**共同工作**

**项目设计目标:小组认为该项目应做到游戏性，美观性，合理性的目标。**

游戏性：完成基本的象棋功能，即在完成基本的正确的象棋的移动，将军，吃子，绝杀等判定的基础上，再加入一个有特殊功能的按钮。

美观性：游戏要做到有自己的UI界面，同时有合适的，美观的棋子，棋盘，点击效果，落子效果，按钮的样式和排列以及背景图。

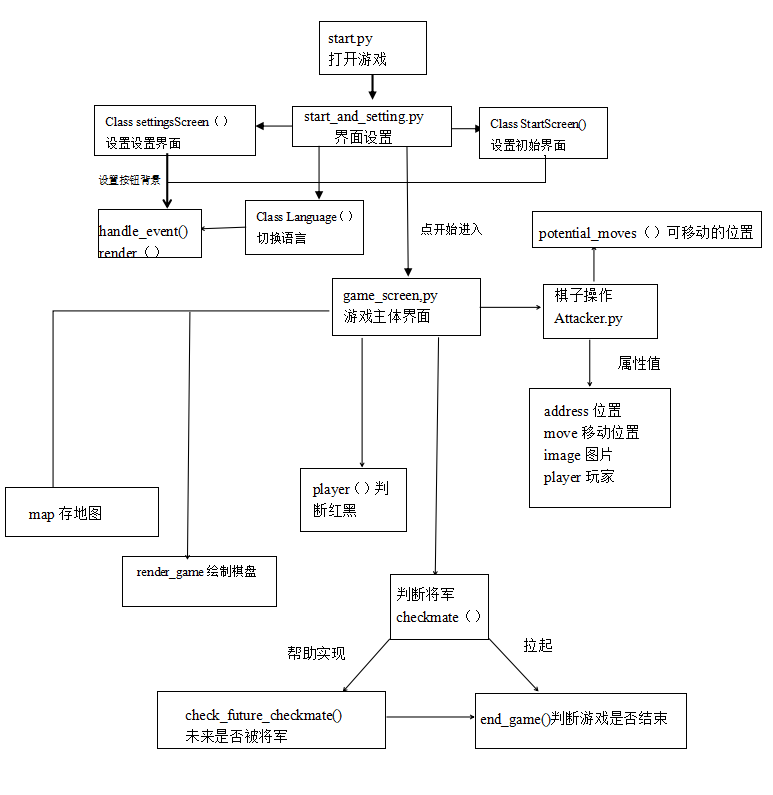
合理性：即符合当前市面上的象棋软件样式，本小组通过考察市场上象棋软件竞品以及询问象棋软件使用者来规范设计样式。使得本小组的象棋软件符合象棋软件的普遍形制。（灵感源于小组成员有参加智运会象棋项目遇上留学生象棋选手）设计有切换英文的功能，能使外国人也能适应该象棋软件。

**项目分工：**

俞乐楠：负责整体的统筹规划，制订代码基本思路，编写代码框架，进行游戏测试，Bug的调试与修复

袁昊旻：负责制订代码的编写，变量的命名规范，负责编写各个棋子对象的功能模块，以及进行UI的调整测试。

傅裕翔：负责编写棋盘显示区域以及绝杀的判定函数和“万宁象棋”功能，负责各阶段汇报的ppt制作

**代码总体框架：**

本项目由start.py, start\_and\_setting.py, game\_screen.py, attacker.py四个主要的文件组成以及image图片组成。其中这四个程序分别代表开始游戏，调设置、英汉，游戏界面，棋子移动。通过各段程序间的联动实现象棋功能，UI设计，行棋/绝杀提示以及中英文切换功能。总共代码行数约1200+行。

