**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 17 日 | | |
| 专 业 | 计算机科学与技术 | 班 级 | 1803 |
| 姓 名 | 王有为 | 学 号 | 20184544 |

报告成绩

|  |
| --- |
|  |

报告撰写的内容与要求  
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。 　  
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。 　  
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，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。 　　  
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。  
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）。

**一·项目简介··············································**

**1·目的···············································**

**2·意义···············································**

**3·任务概况············································**

**4·本人在项目中的分工··································**

**二·项目内容···············································**

**1·系统整体结构········································**

**2·系统框架图··········································**

**3·UML类图···········································**

**4·程序流程图·········································**

**5·重点难点分析及解决方案·····························**

**6·调试难点及解决方法·································**

**7·系统交互界面及结果展示·····························**

**三·总结与体会············································**

1. 项目简介：
2. 目的:

通过对课堂和书本上理论知识的实践，加强我对c++本身的理解。通过对QT的学习，学会如何绘制图形界面。

1. 意义：

编程能力有了提升，知道了图形界面的绘制过程，提高了与他人交流代码的能力，具体包括讲解自己的代码和听懂他人的代码。

1. 任务概况：

首先是完成欢迎界面，游戏界面，帮助界面，复盘界面的整体绘制，其次是完成单独模块的设计，即包含棋盘的绘制，及完成落子的操作，还有完成悔棋功能的实现，以及完成头像的绘制，和完成计时器与操作的绑定，完成滑轮与复盘界面的绑定，并在每个界面之间设立返回按钮完成界面之间的转化。

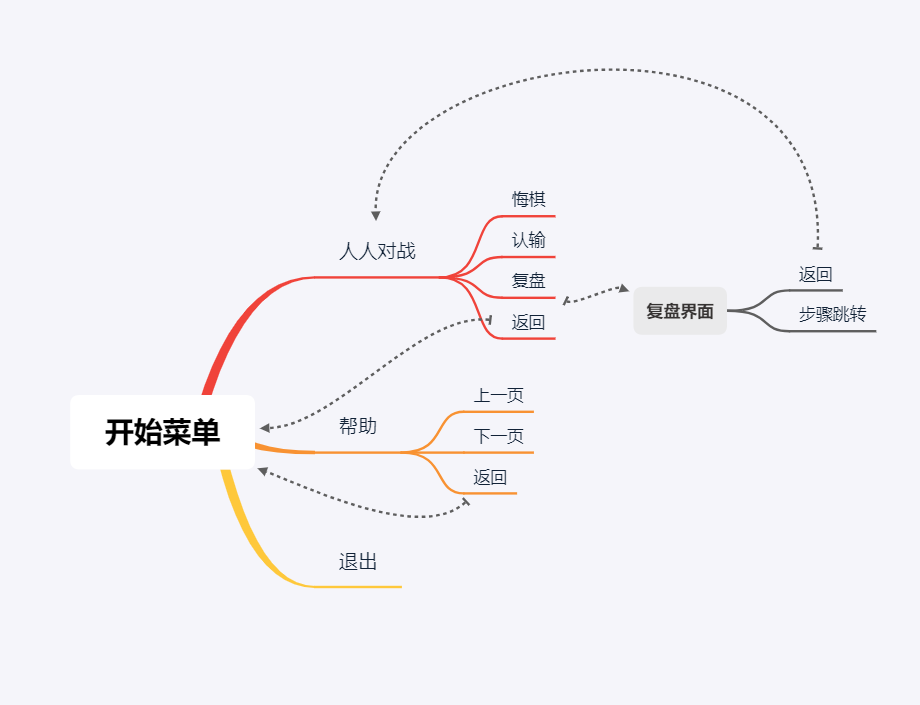
1. 本人在项目中的分工：

我作为小组的一员，经过讨论，主要完成主界面的绘制、实际棋盘的绘制、帮助界面的绘制。其中具体包含整体界面的背景，实际棋盘的各部分颜色选择，棋盘的大小位置规划，帮助界面的背景，帮助界面的换页换图片的操作，以及各个界面的返回功能。

