# 程序设计方法
# 课后作业 #3

## 作业提交时间:2019-04-18 上课前

**注意事项:** 在本作业中请务必做到如下几点:

- 书写工整
- 给出详细的答题思路和解题步骤
- 严禁抄袭，如被发现则本次作业记为零分

## 请阅读课本和参考资料回答以下几个问题，并适当的给出 UML 类图;

### (1). Strategy 模式 (20pts)

现在需要实现一个用于模拟和研究机器人交互的应用程序。首先创建一个简单的应用程序来模拟机器人所存在的运动场。程序中有以下几个类：

**IBehaviour**（策略虚类）：

**1.**界定机器人行为的接口，内部有一个方法 **public int moveCommand();**

**2.**具体的策略为以下一些：进攻行为**(AgressiveBehaviour)**，防御行为**(DefensiveBehaviour)**，休息行为**(NormalBehaviour)**；这里每一个策略代表一种特别的行为。为了决定机器人的行为，该类需要获取从传感器获得例如位置，障碍物等信息。

**Robot(**机器人类**)-** Robot 类可以保存或获取诸如位置，障碍物等信息，并将必要的信息传递给 **IBehaviour** 类。

在应用程序的主要部分中首先创建几个机器人，并创建了几个不同的行为。每个机器人都有不同的行为分配：“大机器人”是一个攻击性的攻击机器，攻击任何其他发现的机器人，“乔治 **v.2.1**”真的很害怕，当它遇到另一个机器人时相反的方向逃跑，**'R2'**很漂亮冷静并忽视任何其他机器人。在某些时候，每个机器人的行为都会改变。

**(2). Façade 模式和 Adapter 模式 (20pts)**

请解释 **Façade** 模式和 **Adapter** 模式的区别，请举例在什么情况下更适合使用 **Façade** 模式而不是 **Adapter** 模式，什么情况下更适合使用 **Adapter** 而不是 **Façade** 模式？**(20pts)**（不少于 200 个汉字，举例请写 Java 程序或者 C++程序，否则本题得零分）

**(3). Observer 模式**

**Java** 语言中可以利用 **java.util.Observable** 包来实现 **Observer** 模式，请参照该网址**(http://www.apihome.cn/api/java/Observable.html)**，请设计如下应用场景：某网站出售的一个畅销商品 **popularProduct** 经常脱销，因此很多客户希望当该产品的数量大于等于 **1** 的时候能够立刻获得通知消息，通知消息可以为一个 **string**，其内容是**"The product is available"**，请实现这个功能画出 **UML** 图并附上你开发的代码,要求该代码可成功执行，并假设仅存在以下三个客户，**user1,user2,**和 **user3**。**(20pts)**，

**(4)**请解释 **Observer** 模式的设计目的，在什么情况下 **Observer** 模式会增加程序的复杂度因而不推荐使用？**(10pts)**（不少于 150 个汉字，举例请写 Java 程序或者 C++程序，否则本题得零分）