关于电梯调度各类的说明文档

# 电梯类（Elevator）

电梯自身的属性有所在楼层和运动状态，且这两个状态均为private变量。其中所在楼层为int型，运动状态共有UP,DOWN,STAY三种。

电梯的默认构造方法将楼层设置为1，这是因为电梯初始状态在1楼的原因。

电梯有getFloor和getState方法，其他类可以通过这两个方法获得电梯的属性。

电梯有一个move方法，它的参数是一个请求，输出为完成这个请求所化的时间，格式为double，并将自身所在的楼层设置为完成请求后的楼层。这个方法的作用是，当dispatcher进行电梯调度时，他将request传给电梯，电梯收到请求并结合自身的状态计算出所需要的时间和电梯的运动状态，并回传给调度器类。

# 请求类（Request）

请求自身有四个属性，时间，目标楼层，请求方向和请求类型。

时间为private整形变量，因为所有合法的时间输入都是非负整数。

目标楼层为private整形变量。

请求方向可以为UP, DOWN，type分为0和1，分别对应ER和FR类型的请求。

请求类可以用以上四个属性可以分别用getTime ,getDfloor, getDirection, getType来获得。

Request的构造方法为Request(int dfloor,int time,int type,int dir)

当type == 0，即输入ER指令时，direction默认设为null；否则分配方向。

# 请求队列类（RQueue）

此类主要用来建立并维护请求队列。主体为一个Request的数组，包括一个rindex用来在新建队列的时候记录request个数的多少。

GetRequest（int i）和getRindex()方法可以获得队列中

其中的addrequest(Request rq)方法可以将传入的rq加载到队列末尾。

# 调度器类（dispatcher）

调度器类是这几个类中任务最为繁重的一个，他负责识别输入生成请求，并将生成的请求加入请求队列中。当完成输入后，他负责根据请求来进行电梯调度，并输出调度的结果。

调度器类自己新建了一个请求队列用来保存将来可能会生成的请求。限定MAX\_FLOOR为10,限定初始时间为timenow为0.

调度器类的buildQueue方法用来建立新的队列，其建立请求的过程主要通过字符逐个识别，具体过程与第一次作业相仿。

调度器类的dispatch方法通过新建一个电梯对象eva1来模拟电梯的运行过程。在这个方法中通过eva1的move方法来获得每次运动所花费的时间，并且通过综合考虑现在的时间与请求的时间的大小来计算请求完成时的时间是多少。并在计算结束后进行输出。