**第三方库介绍：**

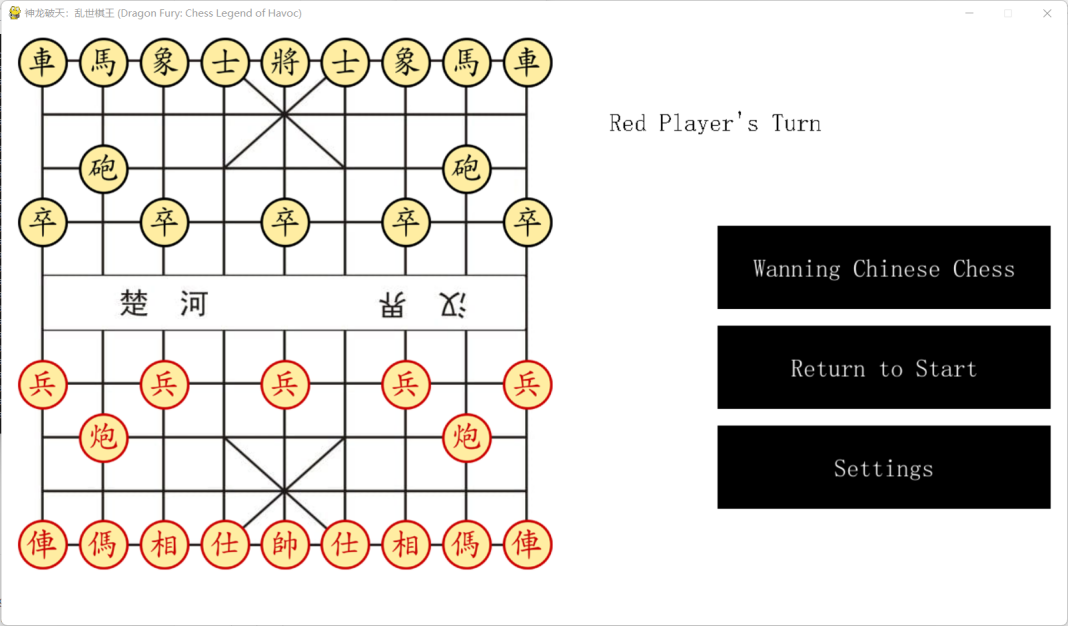
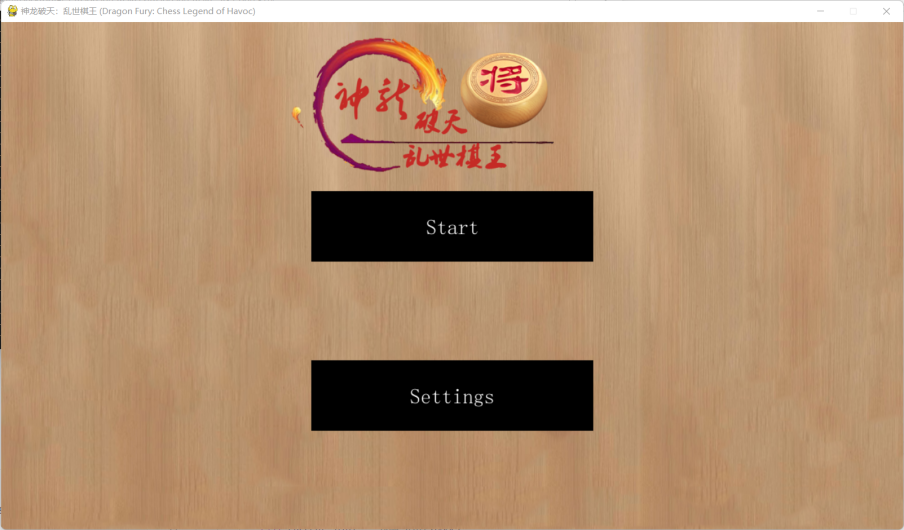
Pygame:Pygame是一个用于开发2D游戏的Python库。它建立在Simple DirectMedia Layer (SDL)库之上，提供了一系列功能丰富的工具和函数，用于创建游戏、图形和多媒体应用程序。提供了处理图像、声音、输入设备和网络等方面的功能。允许创建游戏窗口、渲染图像、处理用户输入以及播放声音和音乐。可以轻松处理碰撞检测、动画、粒子效果和物理模拟等游戏开发中常见的任务。

