**面向对象程序设计（C++）**

**课程设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 五子棋游戏 | | |
| 日 期 | 2019年7月20日 至 2019年8月18日 | | |
| 专 业 | 计算机科学与技术 | 班 级 | 1803 |
| 姓 名 | 胥卜凡 | 学 号 | 20184484 |

报告成绩

|  |
| --- |
|  |

**报告**撰写的内容与要求

1. 项目简介：介绍本次课程设计选题的目的、意义、任务概况、本人在项目中的分工等内容。
2. 项目内容：系统的设计与实现的全面描述，介绍系统整体结构、系统框架图、UML类图、程序流程图、重点难点分析及解决方案、调试难点及解决方法、系统交互界面及结果展示等。本部分内容应以记叙或[白描手法](http://www.so.com/s?q=%E7%99%BD%E6%8F%8F%E6%89%8B%E6%B3%95&ie=utf-8&src=wenda_link" \t "_blank)为基调，在完整叙述的基础上，对自己认为有重要意义或需要研究解决的问题进行重点叙述，其它内容则可简述。
3. 总结或体会：对课设效果进行综合评价，着重介绍自身的收获与体会，内容较多时可列出[小标题](http://www.so.com/s?q=%E5%B0%8F%E6%A0%87%E9%A2%98&ie=utf-8&src=wenda_link" \t "_blank)，逐一列举。总结或体会的最后部分，应针对实习中发现的自身不足，简要地提出今后学习，努力的方向。
4. 报告正文一律采用计算机排版、A4纸**双面**打印，正文字体为**小四号**宋体，**1.35倍**行距，正文页数不低于10页，不超过30页（其中代码总计不超过3页，**中文字数**不少于4000字）。要求语句通顺、论述严谨、规范、正确。
5. 请注意封面页、扉页、评语页等的打印及装订顺序。分别为①封面页（其背面为扉页“实习报告的内容与要求”）、②目录、③正文、④其他附件（如有）。

**目录**

**（一）项目简介……………………………………………………1**

1.目的……………………………………………………1

2.意义……………………………………………………1

3.任务概况………………………………………………1

4.本人在项目中的分工…………………………………1

**(二）项目内容……………………………………………………1**

1.系统整体结构…………………………………………1

2.系统框架图……………………………………………2

3.UML类图………………………………………………2

4.程序流程图……………………………………………4

5.重点难点分析及解决方案……………………………4

6.调试难点及解决方法…………………………………6

7.系统交互与结果展示…………………………………7

**（三）总结与体会…………………………………………………12**

1. 项目简介：
2. 目的：

加深自己对c++的理解，做到知行合一；熟练掌握图形界面的编程方法，熟练掌握QT的使用方法，了解信号与槽机制，为未来通信项目做好十足的准备。

1. 意义：

五子棋是一个十分有趣、老少咸宜的益智类游戏，通过对一个五子棋游戏的设计与实现，既可以加强我的逻辑思维能力，又可以加强自己的c++编程能力，除了上述基本能力的提升外，由于本次课程设计需要使用图形界面的绘制与管理，成功的完成本次任务可以极大的扩展我的编程范围，使原本的只能在控制台输出改为可视化的图形界面，这在我看来是能使我收获很多的。

1. 任务概况：

绘制欢迎界面、说明界面和游戏界面，显示游戏菜单，通过菜单的分支进行界面跳转，绘制游戏说明，使新手能较好的理解游戏界面的各个功能点，实现悔棋功能，并加入选择框减少客户的误点失误；实现复盘功能，且能非常清晰地显示出双方的步骤数目；对于游戏界面，我们用QT自带的绘制功能进行绘制，并通过对其各个事件虚函数的改写执行鼠标按压完成落子操作，鼠标移动执行选择边框的绘制功能；通过定时器的使用，我们实现了游戏的读秒功能；通过对图像的调用，我们实现了点击头像更改玩家头像的设定；对于帮助界面，我们通过PS和截图进行了说明功能，比较清晰明了，并加入了换页功能。

