

README

运行环境

java version : 1.8

IDE : Eclipse Oxygen Oxygen.3 Release (4.7.3)

程序输入及一些细节要求:

- 只允许输入连通的符合要求的地图, 要求地图文件路径为 `D:\\map.txt`。若输入不连通的地图将会在控制台输出信息 `Unconnected Graph` 并直接退出程序。
- 在控制台输入, 格式 `[CR,(x1,y1),(x2,y2)]`, 每一行只能输入一条指令。若不符合该格式或输入的是无效请求(目的地与出发点完全相同, 目的地或出发地超出范围)会提示该条请求无效, 输出 `Invalid Request` 并忽视掉该条请求。不支持前导零以及正负号。
- 请求的同质判断为, 内容完全一样, 同时输入的时间间隔不超过 $100ms$, 并输出 `Same Request`。在实际测试中本程序由于不支持一行多条指令, 使得两条指令的输入间隔有很大可能大于 $100ms$ 。
- 有时候在输入请求时测试会因为输入空格或制表符使得看上去是对的请求被判断错误, 请严格按照规定方法进行测试!
- 租出车状态:
 - `wfs`: 等待服务状态;
 - `service`: 服务状态;
 - `called`: 接单状态;
 - `stop`: 停止状态;
- 出租车线程不会结束, 会一直按照指导书中的要求进行运动。但请求线程在创建 $3s$ 后会自动结束。
- 出租车的初始位置是随机的, 可以同一个位置有多辆车;
- 其他与指导书中相同;
- 不要以GUI中显示的图形界面为判断依据!

测试接口:

- `getTaxiStatus`: 获得组出车当前状态

输出说明:

- 每一个顾客请求都有一个专门的日志: `details[num].txt`, 其中 `num` 代表当前指令为输入的第几条有效指令。所有文件会自动生成都在程序所在的当前项目文件夹中。注意程序结束后和重新运行时各个`details`文件里的内容不会被清空, 需要测试者手动清空。;
- 在 `wfs` 状态中的移动不会被记录到日志中, 但是 `called` 和 `service` 状态中出租车的移动, 会在日志中追加记录;
- 若一辆出租车抢单成功, 那么会在它相应的请求日志中记录下来;
- 若有新的有效请求产生, 将会输出如下示例的内容。
- `Time:1524647785500:New Request:From (23,44) to (54,22)`

- 若某辆车参与了抢单，按照以下格式输出该车信息。
- `Num:56 Position:(67,42) State:wfs Credit:1`
- 以以下字符串为例，说明当出租车抢单成功时控制台与文件中将会输出的内容。
- `"Time:1524647808900:The request is processed by N0.56 taxi." ;`
- 以以下字符串为例，说明当出租车到达乘客出发点与目的地时将会输出的内容。
- `Time:1524647809400:N0.56 Taxi Approach the start point`
- `Time:1524647816100:N0.56 Taxi Approach the destination`
- 以以下字符串为例，被派单的车辆在到达出发点与接到乘客后到达目的地之前在途中经过某点时将会输出的内容。
- `Time:1524647809400:N0.56 Taxi Called From (67,44) to (66,44)`
- `Time:1524647812200:N0.56 Taxi Service:From (61,41) to (60,41)`
- 这里的时间（包括其他字符串里的时间）直接取自系统时间然后向100ms取整，但是程序运行也是要花费时间的，当累计花费的时间到达一定程度时，可能会造成出租车经过两点的时间间隔为300ms。这属于正常误差，请测试者不要将之视为错误。
- `Time:1524647904800:N0.7 Taxi Called From (10,45) to (10,44)`
- `Time:1524647904800:N0.7 Taxi Approach the start point`

派单信息和下面紧跟着的一条途径分支点信息的坐标和时间相同（时间若不同则为系统时间延迟所致，但概率非常小），这里体现出发布派单信息后出租车立刻前往接乘客的设计要求。