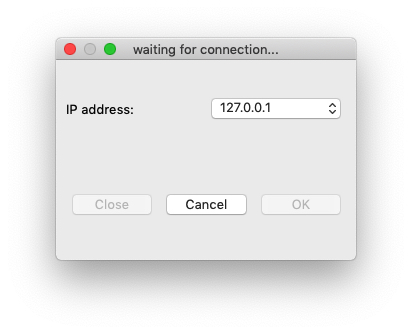
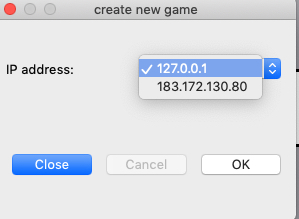
**网络对战国际象棋：说明文档**

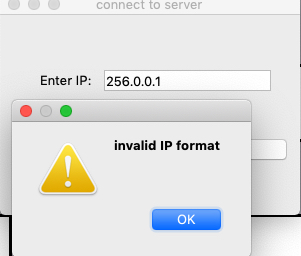
2018011359 计84 乐阳

1. **功能演示**
2. 连接建立

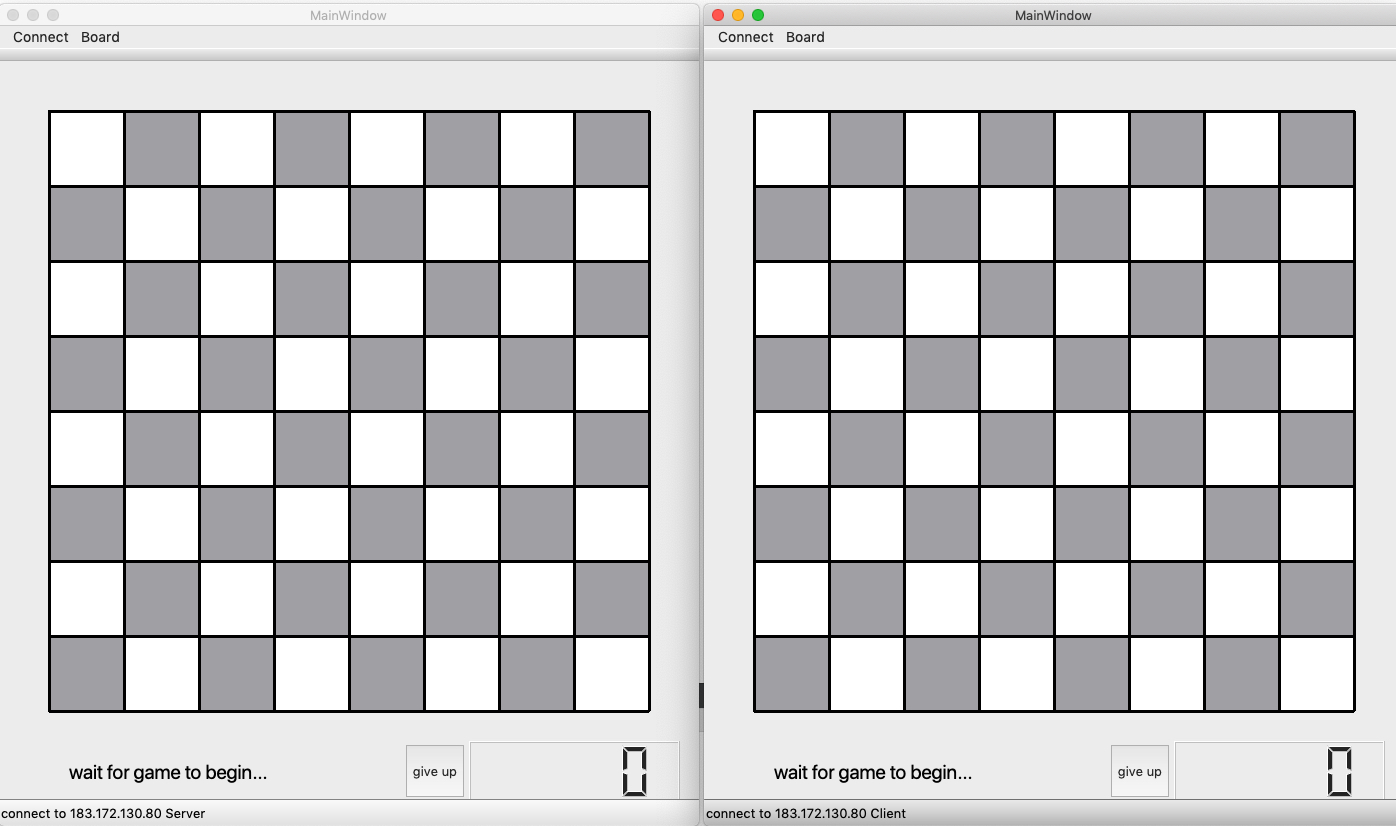
* 打开程序，在建立连接前所有功能不可用。点击Connect菜单下的Server或Client分别以主机和用户的身份连接。
* Server对话框中有一个下拉列表显示当前机器的所有IP地址，可以选择其一开始监听创建游戏。开始监听时标题变为“等待连接”，按Cancel按钮可以取消监听。



