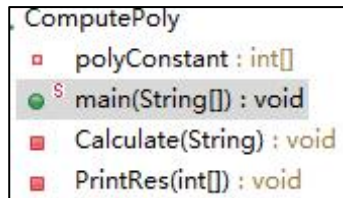


第四次博客作业

一、三次作业程序结构分析

(1) 多项式加减



类：1 个。

属性：1 个。

方法：3 个。

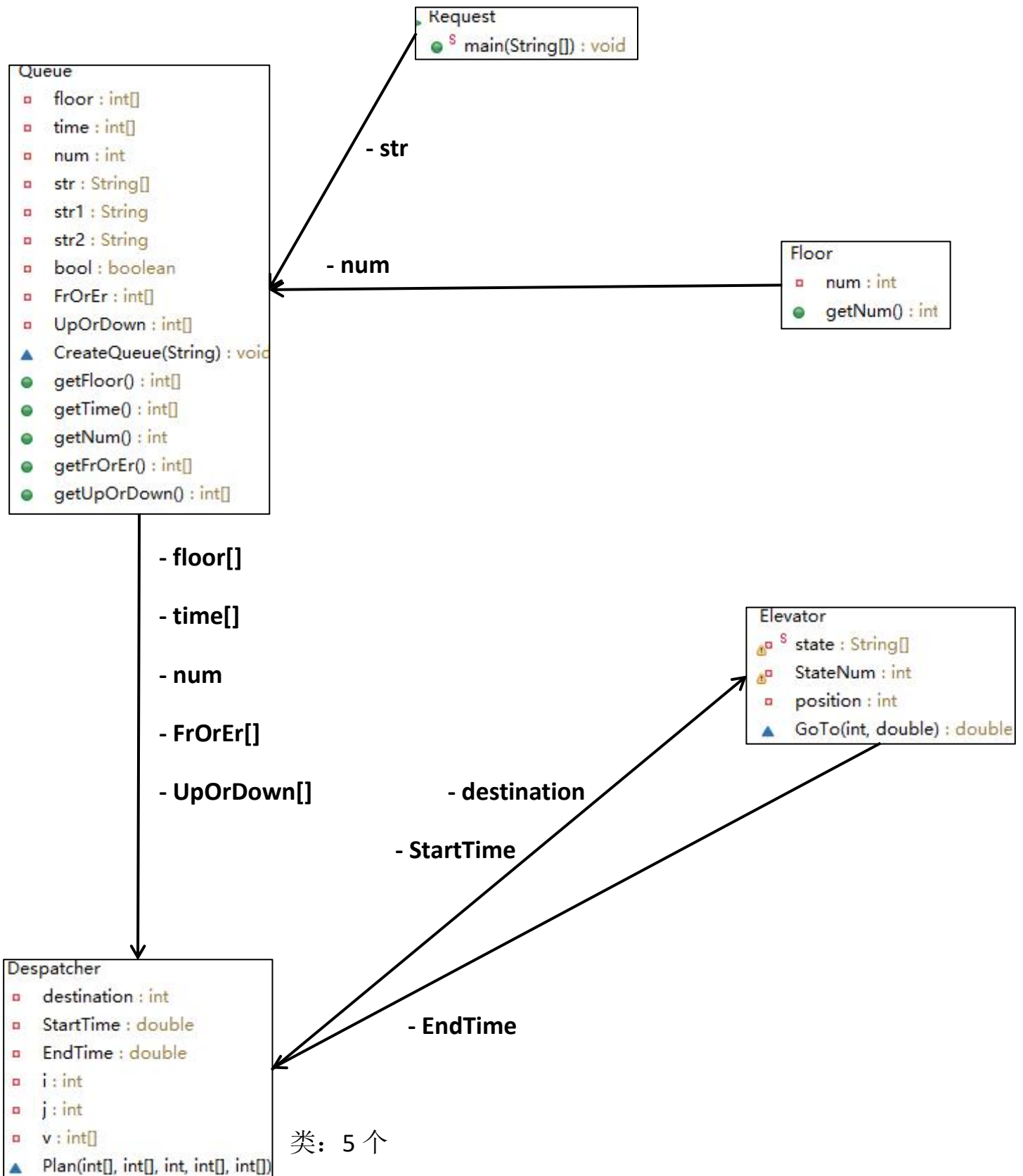
ComputePoly 类：

main 方法：36 行

Calculate 方法：46 行

PrintRes 方法：15 行

(2) 傻瓜电梯调度



Request 类（50 行）：

属性：1 个

方法：1 个（main 42 行）：

Queue 类（90 行）：

属性：9 个

方法：6 个（CreateQueue 38 行，还有 5 个 get、set 方法）

Despatcher 类（40 行）：

属性：6 个

方法：1 个（Plan 30 行）

Elevator 类（36 行）：

属性：3 个

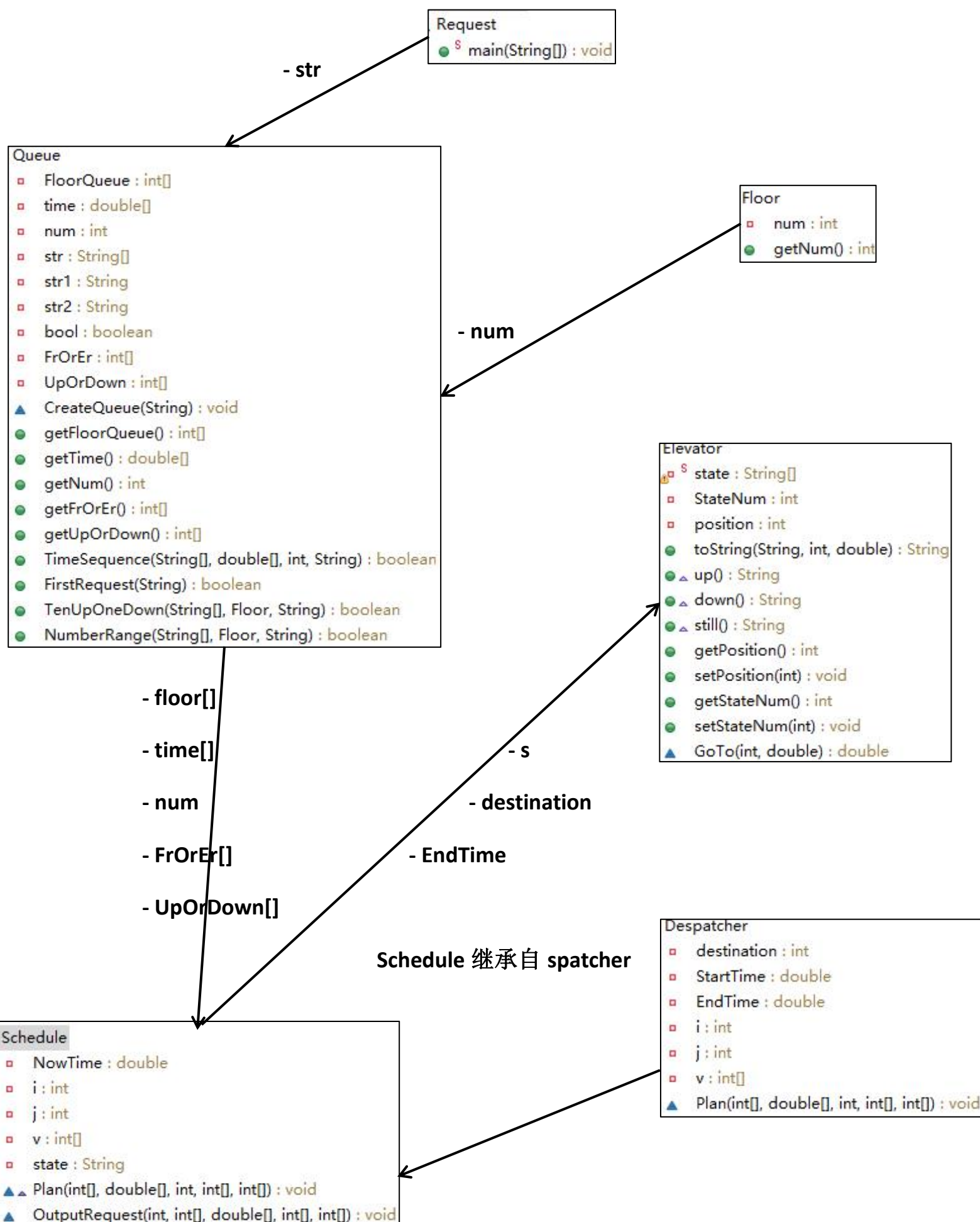
方法：1 个（GoTo 30 行）

Floor 类（11 行）：

属性：1 个

方法：1 个（一个 get 方法）

(3) ALS 单电梯调度



类：6 个

Request 类（50 行）：

属性：1 个

方法：1 个（main 42 行）：

Queue 类（134 行）：

属性：9 个

方法：10 个（CreateQueue 38 行，TimeSequeue 11 行，FirstRequest 9 行，TenUpOneDown 10 行，NumberRange 22 行，还有 5 个 get、set 方法） CreateQueue 调用了 TimeSequeue 、 FirstRequest 、 TenUpOneDown、 NumberRange。

Despatcher 类（40 行）：

属性：6 个