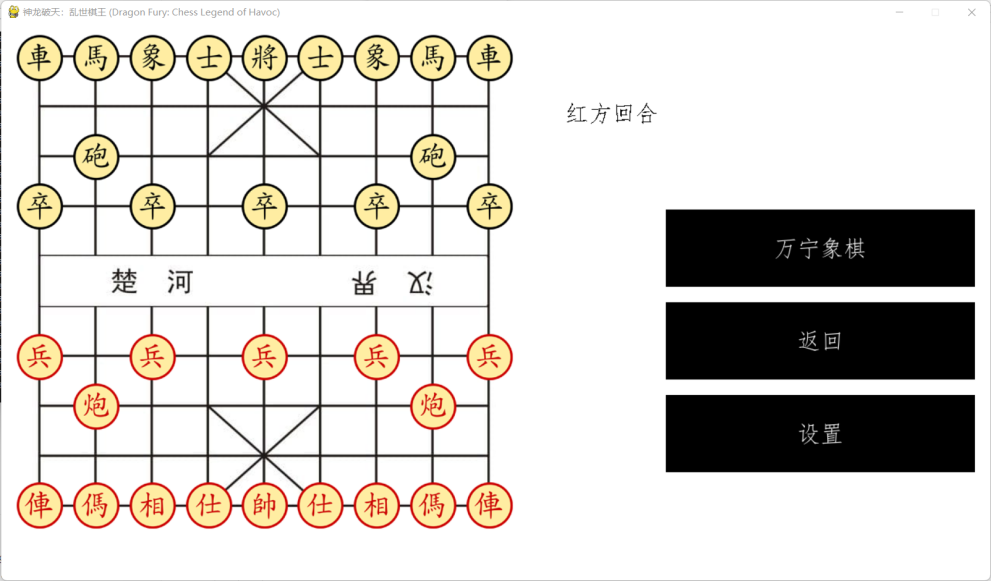
跨平台支持，可以在Windows、Mac和Linux等操作系统上运行。

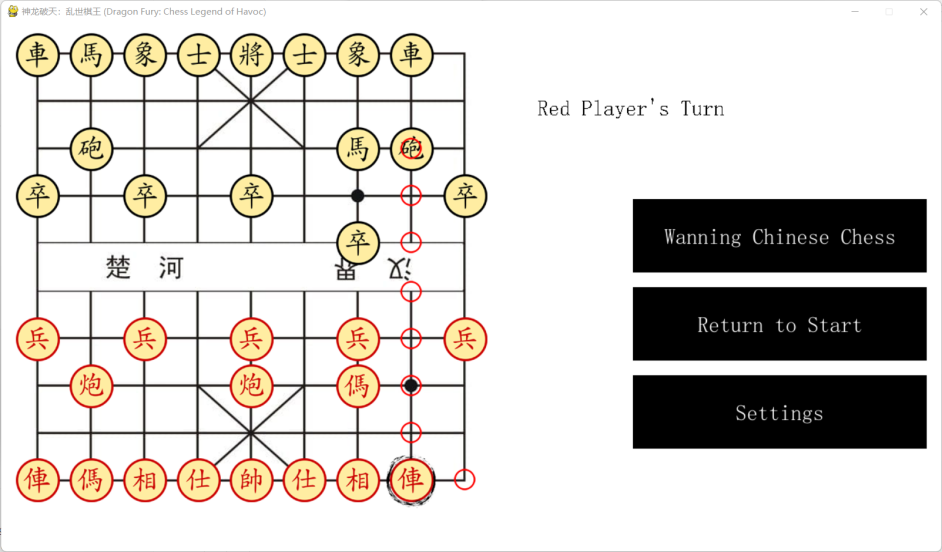
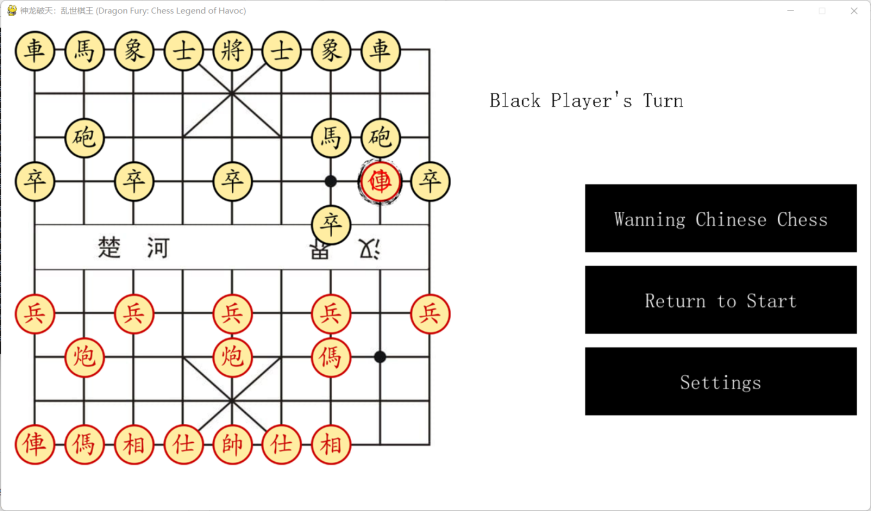
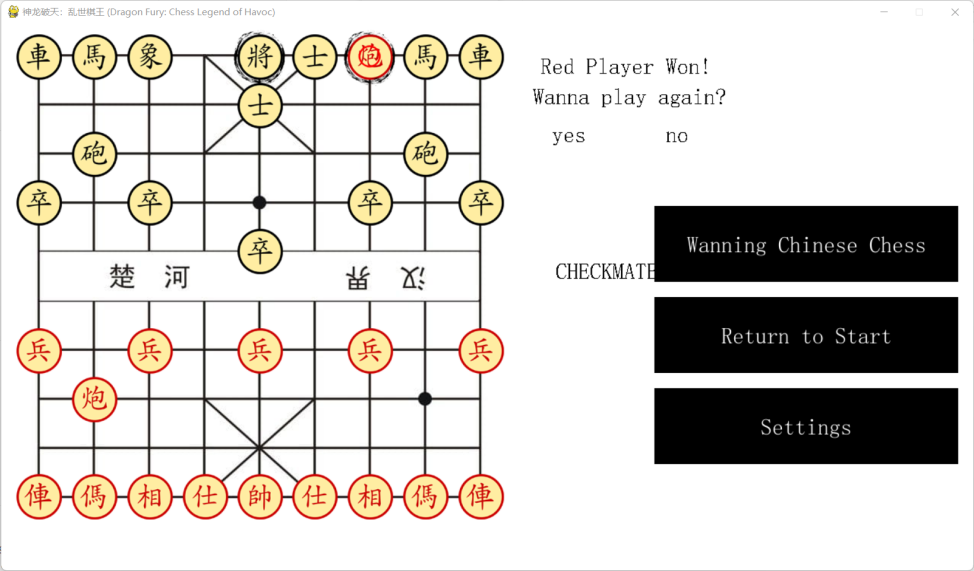
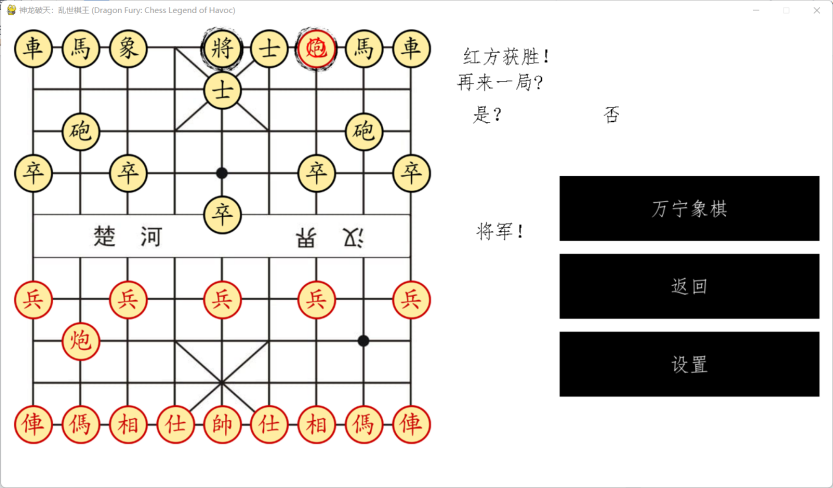
第三方库下载地址：<https://www.pygame.org/>

（不是第三方库但是需要下载的标准库）Copy：copy 是一个用于复制对象的标准库。它提供了两种不同的复制方式：浅复制和深复制。浅复制（Shallow Copy）是创建一个新对象，但是该对象的内容只是原始对象的引用。也就是说，新对象中的引用指向原始对象中的数据。当原始对象发生变化时，新对象也会受到影响。浅复制可以使用 copy.copy() 函数来实现。深复制（Deep Copy）是创建一个新对象，并且该对象的内容是原始对象的副本。也就是说，新对象中的数据是原始对象中数据的独立拷贝。当原始对象发生变化时，新对象不会受到影响。深复制可以使用 copy.deepcopy() 函数来实现。