* Client对话框中有一个输入框可键入连接的IP地址。点击OK，如果格式错误则报错，格式正确则准备连接，标题变为“连接中”。如连接未成功窗口将一直阻塞，点击Cancel可取消当前连接并关闭该对话框。

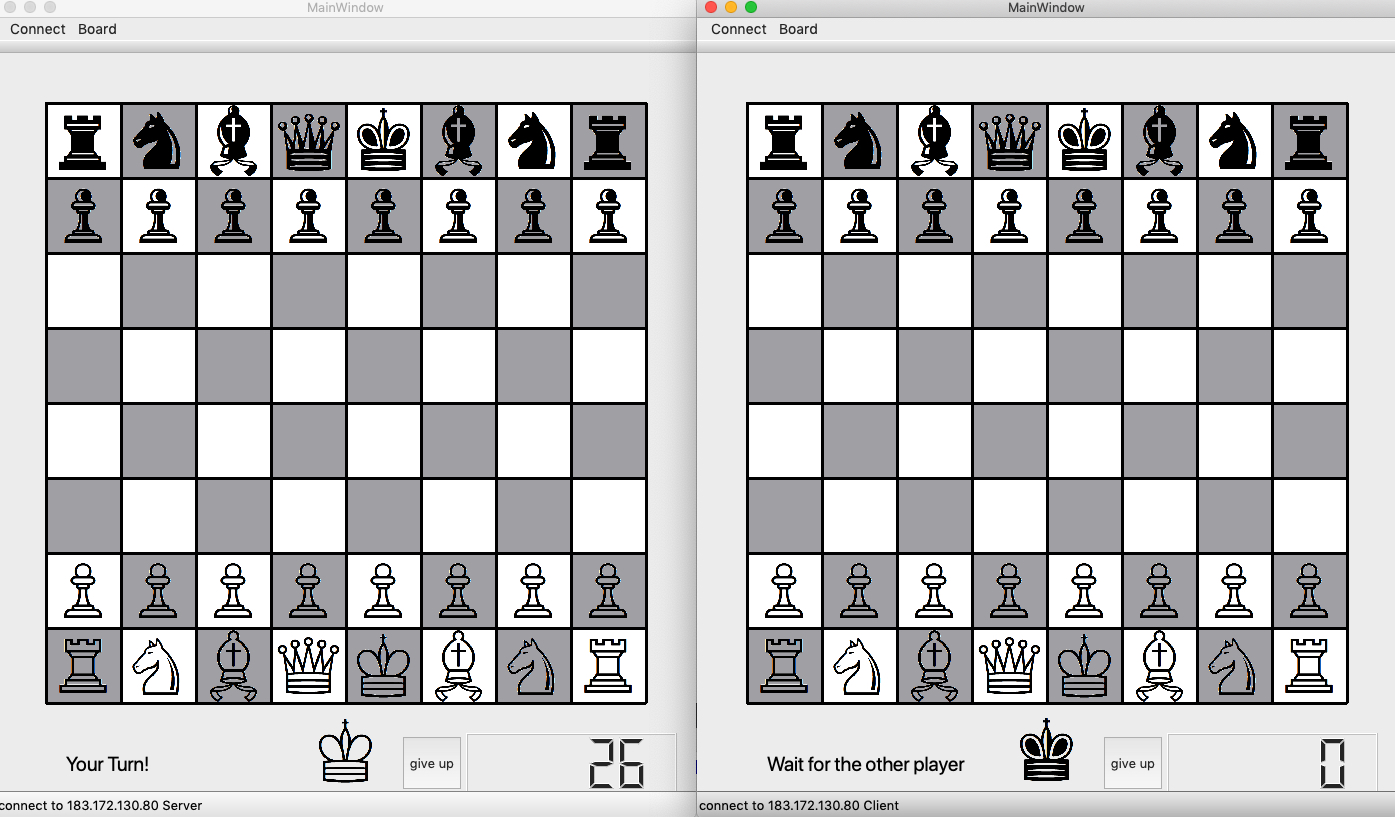


* Server和Client连接成功后，连接的消息会显示在窗口底部状态栏中。此时Server端的Initialize, load file, save file 功能可用，可以开始新对局。（规定只有Server可开始对局）