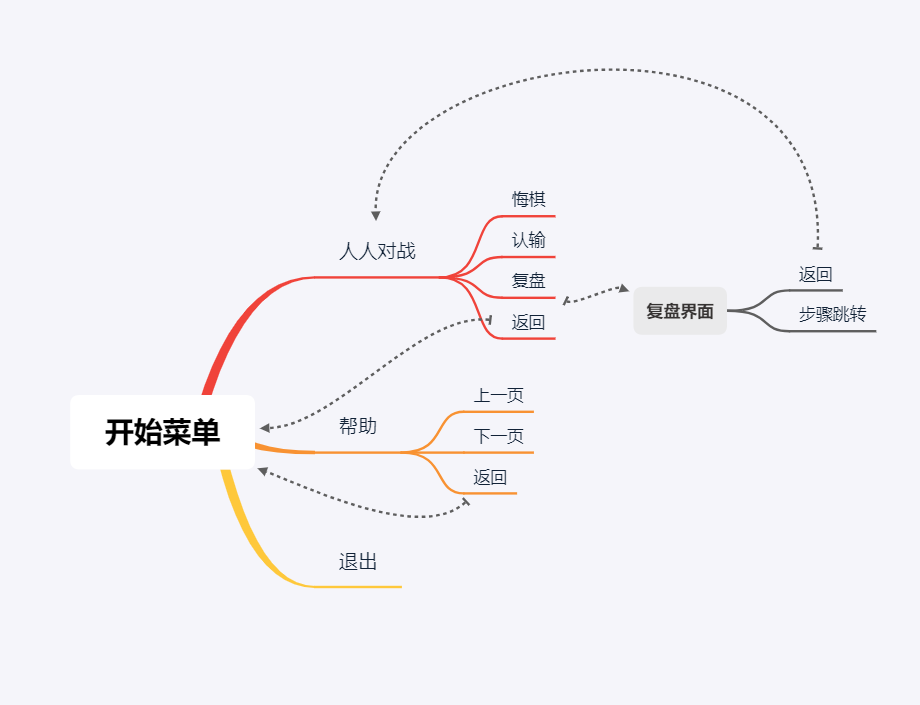
1. 本人在项目中的分工：

我在本次项目之中担任小组组长的角色，主要是进行各个项目的整合，对悔棋功能、复盘功能，人机功能、认输功能的实现，同时在其他项目中也提供了思路，在其他成员遇到困难时提供帮助，监督辅助实现每一个功能的设计。

