**功能说明：**

程序模拟出租车的乘客呼叫与应答系统，采用多线程的设计，并结合了相应的GUI即时显示结果。

**程序运行所需环境和运行指令规范：**

本程序需要在java虚拟机环境下运行，乘客请求从控制台输入，输出结果保存在工程目录下的log.txt文件中。

**运行指令为:**

javac TaxiSystem.class

java TaxiSystem

**注意事项**

1.由于官方GUI的性能问题，当有过多请求同时存在时，会导致GUI很卡，甚至出租车的路径不连续（GUI显示的问题），所以实际运行情况以log.txt中的数据为准。

2.本程序中所有的时间是相对程序开始的时间，所以是从0开始的，但是因为每个线程启动的时间并不相同，所以如果电脑性能较差，或者一次性请求过多，会导致每个线程的初始时间误差大于100ms，最终导致他们的初试时间不同，但从自己的实验来看，还没出现过这种情况。

3.对于在7500ms的窗口内抢单的出租车，一但出租车抢单，信用度立即+1，而不是等到7500ms之后才+1。

4.本程序是每个出租车线程sleep之后才更新其位置信息的，所以在抢单时获取的其位置信息很可能是其所在道路后向的节点的位置。

5.本程序最开始出租车的初始状态为停止状态。

6.针对预设场景，程序在运行开始前，需要测试者输入预设场景的文件名，如test.txt，如果不设置预设场景，则输入NO，程序会加载默认的map.txt。

预设场景的格式参考指导书，并以目录下的init.txt为准。

7.如果出现中文乱码的情况，请到窗口->首选项->工作空间中更改编码为utf-8。

**输入说明**

所有有关坐标的记录都是从0开始的，如地图大小为80\*80，则不存在(80,80)这个点。

请求的从控制台输入:

程序可以接受 [CR,(srcX,srcY),(dstX,dstY)]形式的输入，输入之间的空格都会被替换，因此接受含有空格的输入，但数字前不支持+号。本程序一行只能输入一个请求，多余一个整行都作为无效请求。此外，程序接受输入”end”作为整个程序结束的标志，注意大小写。

打开关闭道路的输入:

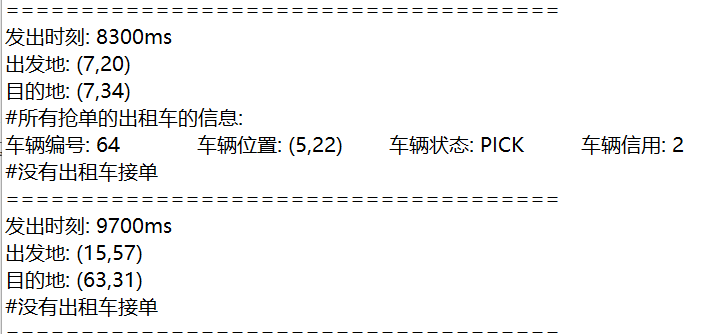
打开道路的指令:[OPEN,(srcX,srcY),(dstX,dstY)]

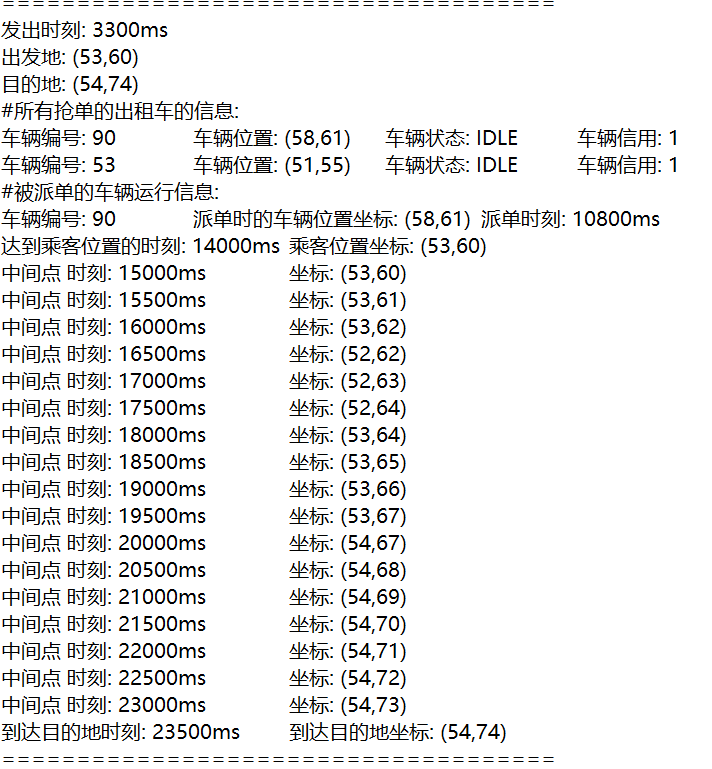
关闭道路的指令:[CLOSE,(srcX,srcY),(dstX,dstY)]

打开关闭道路后需要测试者保证地图的连通性。同时需要测试者保证道路两个点之间有存在边的可能性。

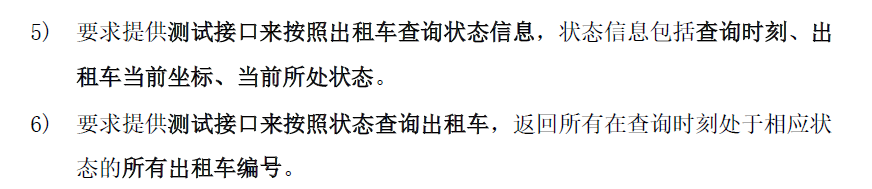
**程序输出说明**

对于每个请求，其处理过程记录在log.txt中，如果像验证请求处理的正确性，请阅读log.txt。下面分别是三个请求的处理过程样例：





**测试接口说明**

****

指导书中所说的提供的测试接口在Scheduler类的最底部，我已经简单的实现了这两个功能，测试者可以看着两个方法的实现过程构造自己的测试方法。

**public** String queryOneTaxi(**int** No)//按照出租车查询状态信息，状态信息包括查询时刻、出租车当前坐标、当前所处状态。

**public** ArrayList<Integer> queryTaxisNowithState(TaxiState ts)//返回所有在查询时刻处于相应状态的所有出租车编号