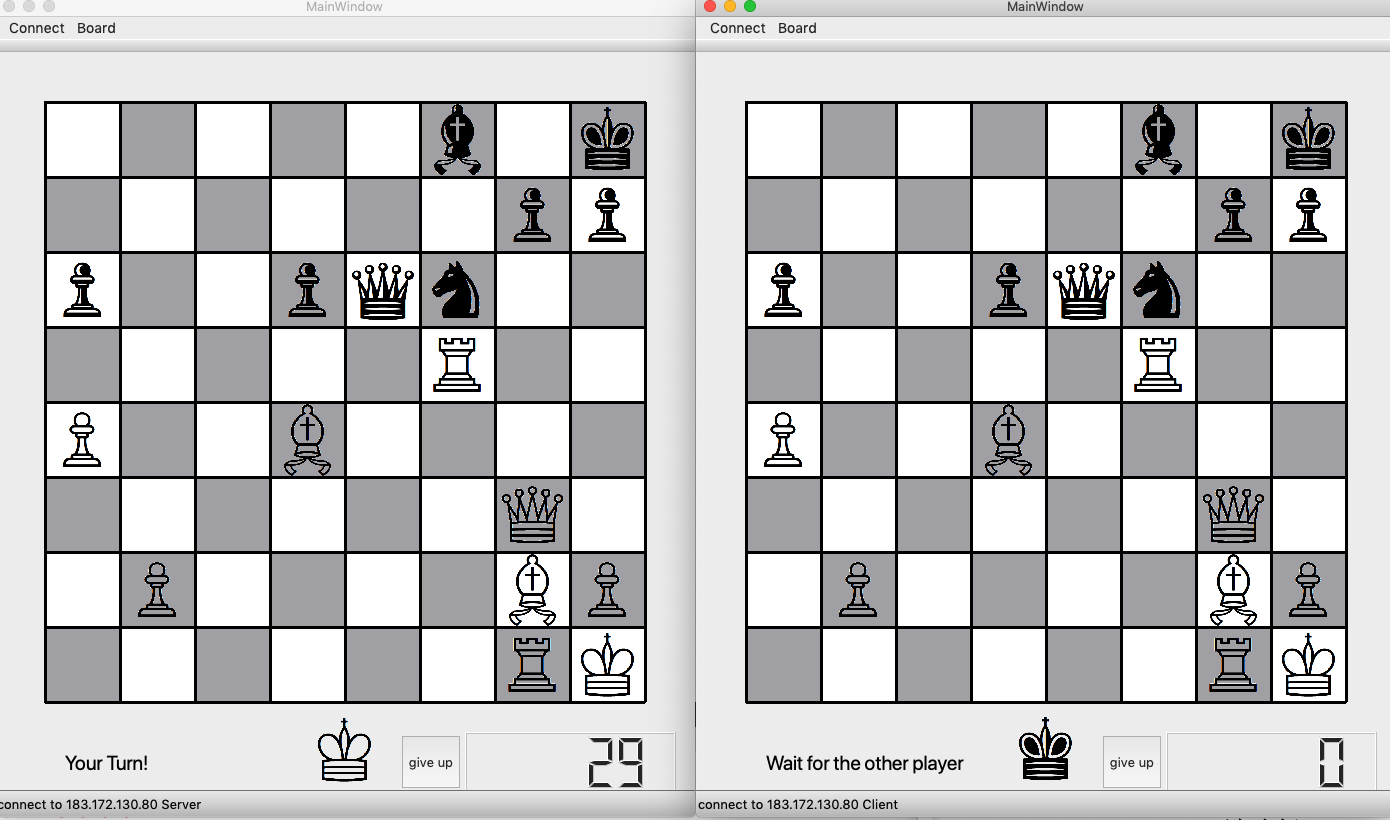


1. 开始对局

* Board菜单下 的Initialize, load file分别代表开始新局以及读入残局开始。Save file弹出对话框保存当前局面到文件。
* Server端选择Initialize后，双方立即开始新对局。默认Server端先手执白。



* Server端选择Load File后，先在Server端弹出对话框选择文件，接着Client端会自动弹出窗口，选择相同的文件后可以开始继续残局。默认残局记载的先手一方为Server。