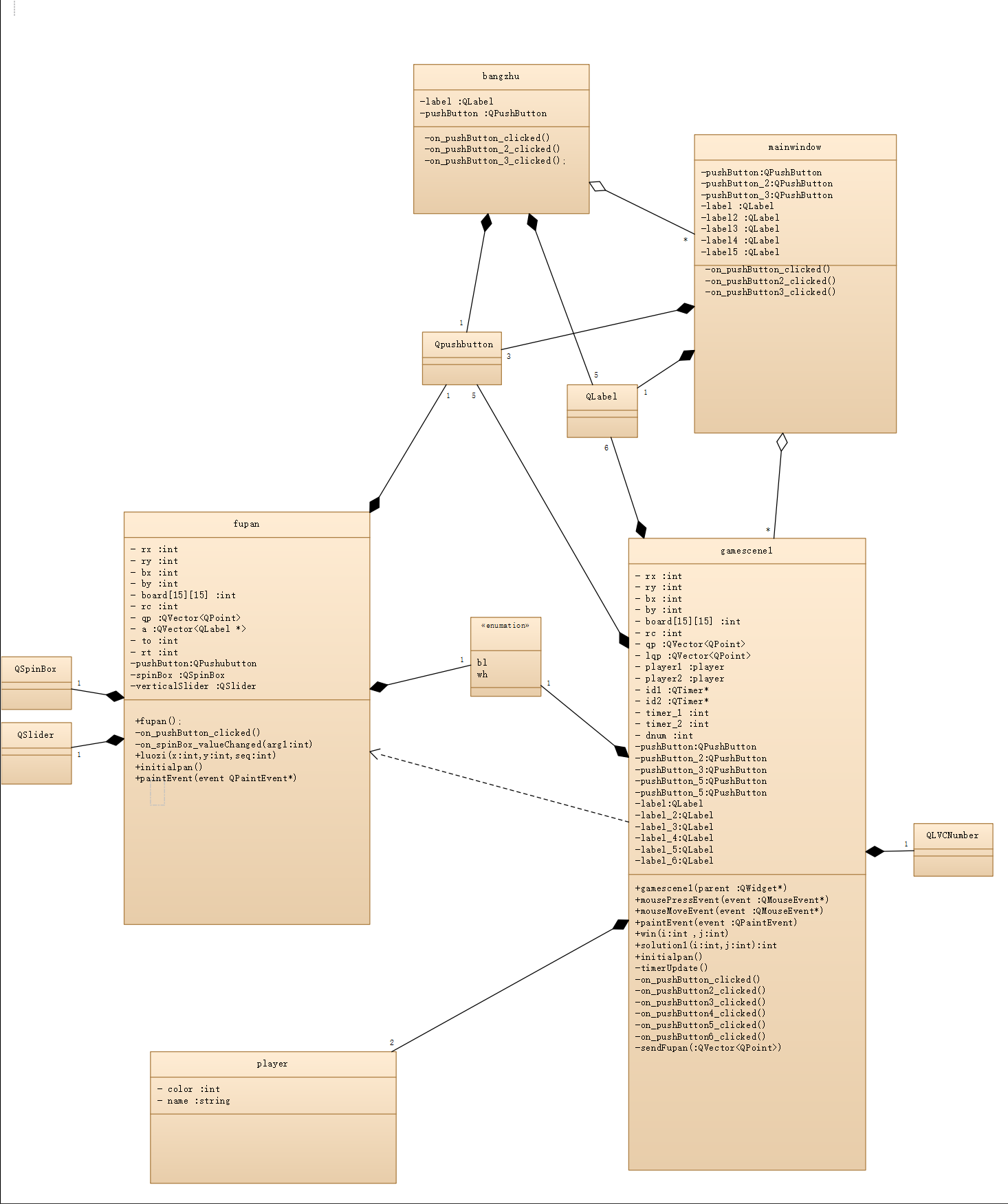
1. 项目内容：
2. 系统整体结构：

主要分为主界面，游戏界面，帮助界面；其中主界面包含三个菜单项:前两个引向游戏界面，帮助界面，最后一个实现退出游戏的功能；游戏界面是一个2000x1500的界面，其中1500x1500实现游戏主界面，即完成棋盘和棋子的绘制功能，另外500x1500的窗口空间实现了玩家1和玩家2的头像绘制，定时器绘制，并包含三个菜单项:悔棋功能，复盘功能，返回；其中每个功能都需要选择框来确认操作，复盘功能进入了一个新的复盘界面，由游戏界面发出的信号来传递棋谱，并提供了返回功能。帮助界面通过“上一页”、“下一页”的选择来进行帮助图片的转换，也提供了返回功能返回上一个界面。

