

咖喱铁路售票系统curry ticketing system -2

题目背景

售票系统最核心的功能是什么？当然是售票啦！现在请你在CTS-1的基础上实现以下功能：

超级管理员

本着“以抢钱为宗旨，待旅客如孙子”的宗旨，咖喱铁路票价的定价权必须掌握在高层的手中。高层管理人员拥有一句神奇的咒语，向系统中输入这句咒语即可进入管理员模式，进行关键数据的修改。

那么这句咒语就是—— `TunakTunakTun` （译：我在东北玩泥巴）

实现以下功能：

命令	功能描述
TunakTunakTun	从标准模式进入超级管理员模式，并输出 <code>DuluDulu</code>
NutKanutKanut	退出超级管理员模式，并输出 <code>DaDaDa</code>

- 异常处理
 - 若已经在管理员模式下执行 `TunakTunakTun` 或在普通模式下执行 `NutKanutKanut` ，输出 `WanNiBa`

注：以下功能前备注（超级管理员模式）为仅可以在超级管理员模式下执行的命令，（标准模式）为仅可以在标准模式下执行的命令，未标注的命令在两种模式下均可执行。

线路和列车管理

为方便起见，我们假定在黄金右手国，所有的铁路线路都是从首都第三新德里市（Delhi-3）出发，沿自己的线路折返运行，跨线换乘需要在第三新德里市下车并重新购买车票，并且乘客中途到站下车不释放席位。在超级管理员模式下可以向系统中添加新建成的线路列车、指定票价、席位数量信息等操作。

线路管理

- （超级管理员模式）添加和删除线路

命令	参数1	参数2	参数2k+1(k>=1整数)	参数2*(k+1)	功能描述
addLine	线路编号	负载能力	站点名2k+1	里程数2k+1	添加一条新的线路，并添加初始站点，若成功则输出 <code>Add Line success</code> 里程数计算方式：起点第三新德里市（站点0）的里程数为0，每个站点的里程数为该站距第三新德里市的里程（里程数可以相同）。线路编号为字符串类型，负载能力表示该线路上能开行的最大列车数量。
delLine	线路编号				删除线路，并删除在该线路上运行的全部列车。若成功输出 <code>Del Line success</code>

异常处理

- 添加线路时站点名和里程数一一对应，里程数应为整数类型，否则不做任何修改并输出 `Arguments illegal`
- 添加线路时同一线路上的两个站名不得重复（没有环线），否则不做任何修改并输出 `Station duplicate`
- 添加线路时线路编号不得重复，否则不做任何修改并输出 `Line already exists`，删除线路时线路号必须存在，否则输出 `Line does not exist`
- 添加线路时负载能力应该在合理的范围内（不能跑车的铁路修了个寂寞），否则输出 `Capacity illegal`。

（超级管理员模式）添加和删除车站

命令	参数1	参数2	参数3	功能描述
addStation	线路编号	新站点名	里程数	向线路中添加新的站点，若成功则输出 <code>Add Station success</code>
delStation	线路编号	站点名		从线路中删除指定站点，若成功则输出 <code>Delete Station success</code>

异常处理

- 线路编号必须存在，否则不做任何修改并输出 `Line does not exist`
- 添加车站时同一线路上的两个站名不得重复（没有环线），否则不做任何修改并输出 `Station duplicate`；删除车站时车站必须存在，否则不做任何修改并输出 `Station does not exist`
- 添加车站时里程数应当符合规范，否则输出 `Arguments illegal`。

查询线路

命令	参数1	功能描述
lineInfo	线路编号	重写线路的toString方法，使得按照如下格式输出按照里程数升序的站点信息

◦ 输出格式

```
1 | [线路编号] [已承载列车数]/[总最大负载] [站点名1] [里程数1] [站点名2] [里程数2] ...  
   [站点名k] [里程数k]
```

例：

```
1 | Line13 0/4 Dazhongshi:28 Zhichunlu:30 Wudaokou:48 Shangdi:96 Xierqi:121  
   Longze:148
```

◦ 异常处理

- 线路必须存在，否则输出 `Line does not exist`

● 列出全部线路

命令	功能描述
listLine	列出全部线路信息，按格式输出按照线路编号字典序升序排列的站点信息，若为空，则输出 <code>No Lines</code>

● 输出格式

```
1 | [1] [线路编号1] [已承载列车数]/[总最大负载] [站点名1] [里程数1] [站点名2] [里程数2] ...  
   [站点名k] [里程数k]  
2 | [2] [线路编号2] [已承载列车数]/[总最大负载] [站点名1] [里程数1] [站点名2] [里程数2] ...  
   [站点名l] [里程数l]  
3 | ...  
4 | [i] [线路编号i] [已承载列车数]/[总最大负载] [站点名1] [里程数1] [站点名2] [里程数2] ...  
   [站点名m] [里程数m]
```