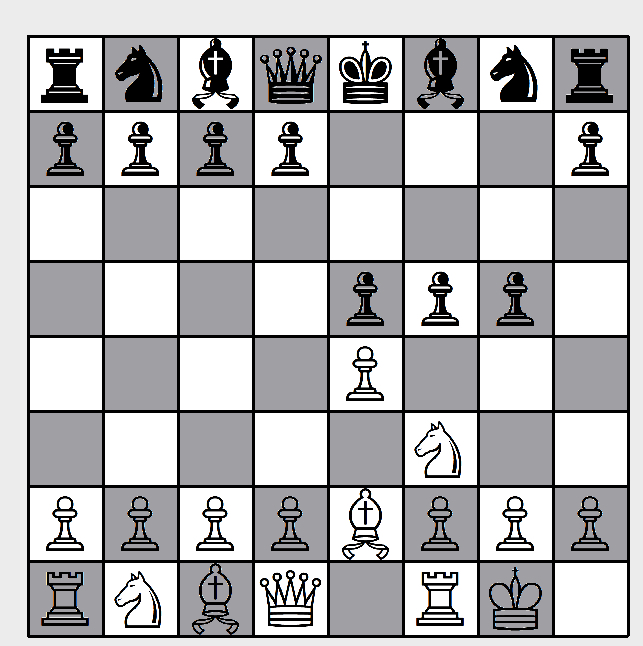
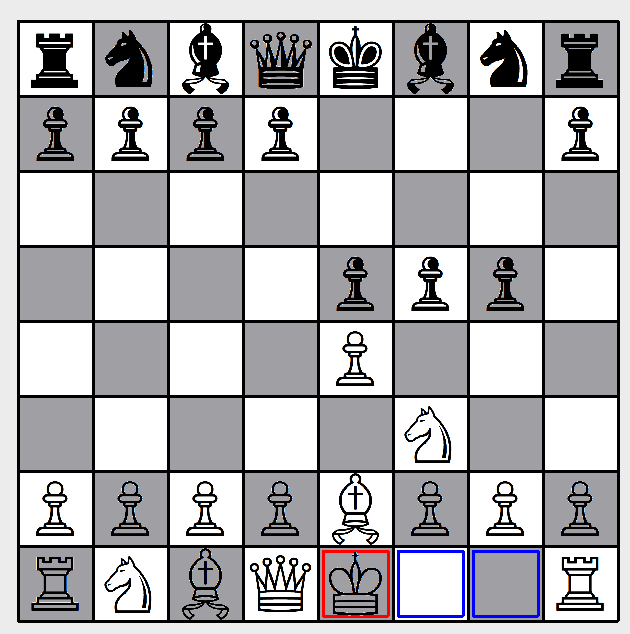
* 当Server端选择Initialize或open file后，认为对局已经开始。对局进行中不能再次选择开局直到双方分出胜负或和棋。上一盘对局结束后，Server端可以重新选择开局模式，不必重新建立连接。事实上本程序只进行一次连接，对局可以连续进行。