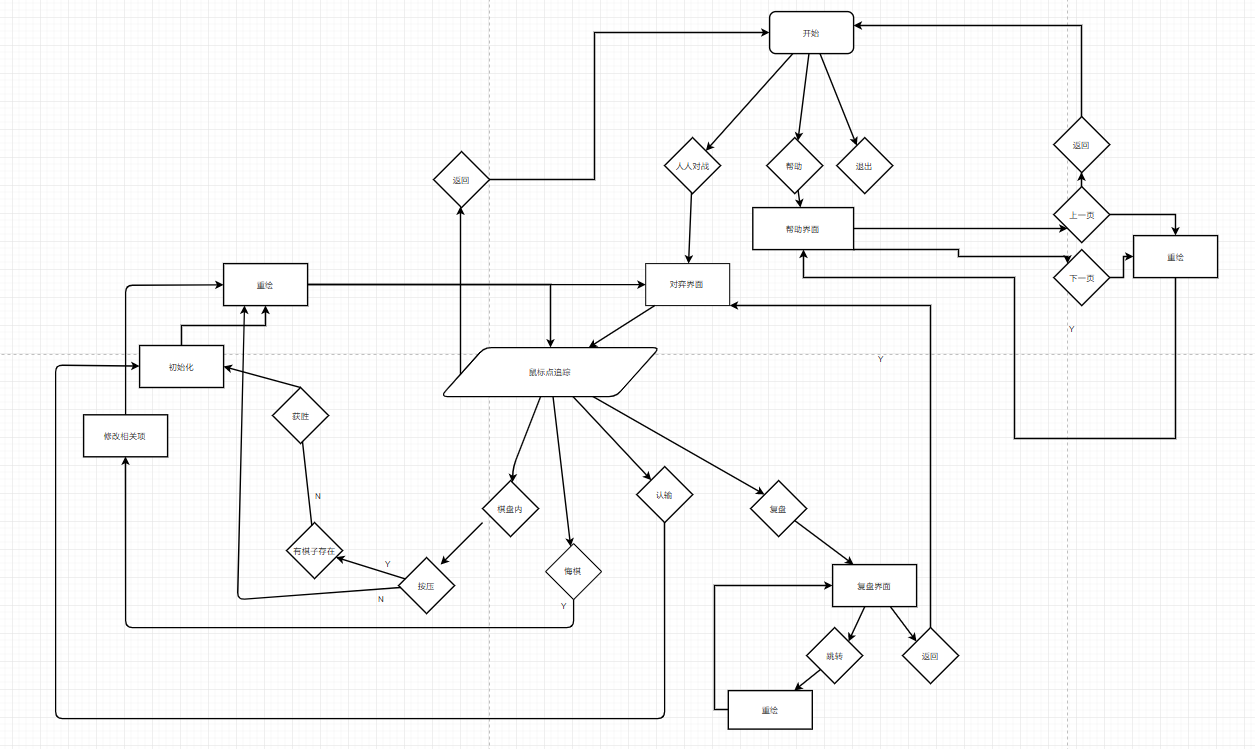
1. 系统框架图：



1. UML类图：



1. 程序流程图



1. 重点难点分析及解决方案
2. 悔棋功能的实现

悔棋，百度上的解释是后悔走了一步棋，想要重新走一步。

从字面上分析，是对上一步感到不满意，进而向对方申请，重新进行一次落子子操作。

整个操作过程需要进行回溯操作，由于回溯又需要保存数据，我们据此就可以开始进行解决了。

对此，为了可以进行上一次操作的回溯，我有想了几种方案，一种是每步进行前都先进行一次操作保存，如果有接到悔棋按钮的点击，就调用操作重置；一种是保存整个动作界面，如果执行悔棋功能，就进行回调即可；一种是用vector容器或者array容器来保存棋谱，每进行一次落子操作就保存一次落子的地点，如果要悔棋，就进行弹出操作，再向系统申请重新绘制整个图谱。

最终我选择了用容器来进行保存，由于系统自带的容器的特殊性，既减少了系统内存的压力，又能对后面复盘功能的实现提供方法和思路，最好地满足了所需。

1. 复盘功能的实现：

复盘，即按原先的走法，把下完的棋再摆一遍。

从字面上我们可以发现复盘也需要进行原先的回溯，因此我们可以利用

从悔棋功能中实现的方法来拓展加深得到复盘功能实现的方法

此外，如果我们除了再摆一遍外还应该提供一些额外的功能，如显示每一步的落子，能够根据想要到达的地方进行跳转，以此来分析归纳，增加自己的棋力从而实现复盘的效果。

复盘需要每一步棋谱的保存，对此已在悔棋功能中得到了所需的答案，即用容器来保存棋谱，每进行一次落子就进行一次推入容器的操作，但由于复盘是要对上一局的对局进行复盘，所以我们选择在主窗口中新增一个容器专门用来保存上一局棋盘的操作，具体来说要通过在胜负检验中加入专门的代码来进行保存本局棋谱和清空本局棋谱的操作，在需要执行复盘功能的时候向复盘界面发送锁保存的棋谱，这里注意要在游戏界面初始化时设好判断的初值，否则很容易出现极其匪夷所思的bug，如；

由于我认为复盘功能需要一个新的界面来实现，所以还需要新窗口和原来窗口进行信息通讯，为此我们使用QT的信号与槽机制来进行棋谱的发送，对于界面绘制我在图像区域对其他成员的代码进行复制实现了复盘与实际棋盘落子时画面的统一，但不仅限于此，为了能够清晰的显示每次落子的步骤顺序，我们采用了许多标签类的成员在初始化姐买你时与各个棋盘的落子点进行绑定，在收到父窗口发来的信号即传递过来上一次对局的棋谱信息后，进行对该类棋谱成员的赋值，再通过重新绘制的功能，对各个棋子落点进行选择性画图，对棋子落点的标签进行选择性的修改文本操作；

为了完成步骤跳转，我为界面增添了滑轮和选择盒子并加以绑定，实现数据的实时互通，并将其中的spinbox与类中的成员绑定来进行界面的绘制，由于本类的特殊性，我们同时实现了通过键盘的上下键来进行步骤的跳转，使行为更加方便、快捷了。

3.认输功能的实现

认输功能其实本身并不难想出如何实现，在我看来，这主要需要两方面：一方面需要确认当前认输的玩家名称，这一步我们可以通过对颜色本身来进行绑定操作，但于此带来的麻烦就是我们在每一次落子或一些相应操作后都一定要保证实现了颜色转换，不然很容易发生一些令人匪夷所思的问题。另一方面，我们需要在每次操作后进行初始化的操作，这一部分可以参考之前的代码。

6.调试难点及解决方法

1. 悔棋功能的调试

对于最终方案即使用容器保存棋谱来执行的调试并不复杂，只要在每次绘制之前进行追踪即可，当然这里出现过程序崩溃的问题，经过追踪出错点我发现是没有设置棋谱的初始值导致的，这使程序在未落子之前执行悔棋功能会发生bug。

1. 复盘功能的调式

由于牵扯到了两个窗口的互动，调试显得十分复杂，所以我先选择对每个窗口进行单独的调试，在保证两者都没有出现异常的情况下后再通过信号和槽来进行两个窗口之间的交互。

1. 图片设计相关的调试

由于QT图形界面设计的一些特殊性，我在一开始导入图片后发生了许多意想不到的事故，如在我预先加载所有的label后会发生程序崩溃的问题，经过不断设置调试点，我发现是因为图片大小过大，导致资源导入界面时内存溢出了，对此我一开始选择对图片本身用格式工厂进行压缩处理，从而减少内存量，但之后发现实际效果并不好，因为这样界面本身会变得十分模糊，基本不能完成界面美观的实现，对此经过我上网搜索解决方案后发现只要在其pro工程中添加一行代码即可以接触QTCreator这一编译器本身对内存的限制，当然这样可能会导致一些新的意想不到的bug，但经过不断反复的断点设置和调试，我们最终成功的修复了图片导入失败的bug。