方法：1 个（Plan 30 行）

Elevator 类（36 行）：

属性：3 个

方法：8 个（GoTo 30 行，up 3 行，down 3 行，still 3 行，还有四个 get、set 方法）

Floor 类（11 行）：

属性：1 个

方法：1 个（一个 `get` 方法）

Schedule 类（229 行）：

属性：5 个

方法：2 个（Plan 200 行，OutPutRequest 10 行）

总评：第一次作业我只写了一个类多个方法，这可能导致方法之间的联系过于紧密，将来对类进行修改时可能会比较麻烦。

第三次作业中，我在重写 Schedule 类中父类 Plan 方法时，没有规划好而将许多功能合并到一起，导致 Plan 方法竟然有 200 行，而且其中有大量重复使用的代码，而我也没有将其封装为一个单独的方法。三次作业中，我没有通过第二次的三个对于 `int` 类型范围限制的测试点，因为我在第二次作业中限定输入最大为 8 位数，这确实是有偷懒的嫌疑，主要是害怕被人测试时超过 `int` 类型。除此之外我还因为粗心而错了两个公测。

三次作业我的优点应该在勤写注释和命名规范。只写代码不写注释是非常坑人的行为，这一点我深有体会，如果没有注释，哪怕对于自己很久之前写的代码也需要很长的时间才能看懂，更何况是别人来看。除此之外，掌握一种命名规范可以很好的帮助阅读者理清代码

逻辑，而不用四处找一个变量的出现位置。

在发现他人 **bug** 方面，主要是阅读代码，根据代码来针对性地测试，尤其是对其中的数组是否越界、变量是否超过大小，以及自己在编写时曾经犯过的错误，进行重点分析。

最后讲讲我测别人程序时发现的几个问题，首先有同学写了上千行代码，结果只有一个 **.java** 文件，还有些同学只写代码不写注释。这些都是小事，但是会给他人和自己的阅读造成较大的困难，希望有这些习惯的同学能改正。