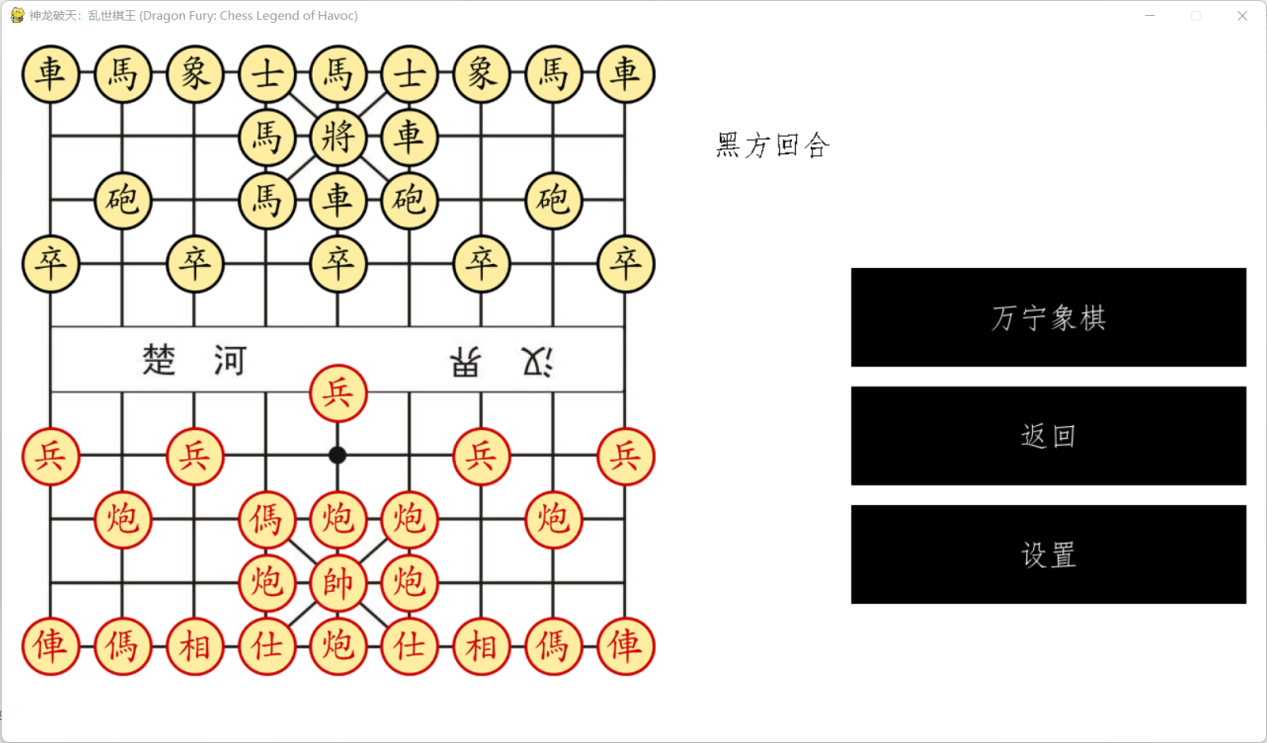
**软件运行截图：**

中英文版界面（封面参考天天象棋的封面并加上自己制作的logo，游戏界面采取简约风格）

选中，落子特效：

被绝杀样式

万宁象棋（按下后将帅旁随机出现车马炮）

**完成度评价：95%**

本项目基本完成了项目立项时期定下的目标，在小组成员的通力合作下，实现了界面切换，各个棋子的功能，棋盘显示，棋子操作以及象棋规则判定。做出了合理的UI，完成了这个面向对象的象棋项目。同时，为了使这个单纯的双人象棋对弈游戏更为有趣，加入了项目目标之外的但是在现今较为火热娱乐项目。

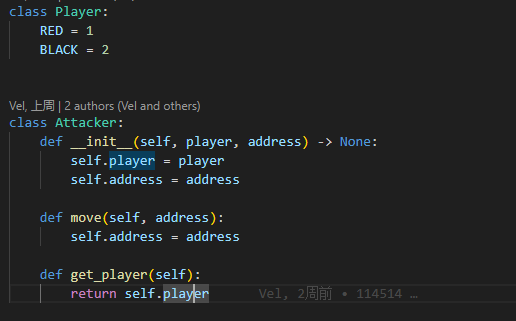
同时，小组通过git更新代码，加强对python使用的信息检索，合作完成这个开源项目，也给了我们很多宝贵的项目开发经验。

不足之处：小组成员在项目开发上的经验不足，处理方式略显稚嫩，例如写代码的时候易漏下self，把对象当成普通的字符串处理，以及将bug不全部修改完全便投入使用导致整个程序需要大幅度的，更费神的修改。