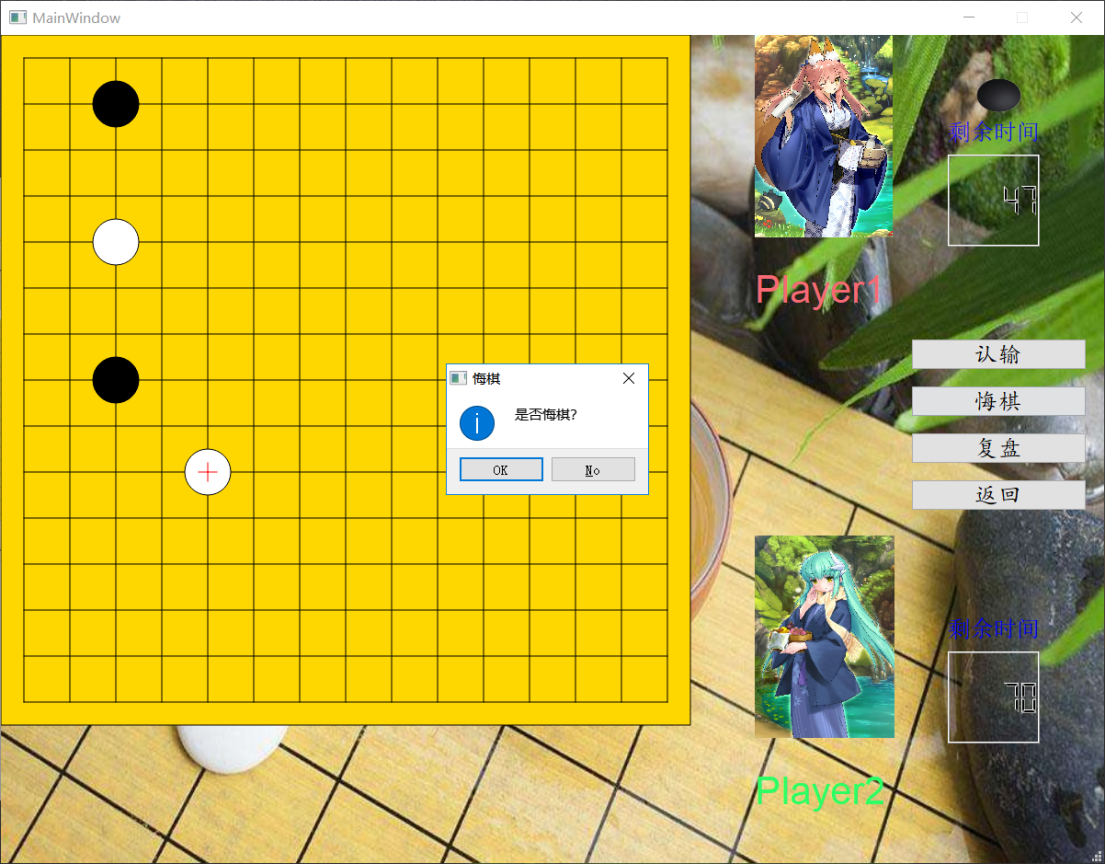
此外，对于Label标签本身的操作有时经常发生问题，这很难调试，原因是我们所使用的标签类是由QT本身定义的，当我们深入其本身的各个函数时很难找到入手点，对此我们不进行仔细的逐行查询错误，因为这工作量太大，我们选择预先想出比较多的加载图片的方法，在经过不断尝试后选择能够成功显示的方法并进行简洁性的筛选。

1. 认输功能的调试

主要来说首先我需要检测之前所说过的颜色、玩家是否一一对应的问题，其次需要检测每次认输后，相应的初始化功能是否得到了解决。这写调试虽然繁琐，但在我看来是非常有必要的，如果不能保证这些基本内容的稳定性，程序的BUG就会层出不穷

