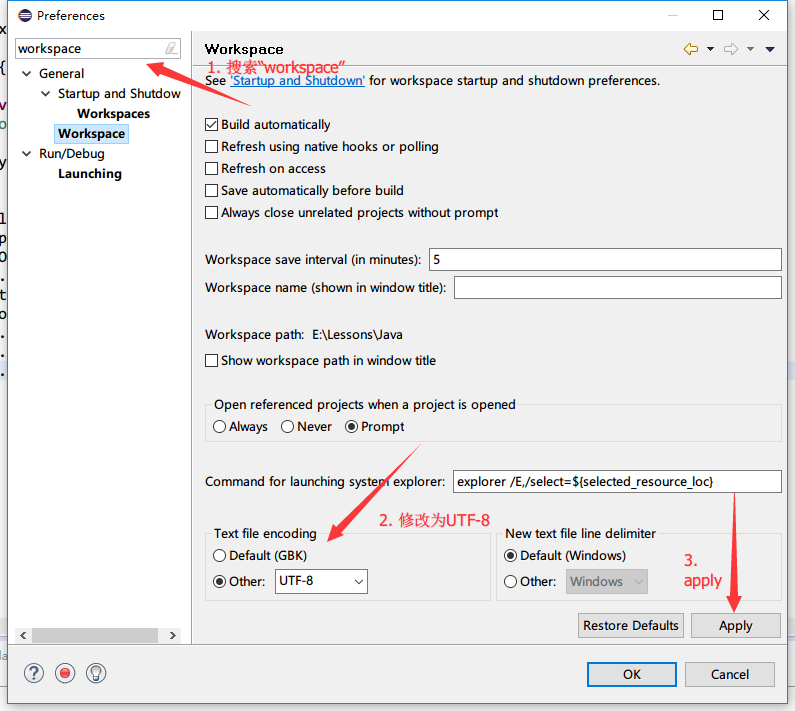
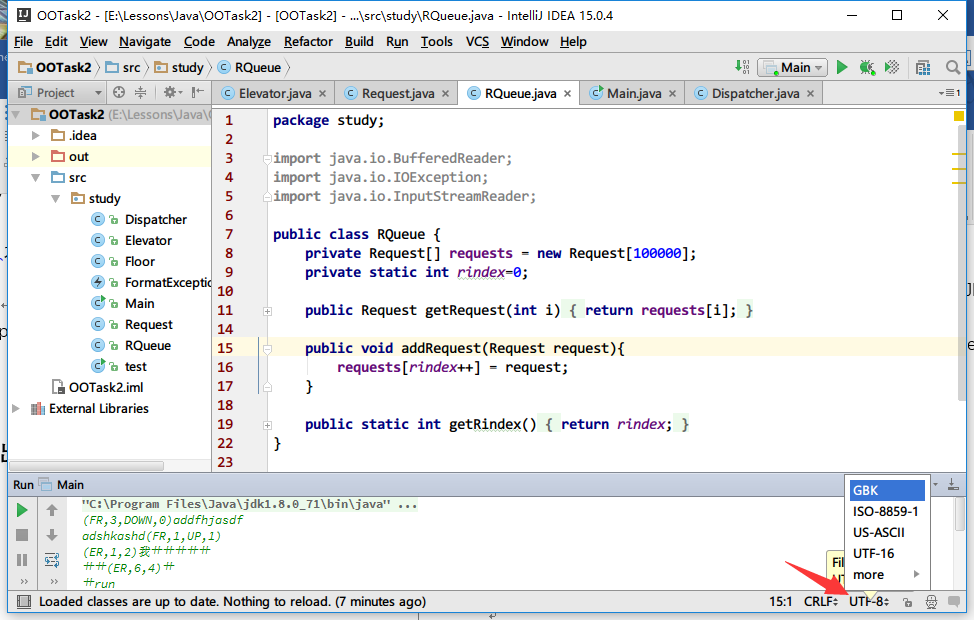
本程序的编码格式为UTF-8格式，所用IDE为Intellij IDEA，JDK版本为1.8。

IDEA导入程序只需将src整个文件夹拖入工程即可，

Eclipse导入工程需要新建一个名为study的包，并import所有.java文件即可。

如果你的JDK版本低于1.8，建议更新JAVA版本。

遇到中文乱码的情况，请调整编码设置~如下图，上为eclipse，下为IDEA

**最后、感谢你对我的程序所做出的付出，恭祝学业有成！**