**实验3 结构型设计模式实验**

**一、实验目的**

熟练使用PowerDesigner和任意一种面向对象编程语言实现几种常见的结构型设计模式，包括适配器模式、组合模式，理解每一种设计模式的模式动机，掌握模式结构，学习如何使用代码实现这些模式。

**二、实验内容**

1. **“喜羊羊逃命”游戏：**喜羊羊被灰太狼追，喜羊羊最多5条命，灰太狼每咬到喜羊羊一次，喜羊羊就要少一条命。在逃的过程中喜羊羊可以吃到三种苹果，吃“红苹果”可以给喜羊羊加上保护罩，吃“绿苹果”可以加快喜羊羊奔跑速度，吃“黄苹果”可以使喜羊羊趟着水跑。应用装饰模式，用java控制台应用程序实现该设计。绘制该模式的UML图。

提示：这个例子如果用类的继承来实现的话那可就麻烦了，你需要为喜羊羊派生3\*2\*1=6个子类（有保护罩的喜羊羊，奔跑速度加快的喜羊羊，会趟水的喜羊羊，既有保护罩又会趟水的喜羊羊，奔跑速度快且会趟水的喜羊羊，有保护罩且奔跑速度快的喜羊羊，有保护罩、奔跑速度快且会趟水的喜羊羊），如果使用装饰模式的那就不用派生诸多子类了，当喜羊羊每吃到一个苹果，我们就用装饰模式给喜羊羊加一个动态增加一个新功能即可。

2. 现有一个接口DataOperation定义了排序方法sort(int[]) 和查找方法search(int[], int)，已知类QuickSort的quickSort(int[])方法实现了快速排序算法，类BinarySearch 的binarySearch(int[], int)方法实现了二分查找算法。试使用适配器模式设计一个系统，在不修改源代码的情况下将类QuickSort和类BinarySearch的方法适配到DataOperation接口中。绘制类图并编程实现。（要求实现快速排序和二分查找，使用对象适配器实现）

3. 在某应用软件中需要记录业务方法的调用日志，在不修改现有业务类的基础上为每一个类提供一个日志记录代理类，在代理类中输出日志，例如在业务方法method()调用之前输出“方法method()被调用，调用时间为2014-11-5 10:10:10”，调用之后如果没有抛异常则输出“方法method()调用成功”，否则输出“方法method()调用失败”。在代理类中调用真实业务类的业务方法，使用**代理模式**设计该日志记录模块的结构，绘制类软图并编程模拟实现。

4. 使用组合模式设计一个杀毒软件(AntiVirus)的框架，该软件既可以对某个文件夹(Folder)杀毒，也可以对某个指定的文件(File)进行杀毒，文件种类包括文本文件TextFile、图片文件ImageFile、视频文件VideoFile。绘制类图并编程模拟实现。

5. 某教育机构组织结构如下图所示：



在该教育机构的OA系统中可以给各级办公室下发公文，试采用组合模式设计该机构的组织结构，绘制相应的类图并编程模拟实现，在客户端代码中模拟下发公文。

**三、实验要求**

1. 结合实例，正确无误地绘制适配器模式、组合模式和外观模式的模式结构图；

2. 使用任意一种面向对象编程语言实现适配器模式、组合模式和外观模式实例，代码运行正确无误。

**四、实验步骤**

1. 结合实例，使用PowerDesigner绘制适配器模式实例结构图并用面向对象编程语言实现该模式实例；

2. 结合实例，使用PowerDesigner绘制适配器模式实例结构图并用面向对象编程语言实现该模式实例；

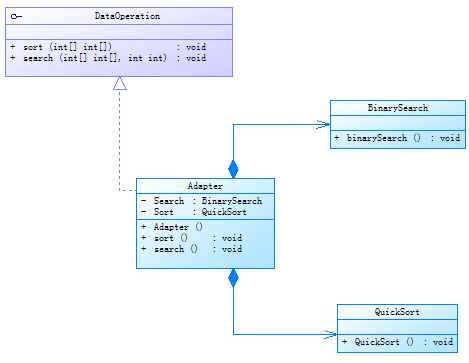
3. 结合实例，使用PowerDesigner绘制组合模式实例结构图并用面向对象编程语言实现该模式实例；

