# 棋类对战系统大作业实验报告

## 设计思路

**首先进行需求分析。围棋和五子棋共同的功能有：**

局面保存：保存当前的游戏类型等游戏局面

局面读取：恢复最新保存的局面

选择游戏类型

设定棋盘大小（8-19）

重新开始游戏

悔棋一步：只能一步一步地悔，注意处理特殊情况：无棋可悔

投负：比赛结束立即结束。

**围棋的其他功能有：**

落子位置不合法：落子后，看是否一片子都无气了，如果是就不能下在这里

提子

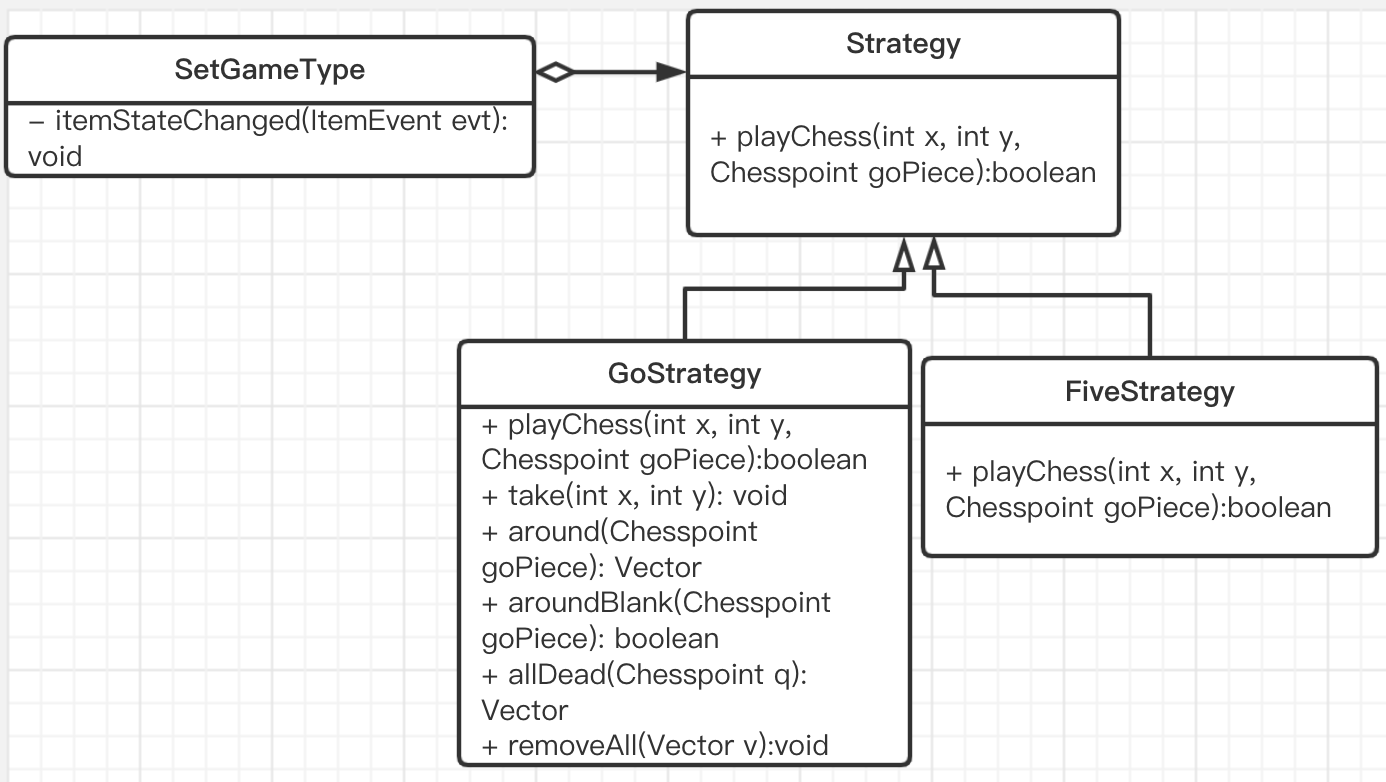
虚着

**五子棋**

判断胜负

**设计模式、设计原则与UML图**

选择游戏类型以及五子棋、围棋下棋的不同规则，可以采用简单工厂模式和策略模式。

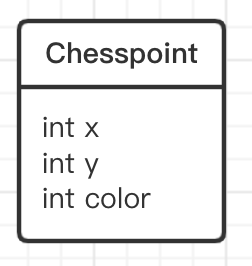


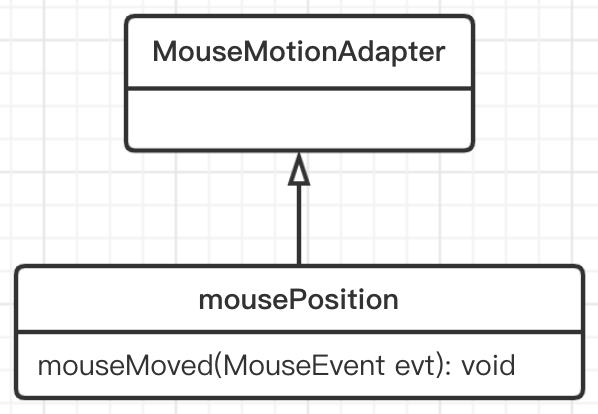
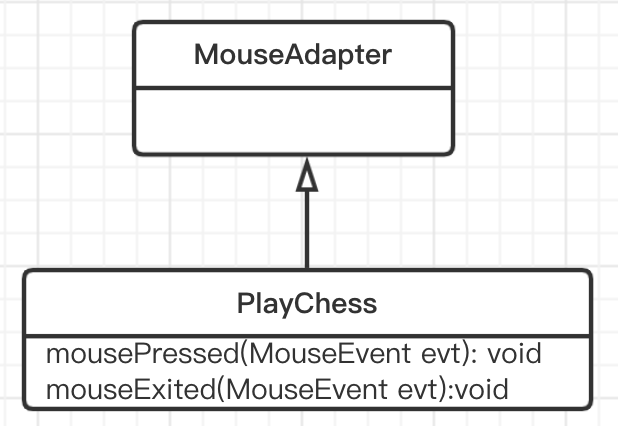
里氏代换原则：所有引用其（基类）父类的地方都能够透明的引用其子类。