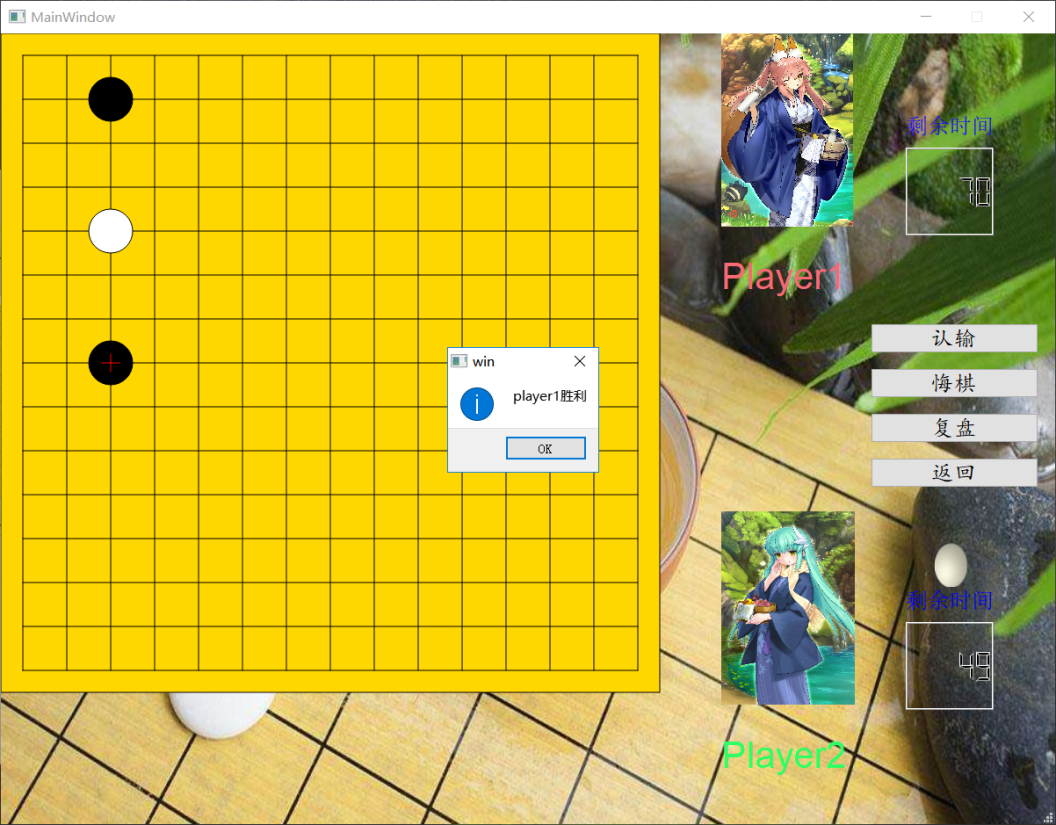
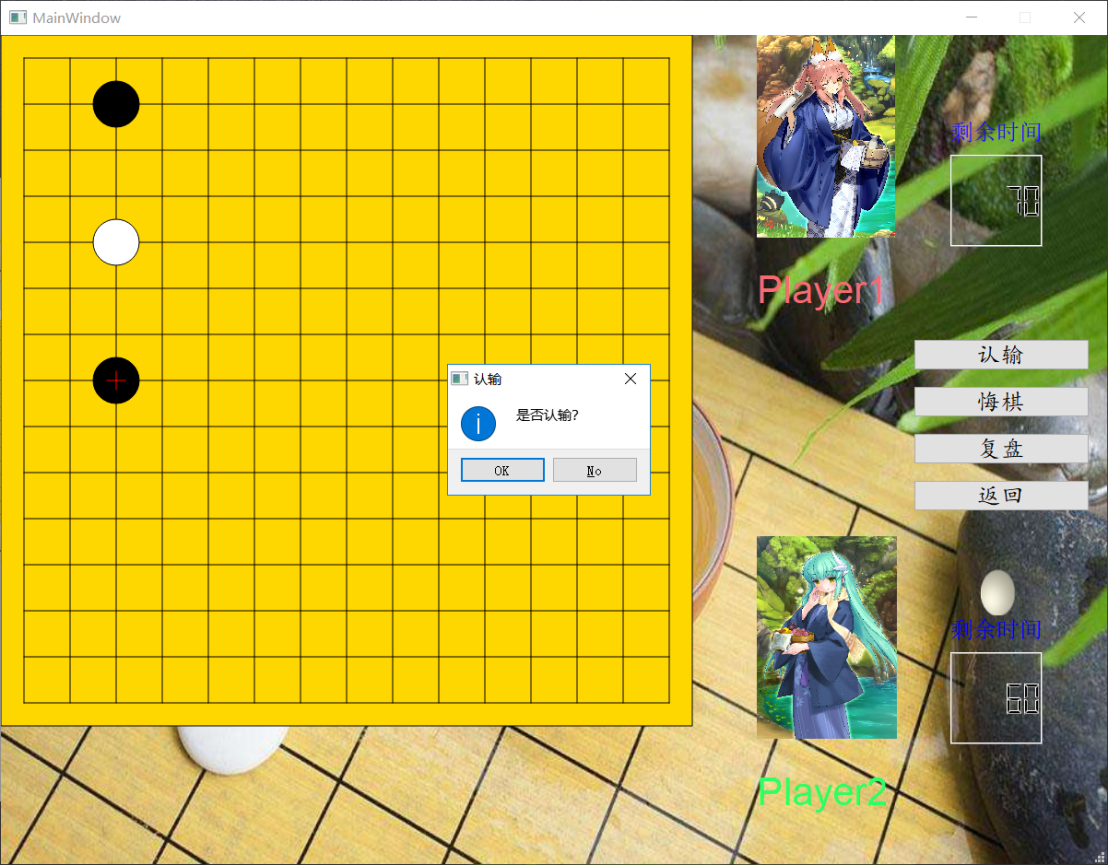
7.系统交互界面及结果展示

