典型的抽象构件角色代码：

```java
public abstract class Component{
  public abstract void add(Component c);
  public abstract void remove(Component c);
  public abstract Component getChild(int i);
  public abstract void operation();
}
```

典型的叶子构件角色代码：

```java
public class Leaf extends Component{
  public void add(Component c){
    //异常处理或错误提示
  }
  public void remove(Component c){
    //异常处理或错误提示
  }
  public Component getChild(int i){
    //异常处理或错误提示
  }
  public void operation(){
    //实现代码
  }
}
```

典型的容器构件角色代码：

```java
public class Composite extends Component{

  private ArrayList list = new ArrayList();

  public void add(Component c){
    list.add(c);
  }

  public void remove(Component c){
    list.remove(c);
  }

  public Component getChild(int i){
    (Component)list.get(i);
  }

  public void operation(){
    for(Object obj : list){
      ((Component)obj).operation();
    }
  }
}
```