1. 对局进行

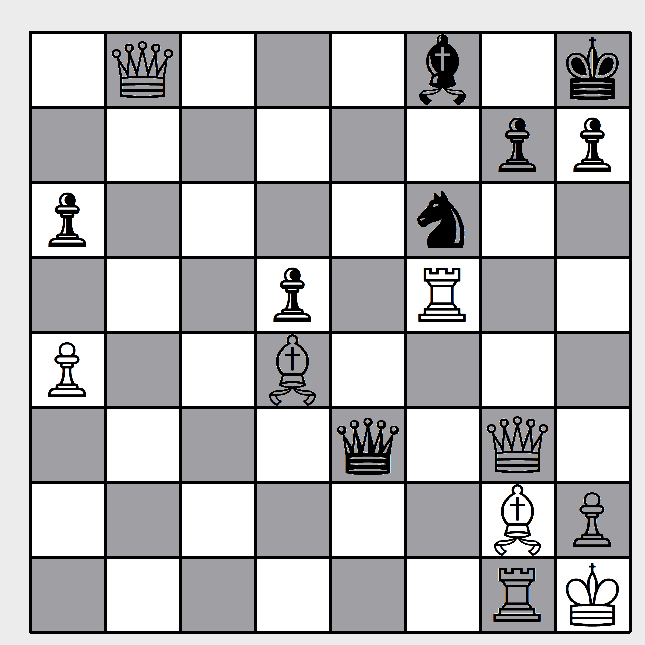
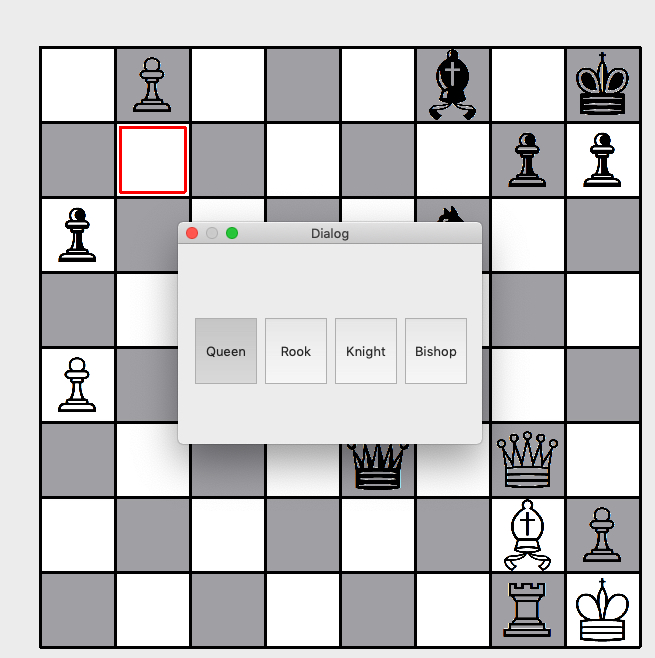
* 开始对局时，棋盘下方的各个控件指示了当前是否为己方回合、己方阵营（用国王图标表示）、认输按钮、当前剩余时间。
* 当对局进行到己方回合时，玩家可以操控己方棋子移动。点击一下为选中，棋子周围显示红色框，该棋子可到达的位置显示为蓝色。



