所有类

request、request\_list、elevator、scheduler、als\_scheduler、floor

接口

abstract\_elevator

类

request

属性：

String[] req; //将完整请求以字符数组形式保存

double req\_t; //请求发出的时间

double finish\_time; //在调度时请求完成的时间

boolean judge; //请求是否有效

int des\_floor //请求的目标楼层

String req\_str //储存原始的请求字符串

方法：

\*构造方法

request(String str) //初始化属性，将字符串型的请求转换成数组保存

\*所有方法

限定符和类型 方法和说明

double get\_finish\_time() //返回请求完成的时间

double get\_req\_t() //返回请求发出的时间

String[] get\_request() //返回保存完整请求内容的数组

boolean judge() //返回judge

void set\_finish\_time(double t) //设置请求完成时间

void check\_str(str) //检查请求是否格式正确

String get\_req\_str() //返回原始请求字符串

Int get\_des\_floor() //返回请求目标楼层

String get\_req\_class() //返回请求类型

String get\_req\_direction() //返回请求方向 如果是ER类型返回“ER”

String toString() //返回重组的请求字符串

类

request\_list

属性：

Vector<request> req\_list; //存储有效请求的队列

方法：

\*构造方法

request\_list() //为req\_list分配空间

\*所有方法

限定符和类型 方法和说明

void add\_req(request req) //将有效请求加入请求队列并对无效请求输

出提示信息

request get\_req(int i) //返回请求队列中的第i+1条请求

int left\_req() //返回请求对列中的请求总数

void remove\_req(int i) //移除请求队列中的第i+1条请求 用于移除

相同请求

类

elevator

属性：

int floor; //电梯所在的楼层

String direction; //电梯的运动方向

方法：

\*构造方法

elevator() //初始化电梯属性 最开始楼层数为1 运动方向为STILL

\*所有方法

限定符和类型 方法和说明

String get\_direction() //返回电梯的运动方向

int get\_floor() //返回电梯所在的楼层

void set\_floor(int f) //使电梯由当前楼层运动到f楼层 并根据两

个楼层的关系确定电梯运动方向

String toString(int t) //重载toString() 使得电梯输出(e\_n,sta,t)状态

类

scheduler

属性：

double t; //时间

int iterator; //请求队列中的当前请求序号

static Scanner s\_in //进行输入

方法：

\*构造方法

scheduler() //初始化属性 t为0 iterator为0

\*所有方法

限定符和类型 方法和说明

void command(elevator ele, request\_list req\_list) //从请求队列中

不断取出请求，并调用schedule函数来调度电梯

void schedule(elevator ele, request req, request\_list req\_list) //判断相同请求

并调度电梯运行，完成请求后输出

String Make\_String(String[] req) //将请求由数组转换成字符串

类

als\_scheduler

属性：

double t; //时间

int iterator; //请求队列中的当前请求序号

static Scanner s\_in //进行输入

方法：

\*构造方法

als\_scheduler() //初始化属性

\*所有方法

限定符和类型 方法和说明

void command(elevator ele, request\_list req\_list) //从请求队列中

不断取出请求，并调用schedule函数来调度电梯

void schedule(elevator ele, request req, request\_list req\_list) //判断相同请求

并调度电梯运行，完成请求后输出

String Make\_String(String[] req) //将请求由数组转换成字符串

static void main(String[] args) //主函数 完成输入

接口

abstract\_elevator

方法：

String get\_direction() //返回电梯的运动方向

int get\_floor() //返回电梯所在的楼层

void set\_floor(int f) //使电梯由当前楼层运动到f楼层 并根据两

个楼层的关系确定电梯运动方向