4. 结合实例，使用PowerDesigner绘制组合模式实例结构图并用面向对象编程语言实现该模式实例；

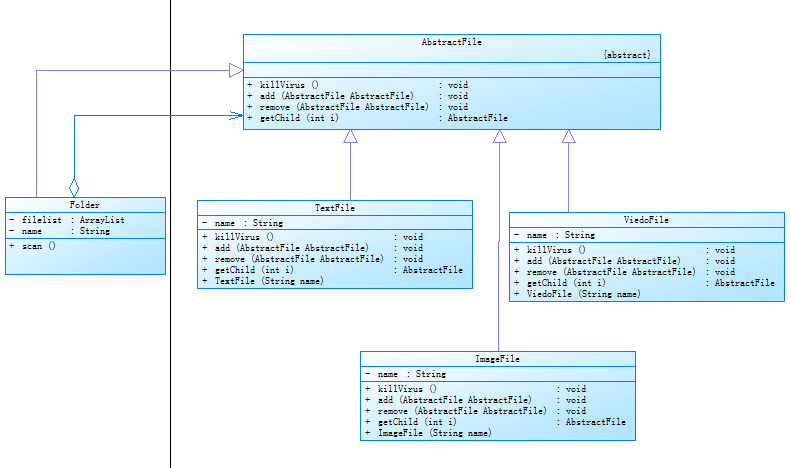
**五、实验结果**

1.装饰模式：

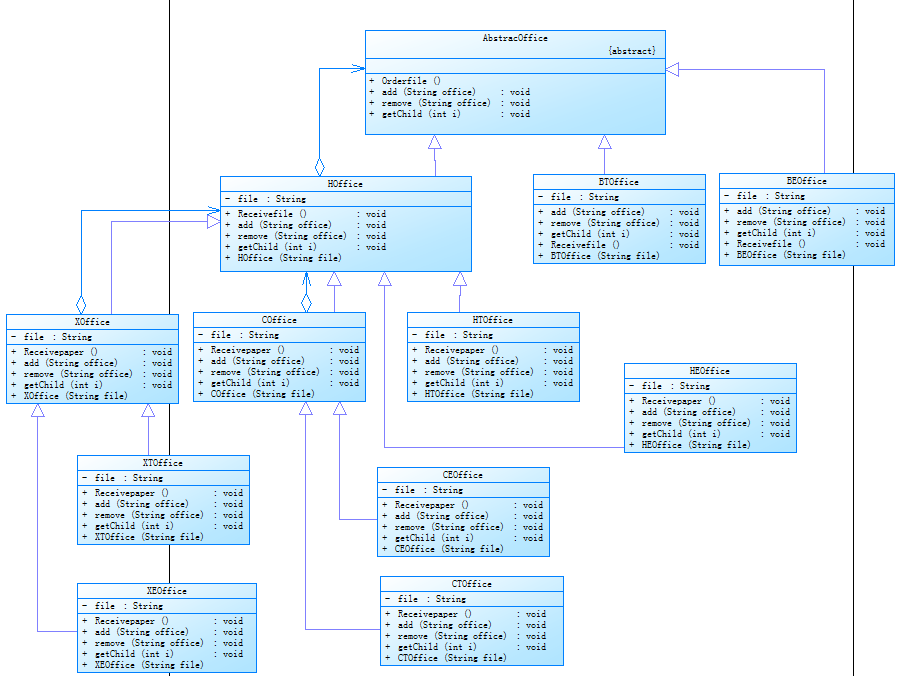
2.适配器模式：



3.代理模式：

4.组合模式：

5.组合模式：



**六、实验小结**