* 王车易位的操作也用蓝色框显示。选中国王棋子后，若王车易位条件满足，则王易位后的位置（左或右边两格）显示为蓝色，提示玩家可以进行王车易位。



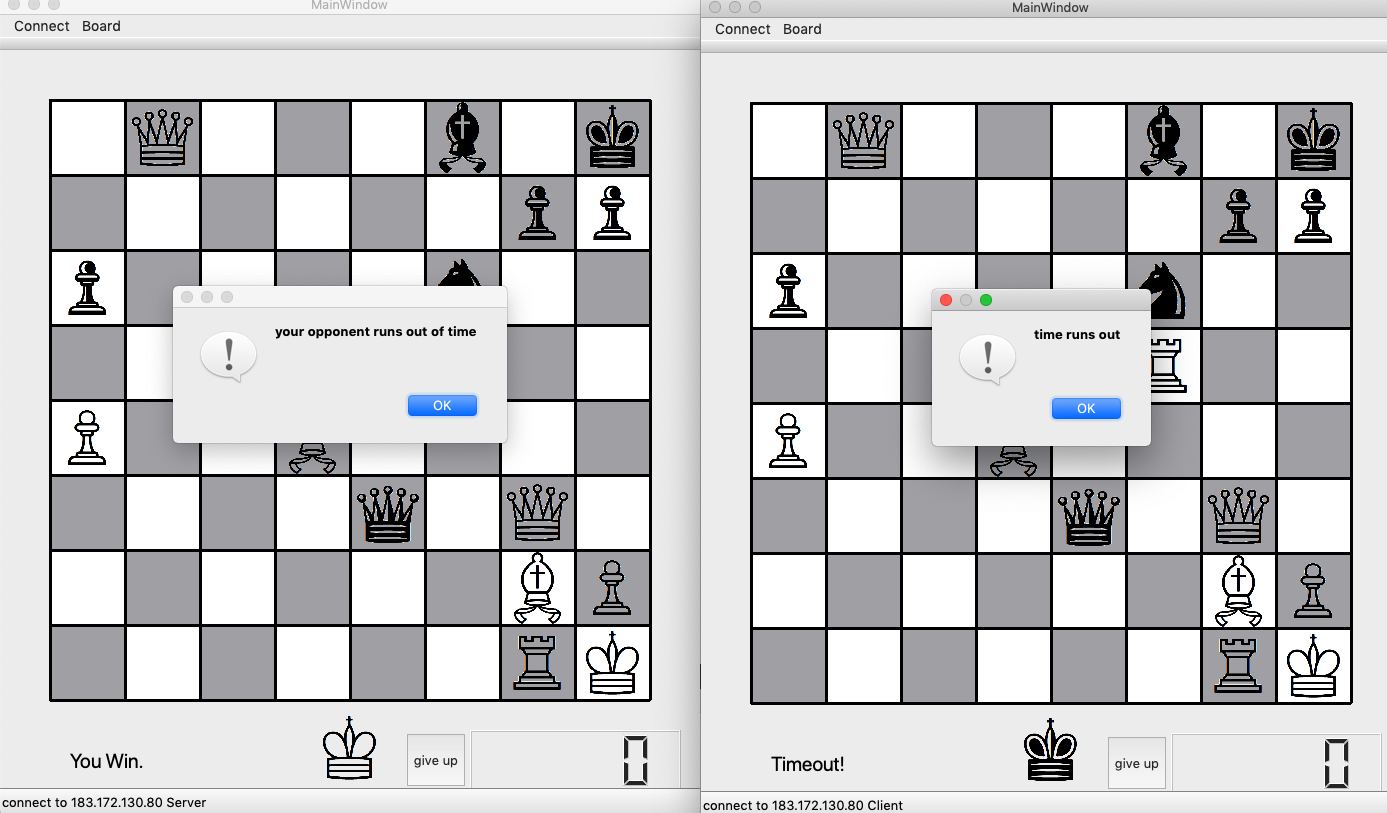
* 兵升变操作通过弹出对话框完成。玩家可以选择升变成后、车、马、象。



* 当一方王被吃掉时（判负），对局双方分别弹窗告知对局结果



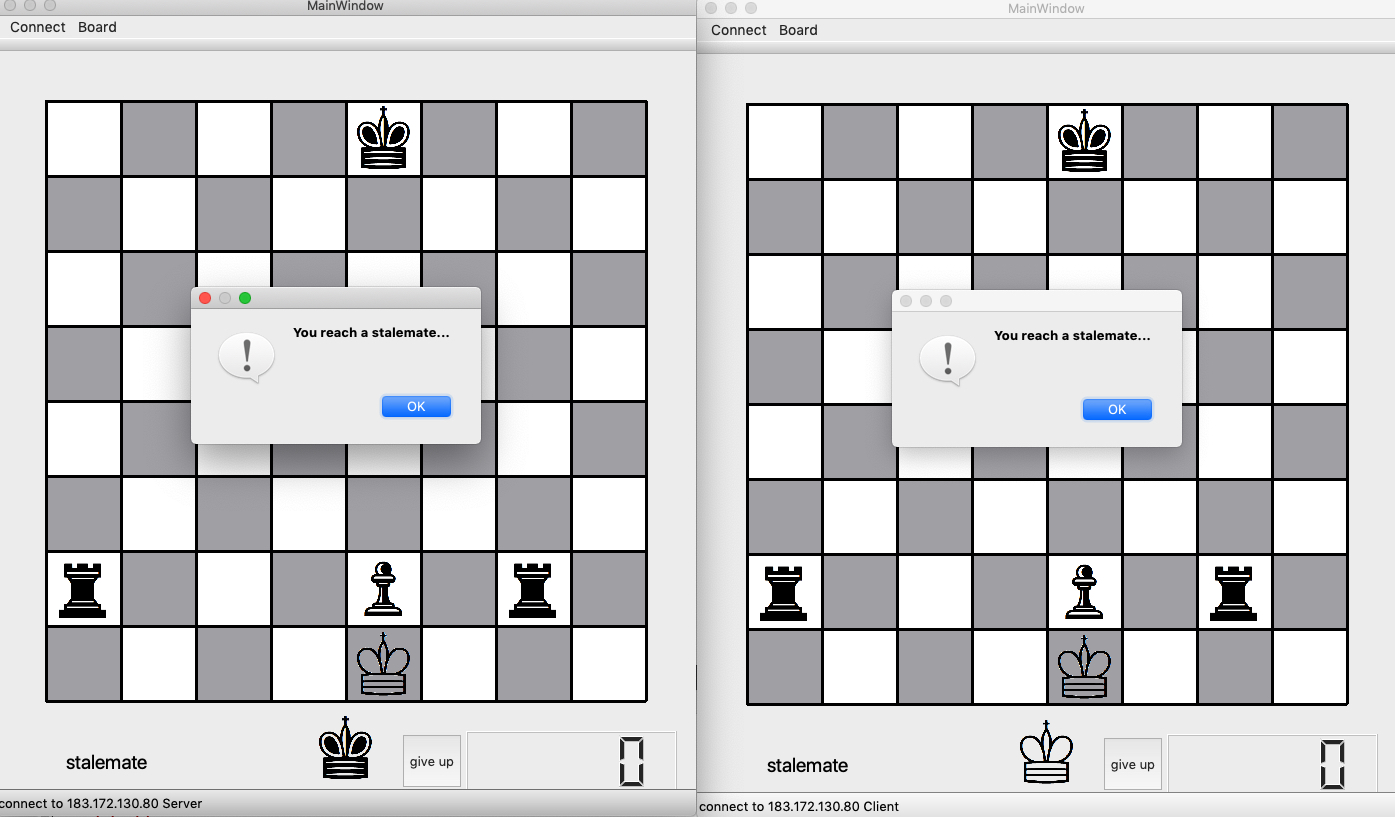
* 当一方30秒倒计时耗尽时（判负），对局双方分别弹窗告知结果。



* 当一方主动认输时（判负），对局双方分别弹窗告知结果。



* 当一方未将军，而另一方在不送吃王的情况下无棋可走时，形成逼和。双方弹窗告知结果。



1. **设计模式**
2. 主体架构

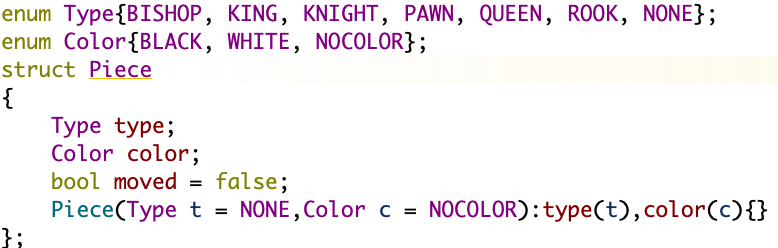
本程序继承QMainWindow开发，其CentralWidget设置为自定义的棋盘类Board。Board负责绘图以及捕捉玩家与棋盘的交互操作。

建立连接基于QTcpServer以及QTcpSocket，分别使用了Server和Client两个对话框。在对话框类内完成连接的监听和建立，在连接成功后将准备好读写的socket返回给MainWindow类。在对局进行中，由MainWindow类负责收发数据并解析。