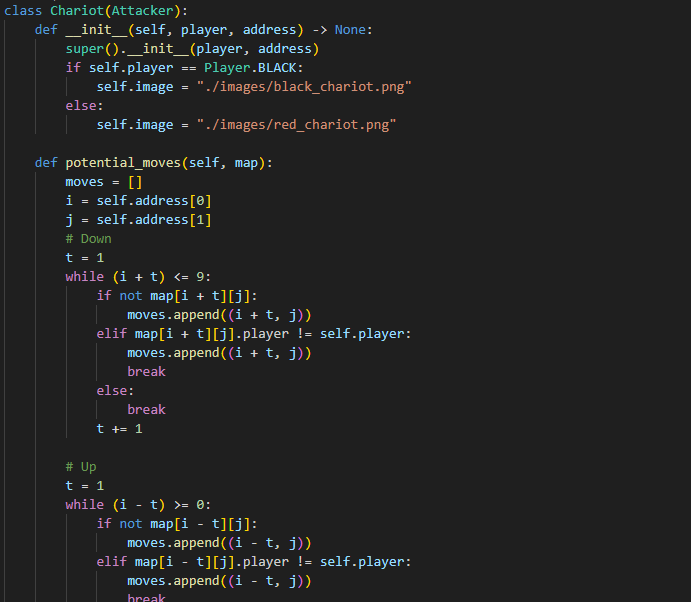
同时，通过调研采访我们了解到象棋游戏一般不会加入背景bgm，但是由于棋子底层代码的编写导致我们不好加入吃子的音效，为该项目的遗憾之一。