例：

```
1 | [1] Beihei 5/8 Erjing:16 Erlongshantun:35 Wudalianchi:52 Longzhen:63 Xianghe:81  
   Longmenhe:95 Chenqing:137 Qingxi:167 Sunwu:187 Sunwubei:192 Eyu:218 Hushui:234  
   Xigangzi:257 Sanjitun:282 Jinhe:287 Heihe:302  
2 | [2] Line19 0/1 Mudanyuan:9 Jishuitan:34 Pinganli:50 Taipingqiao:77 Niujie:98  
   Jingfengmen:117 Caoqiao:144 Xinfadi:170 Xingong:198  
3 | [3] Manchu 0/2 Shahe:20 Hamazhen:59 Wulongbei:150 Tangshancheng:207  
   Fenghuangcheng:272 Gaolimen:281 Sitaizi:430 Jiguanshan:498 Benxihu:586
```

列车管理

黄金右手国的列车分为以下三种：

- 普通车：可以挂人的小火车，车次以0开头，出售坐票（CC）、站票（SB）、挂票（GG）
- 国产快车 गतिमान (Gatimaan, “搬家”号)：车次以G开头，出售软座票（SC）、硬座票（HC）、站票（SB）
- 锌淦线 曠野 (Koya, “旷野”号)：象征着和霓虹国友谊的高速铁路，车次以K开头，出售一等座票（1A）和二等座票（2A）

假定旅客列车车次号由1位车次代码（G/K/0）+4位车次数字组成，不同列车的不同坐席有不同的每公里单价，价格可以取任意浮点数，张数可以取任意自然数。

实现以下功能：

- （超级管理员模式）添加删除列车

命令	参数1	参数2	参数3	参数4	参数5	参数6	参数7	参数8	功能描述
addTrain	列车车次	线路号	坐席1 票价	坐席1 张数	坐席2 票价	坐席2 张数	坐席3 票价	坐席3 张数	添加一班列车，并指定线路、各个席别的每公里票价和余票张数。对于普通车，有效参数为18，坐席1、2、3分别为坐票、站票、挂票；对于“搬家”号，有效参数为18，坐席1、2、3分别为软座票、硬座票、站票；对于“旷野”号，有效参数为1~6，坐席1、2分别为一等座票、二等座票。若添加成功，则输出 <code>Add Train Success</code>
delTrain	列车车次								删除列车，若删除列车成功，则输出 <code>Del Train Success</code>

异常处理

依次检验以下错误

- 列车号需符合规范，否则输出 `Train serial illegal`
 - 添加列车时车次号不得重复，否则输出 `Train serial duplicate`，删除列车时车次号必须存在，否则输出 `Train does not exist`
 - 线路号必须存在，且未达到负载上限，否则输出 `Line illegal`
 - 票价具有实际意义，否则输出 `Price illegal`
 - 张数具有实际意义，否则输出 `Ticket num illegal`
- （标准模式）查询火车的余票和票价信息

命令	参数1	参数2	参数3	参数4	功能描述
checkTicket	列车车次	出发站	目的站	席位代号	查询列车余票和票价信息。

输出格式

```
1 | [[列车号]: [出发站]->[目的站]] seat:[席别代号] remain:[剩余票数] distance:[里程数]
    price:[最终票价(2位小数)]
```

例:

```
1 | [K1151: Mudanyuan->Niujie] seat:1A remain:90 distance:89 price:1958.00
```

异常处理

- 列车号需符合规范，否则输出 `Train serial illegal`
- 车次号必须存在，否则输出 `Train serial does not exist`
- 车站必须存在，否则输出 `Station does not exist`
- 席位必须与车次类型对应，否则输出 `Seat does not match`

列出火车信息

命令	参数1	功能描述
listTrain	线路编号	列出某一线路上的全部列车信息，若参数1为空，则列出全部线路上的全部列车信息。若没有火车，输出 <code>No Trains</code>

输出格式

按照如下格式输出列车信息，排序方式为“旷野”号、“搬家”号和普通车，相同类型按字典序排序。若线路上没有列车运行，则输出 `No Trains`

```
1 | [1] [列车号1]: [线路编号1] [[席位代码1]][席位价格1]:[剩余席位数1] ... [[席位代码k]:[席位价格k][剩余席位数k]
2 | [2] [列车号2]: [线路编号2] [[席位代码1]][席位价格1]:[剩余席位数1] ... [[席位代码k]:[席位价格k][剩余席位数k]
3 | ...
4 | [m] [列车号m]: [线路编号m] [[席位代码1]][席位价格1]:[剩余席位数1] ... [[席位代码k]:[席位价格k][剩余席位数k]
```

例:

```
1 | [1] K1151: Beihei [1A]14.00:100 [2A]7.00:190
2 | [2] G1121: Beihei [SC]3.00:90 [HC]2.60:120 [SB]2.40:200
3 | [3] G1151: Beihei [SC]22.00:90 [HC]7.00:120 [SB]2.40:200
4 | [4] G1191: Beihei [SC]3.00:90 [HC]2.60:120 [SB]2.40:200
5 | [5] 01151: Beihei [CC]3.00:100 [SB]7.00:10 [GG]2.40:0
```

异常处理

- 线路必须存在，否则输出 `Line does not exist`

输入错误处理

- 如果存在多种非法情况，按以下顺序进行检查，只输出最先发生的非法信息。
 - 命令是否存在，若不存在输出 `Command does not exist`
 - 对应参数数量和类型是否正确，若不正确则输出 `Arguments illegal`
 - 命令规定的异常处理内容，按照上文描述依次检验处理
 - 其他错误处理，发生未在上述列举出的错误，输出 `Unknown error`