OO-第二次作业-各个类说明

2017/3/13

# 请求类(Request)

功能：处理

类属性：

**static** **final** **int** ***max\_time*** = 2147483647;//可以读到的时间的最大数

**private** **int** maxFloor, minFloor;//表示读入的楼层允许的最大值和最小值

**private** **int** i, j, l, type, num1, num2, updown;

//i,j:循环变量 l:字符串的长度

//type:该行读入的类型(FR还是ER) num1:读入的第一个数(楼层)

//num2:读入的第二个数(类型) updown:上下信息

**private** **boolean** sign;//标记这个表达式是否合法

**private** String str;//读入的字符串

**private** **char**[] strArr;//把读入的字符串转化为的字符数组

方法：

Request(String s,**int** min\_f,**int** max\_f);//构造函数，初始化一些变量

**void** work();//主要的处理字符串过程，判断其是否合法，并保存信息到布尔变量sign

**boolean** getsign();//返回sign，若为true代表这行没有语法错误

**int** getnum1();//返回表达式的第一个数，为楼层信息

**int** getnum2();//返回表达式的第二个数，为时间信息

**int** getupdown();//返回是上还是下

**int** gettype();//返回时FR还是ER

电梯、楼层、请求队列、调度器、请求。

# 请求队列(RequestQueue)

类属性：

**static** String *str*;//读入一个字符串

**static** ArrayList *list\_t* = **new** ArrayList();//存储所有合法的时间信息的Arraylist

**static** ArrayList *list\_f* = **new** ArrayList();//存储所有合法的楼层信息的Arraylist

**static** ArrayList *list\_type* = **new** ArrayList();//存储所有合法的类型信息的Arraylist

**static** ArrayList *list\_updown* = **new** ArrayList();//存储所有合法的上下信息的Arraylist

**static** **int** *last\_t*,*time*,*floor*,*i*,*tot*,*last\_floor*,*updown*,*last\_updown*,*type*,*last\_type*;

//last\_t 上一个请求的时间 time：当前时间 其它类似

**static** **boolean** *flag*;//当前读入的这一行是否合法

**static** **int** *minFloor*,*maxFloor*;//表示读入的楼层允许的最大值和最小值

方法：

RequestQueue(**int** min\_f, **int** max\_f);//构造函数，初始化minFloor,maxFloor

**void** work(){//逐行读入，直至出现，调用Request来判断是否是合法的。并判断是不是满足时间的递增的要求

**int**[] getArray\_time(){//返回所有的合法的输入的时间信息，以数组形式

**int**[] getArray\_floor(){//返回所有的合法的输入的楼层信息，以数组形式

**int**[] getArray\_updown(){//返回所有的合法的输入的上还是下信息，以数组形式

**int**[] getArray\_type(){//返回所有的合法的输入的类型(FR还是ER)信息，以数组形式

# 电梯类(Elevator)

类属性：

**final** **static** String[] ***s*** = {"UP","DOWN","STILL"};

**private** **int**[] time;//保存所有合法的时间的数组

**private** **int**[] floor;//保存所有合法的楼层的数组

**private** **int**[] type;//保存所有合法的类型的数组

**private** **int**[] updown;//保存所有合法的上下的数组

**private** **int** n, i, j, last\_time;//n:所有合法输入的个数

**int** now\_f;//当前的行数

**double** now\_t;//当前的时间

**public** Elevator(**int**[] t, **int**[] f,**int**[] typ,**int**[] upd) {//构造函数

**void** work(){//求出输出结果

**static** **int** myabs(**int** t);{//求一个数的绝对值

方法：

**public** Elevator(**int**[] t, **int**[] f,**int**[] typ,**int**[] upd) {//构造函数

**void** work(){//求出输出结果

**static** **int** myabs(**int** t);{//求一个数的绝对值

# 调度类(Dispatch)

方法：

void work(int min\_f,int max\_f){//构造一个请求队列的对象。并把其读入的合法信息放入电梯类进行计算

# 楼层类(Floor)

方法：

Floor(**int** min\_f,**int** max\_f){//构造函数

**int** getMin();//得到楼层允许的最大值

**int** getMax();//得到楼层允许的最小值

# 主类(Main)

方法：

**public** **static** **void** main(String[] args);//构造一个楼层类的对象，并把允许的最大值传入一个调度类的对象。