依赖倒置原则：抽象不应该不依赖于细节，细节应该依赖于抽象。针对接口编程，而不是针对实现编程。

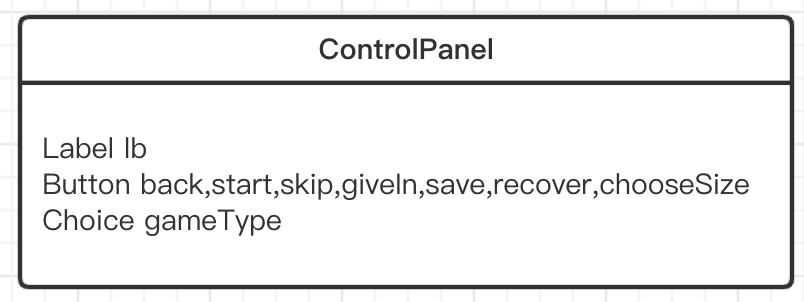
合成复用原则：尽量使用对象组合，而不是继承来达到复用的目的。

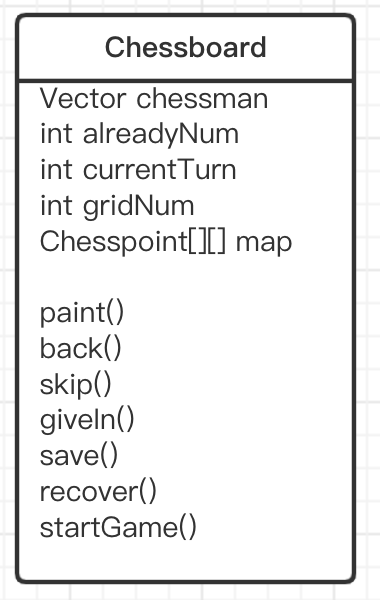
**关键类和函数功能说明**

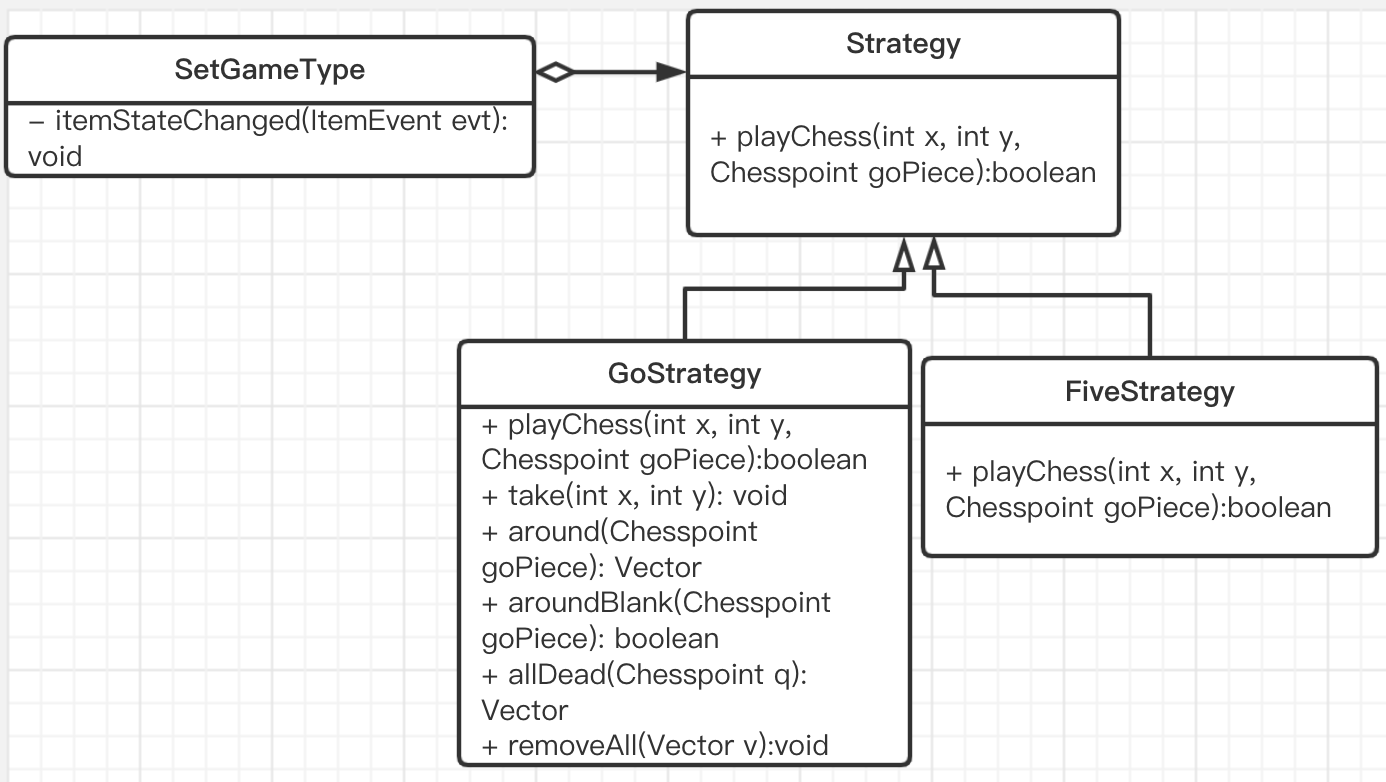
Chesspoint表示棋子类，有坐标和颜色三个参数。



PlayChess类继承MouseAdapter，监听鼠标点击事件。mousePressed方法中调用strategy.playChess方法，不同棋类使用不同策略。  
MousePosition类继承MouseMotionAdapter，监听鼠标移动事件，进行落子位置的提示。

ControlPanel是控制栏类，包括标签、多个按钮和选择棋类型的选择标签。

Chessboard类存储当前棋局的基本信息，比如已经下的棋子Vector chessman，棋子在棋盘上的位置Chesspoint[][] map，当前轮到哪方下currentTurn，下了多少手alreadyNum等等。同时负责绘制棋盘paint()。悔棋、投负、保存棋局、恢复棋局等不同棋类的共同功能，也在其中实现。



GoStrategy为围棋策略，take方法为进行提子操作，需要通道aroundBlank, allDead, removeAll等方法。playChess为父类的方法，实现鼠标点击落子后围棋的操作，包括判断是否有气、是否提子等。

FiveStrategy为五子棋策略，playChess方法中与围棋不同的是要进行胜负判断。

**测试输入和输出**