象棋规则的核心为Chess类，其中实现了象棋的全部规则，包含当前对局的各种信息。整个程序使用一个全局Chess对象c来管理，其他所有类都可以直接访问它本身以及它的成员，十分方便。

1. 局面的表示与计算

用结构体Piece表示一个棋子或一个棋盘中的格子。空格子用NONE和NOCOLOR表示。在Chess类内有一个8\*8的Piece数组grid，完整记录本地局面情况。



Chess::cal\_avail是象棋规则的核心，它根据基本的移动规则计算某位置的棋子从当前位置可以到达的区域，存储在available数组中。

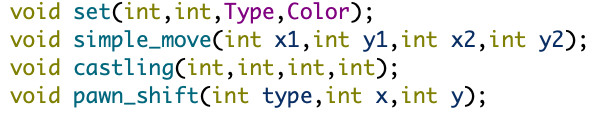


王车易位检查实现在check\_castling中，需要调用generate\_attack\_area计算出当前某方所有棋子攻击范围的并集。进而判断出王车易位路线是否符合规则。

胜负检查check通过简单寻找某方王是否还存活实现。

逼和检查check\_stalemate遍历己方所有棋子的所有可能走到的位置，在后台模拟走该位置后王是否有危险（如有危险视为不可走）。若所有棋子都无可走的地方，同时对方不在将军状态，则返回逼和结果。

普通走棋调用simple\_move，王车易位调用castling，兵升变调用pawn\_shift。



本方走棋时的判定工作交给choose\_from和choose\_to两个函数完成。

1. 通信协议

本程序通过五个整数(type, x1, y1, x2, y2)组成的packet传输数据。第一个整数表示包种类，剩下四个数承载包的信息。具体含义如下表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| type | 含义 | x1 | y1 | x2 | y2 |
| 0 | 开始对局 | 0代表新局  1 代表读文件 | 0 | 0 | 0 |
| 1 | 普通移动 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 2 | 王车易位 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 3 | 兵升变为后 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 4 | 兵升变为车 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 5 | 兵升变为马 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 6 | 兵升变为象 | 起点横坐标 | 起点纵坐标 | 终点横坐标 | 终点纵坐标 |
| 11 | 对方超时 | 0 | 0 | 0 | 0 |
| 12 | 对方认输 | 0 | 0 | 0 | 0 |

对局过程中，棋面走子规则判定、胜负判定、逼和判定都在本地进行。只传输棋子的移动。