悔棋功能：

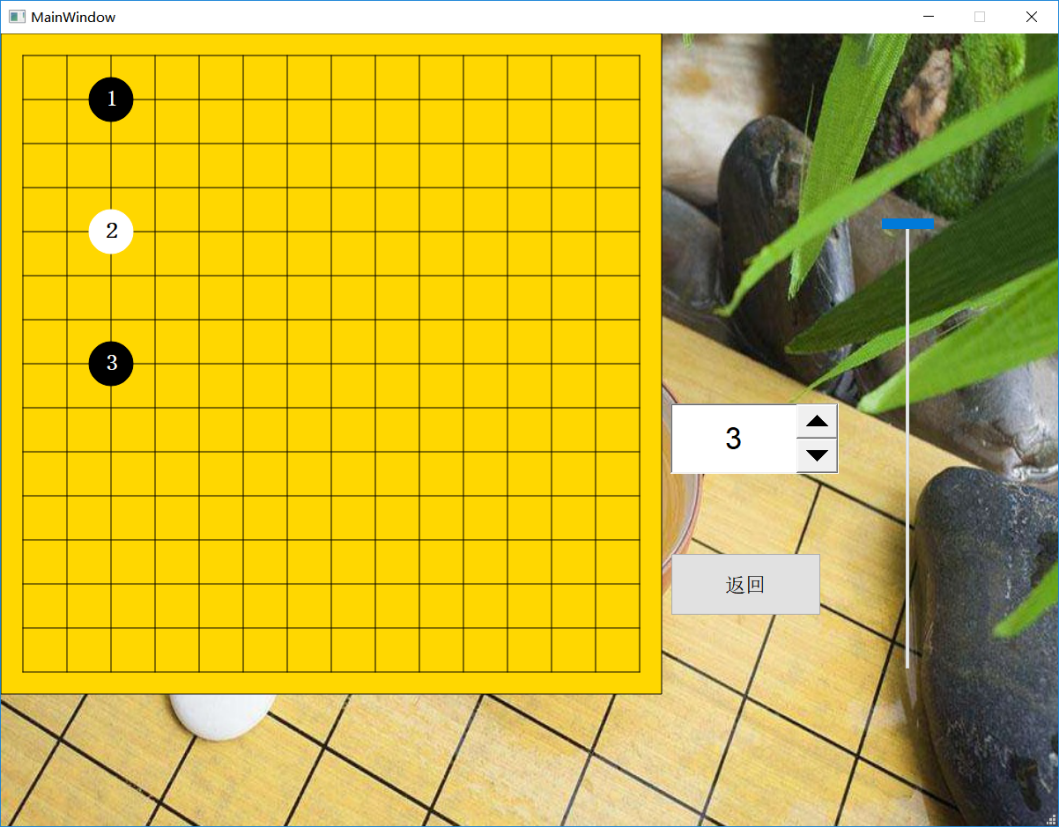


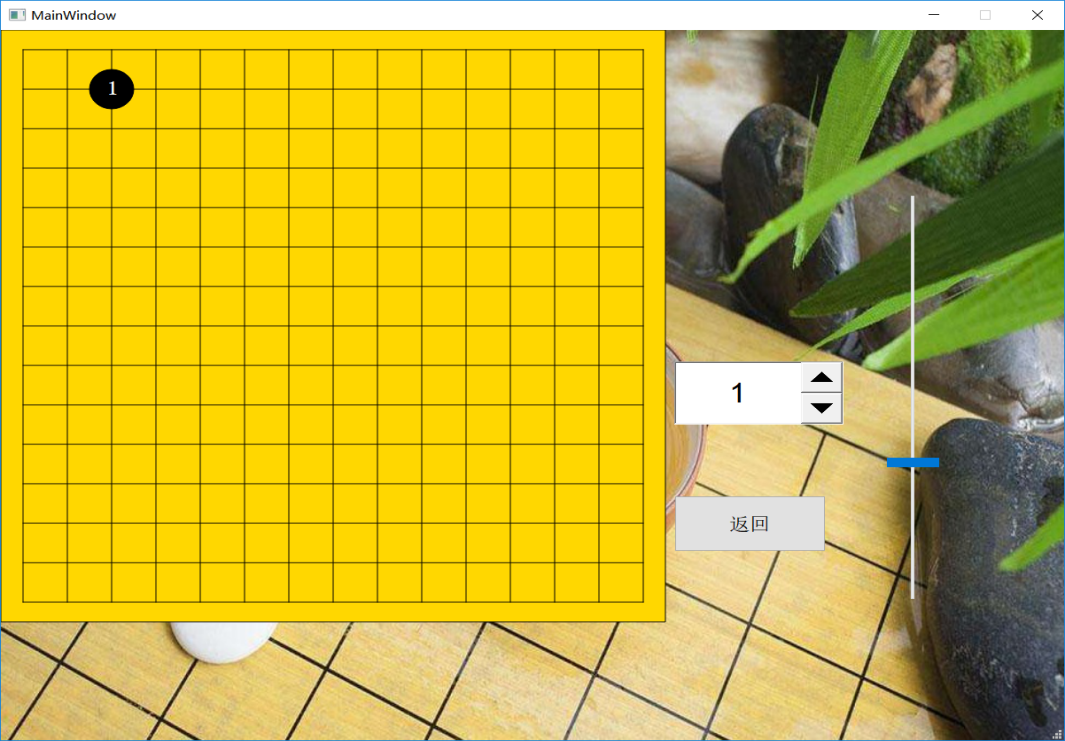


认输功能：



复盘功能：





1. 总结或体会：

首先感谢各位老师，助教及各个同学对我们的帮助，感谢小组成员每个人的付出，这是我们完成项目的基础。

在激烈的讨论和长时间的代码编写与修改下，我认为我们所设计的程序很好的达到了本次课程设计五子棋游戏题目中的要求，功能虽然不够丰富，但也在预定之外增加了许多额外的功能：如对复盘功能设置了步骤跳越，在每次进行操作前都要发放选择框进行选择确认，在游戏界面中我们增加了头像，并能通过文件打开操作加以修改，我们增加了计时器的功能，并能够在计时器结束时进行胜负判断，以及每次落子后进行计时器数目的重置，我们还增加了显示当前操作玩家方和上一次操作方的功能，对于落子的判定，我们增加了鼠标移动的函数，通过适当的函数判定，我们能够在鼠标移动到的棋盘位置显示出红框从而达到界面友好的特点。但同时我们设计的程序还有很多不足之处，如基本没有什么特效，不能实时分析局势，对于棋谱的保留也需要额外的空间支出，没有实现多线程的功能，没有实现人机对战的功能，我们有实现过，但速度实在太慢，且容易发生内存溢出等问题，最终选择删除掉，没有实现联网对决的功能，这些种种都是需要经过优化、添加和修改的，对于这些不足，我在此次课程设计后也会继续这方面知识的深造。

当然，本次课程设计对我个人来讲可谓是收获颇丰了。