**个人工作**

本人工作主要是负责attacker.py这一文件的部分内容编写以及游戏中UI的汉化

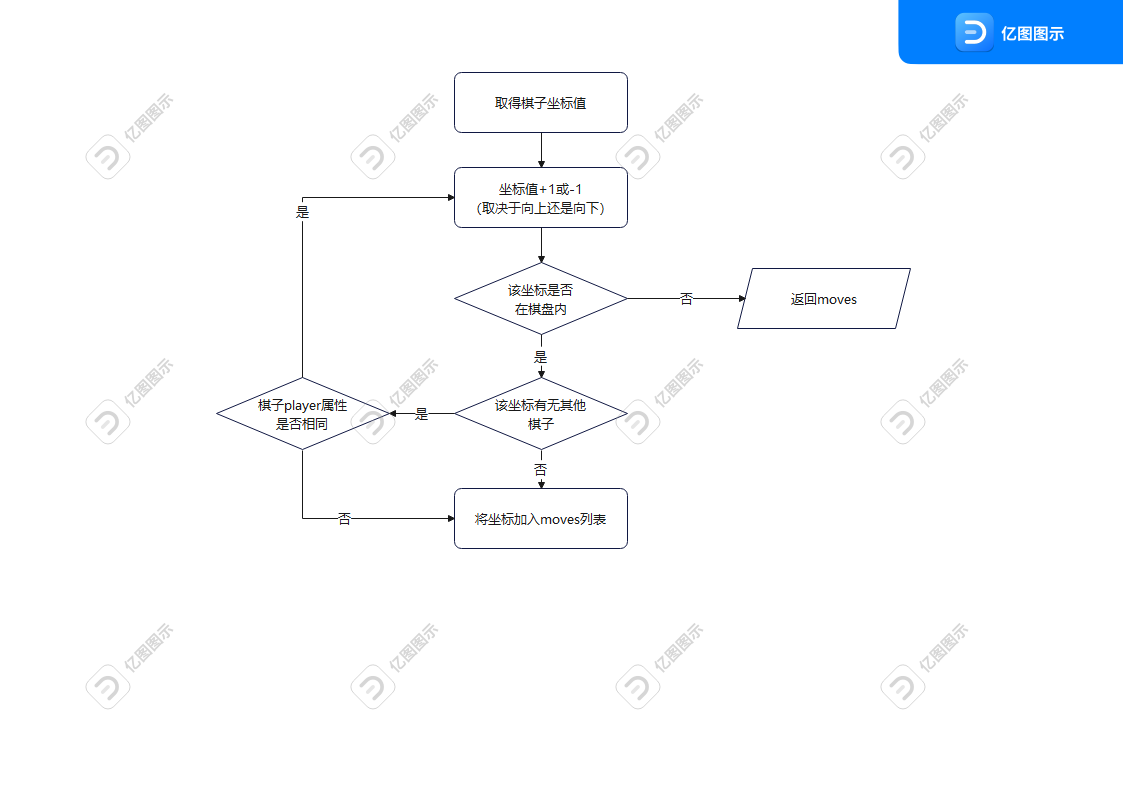


如图，首先定义了Player类以确定是红棋还是黑棋，然后定义attacker类表示棋子，棋子拥有两个方法：移动棋子（更改棋子坐标）、获取棋子的颜色；棋子又有子类，表示不同样式的棋子，车、马、象等等。

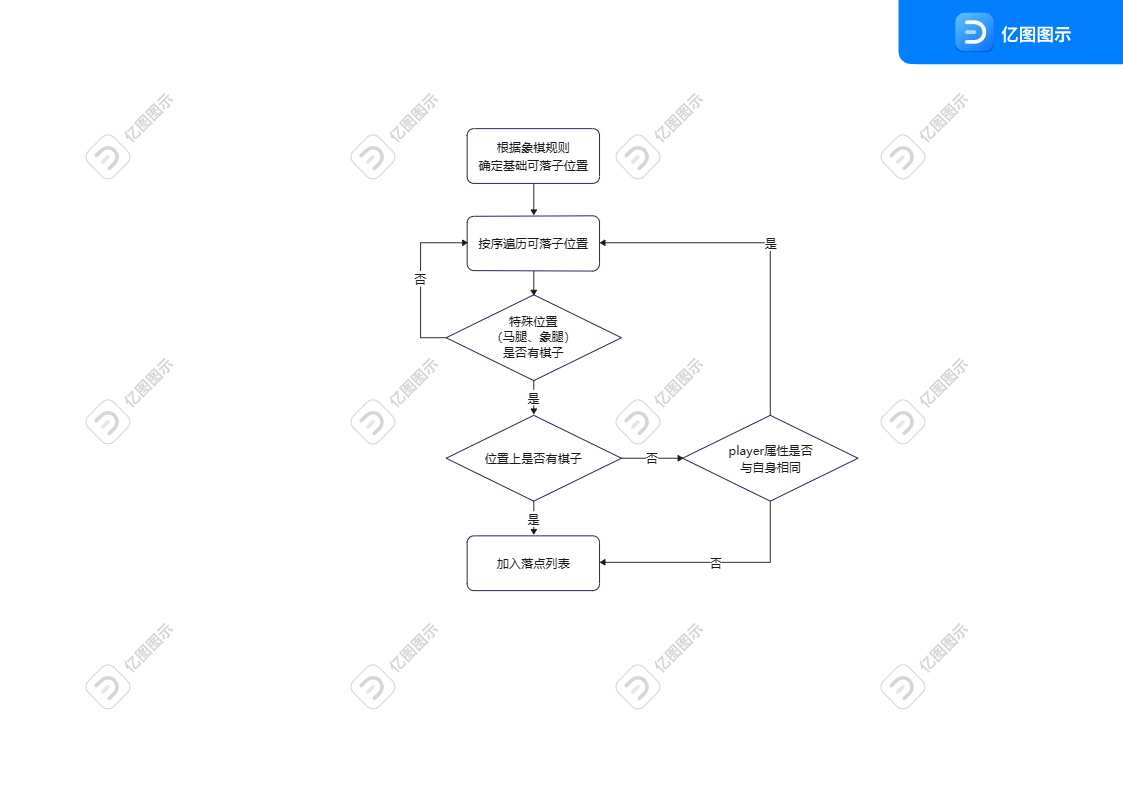


以车为例，子类包含了三个元素：player（表示棋子阵营）、address（表示棋子位置）、image（棋子图片路径）和一个方法potential\_moves（可能的落子点）

可直线行走的一般棋子的potential\_moves方法的流程图如下：



特殊行走方式（马、象）额外的判定方法流程图：



接着针对不同棋子的特性进行额外的编写（小兵过河不回头、炮吃子需要炮架等），最终完成每个类的包装。

在主程序中可直接调用类，通过类的方法获得可以移动的点位。

汉化则是通过创建一个language类，其中属性包含一个名为texts的字典属性和一个current\_language属性；同时包含一个get\_text方法，根据self.current\_language属性通过字典对照返回对应的文字。