首先，我的知识面拓展到了图形界面当中，我通过3天的学习大致掌握了QT的基本用法，知道了如何绘制界面，如何添加图片，如何建造窗口，什么是信号与槽机制——对于多窗口的互动很有效果，如何进行事件的处理……再经过对所需信息的不断查询——这要感谢网络上的各个博主对知识的大力推广和QTcreator自带的帮助面板——我对QT作为图形界面的大部分功能已经有了较为理想的掌握，这是我在上一次课设时就很想做的了，毕竟一直操作的Windows系统就是一个图形界面。

其次，通过对悔棋和复盘的设计。我掌握了vector容器的使用方法，通过对其进行弹出和推进的操作我能够实现悔棋功能，其实队列对这个操作更为友好，更不容易出现bug，但是除了悔棋功能我还要实现复盘功能，所以我采用了vector这个容器，既能对内存进行有效利用，又能更方便地进行步骤跳跃，且QT自带的QVector更是较好的支持了图形界面的使用，所以我认为这次课程设计较好的提高了我对c++课程理论知识的实践利用，提高了我利用c++编程的能力。

最后我还进行了额外的学习，即人机对战，学习了策略树，极大极小值搜索，alpha-beta剪枝等等，可谓是对我的逻辑思维方面的知识进行了极大的补充。

此外，通过自己去图书馆查阅相关书籍，上网查询资料，自己在程序中不断进行调试、修改，我的自学能力得到了很大的提升，这在我看来也是最重要的一个收获了。

但因为我经常是自学，所以总是会囿于己见，但还好我有两个很棒的组员，虽然我的代码能力比他们两人较好一点，但他们有时也能够鲜明的指出我所设计的程序中出现的问题，经过反复的讨论，不但我们的程序日臻完美了，我的团队协作能力也得到了极大的提升。因此在未来我会更多地，主动地去向同学们寻求讨论问题的机会，去争取合作交流的机会，去赢得向老师们、领域精英们深入问题研究的机会。

在总结的最后，再次感谢对我们大力支持的老师、助教、同学，以及陪我一起在代码中遨游了日日夜夜的组员们，这个程序是一个团队真实努力的结晶，非常感谢！