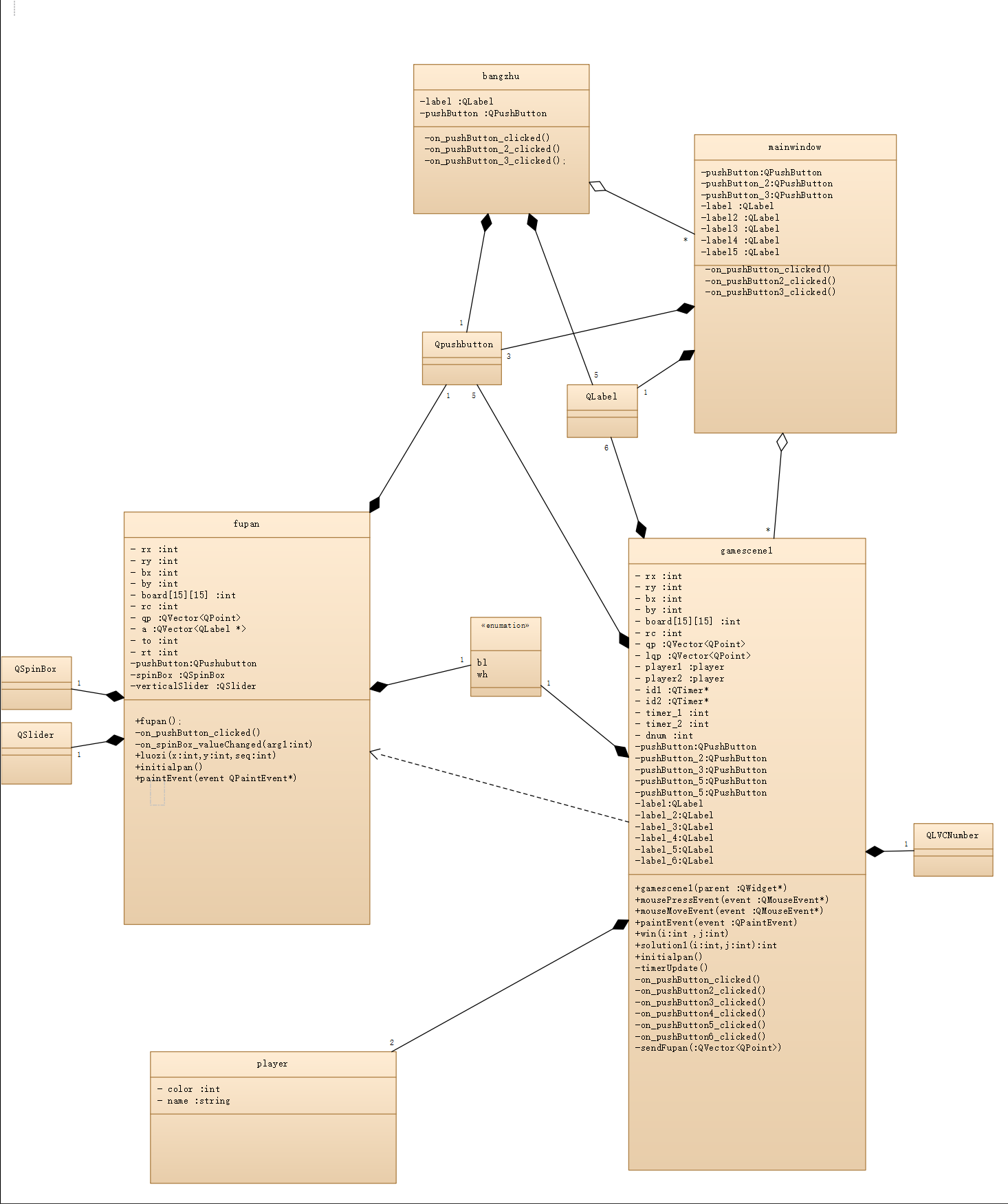
1. 项目内容：
2. 系统整体结构：

本系统总体上分为开始界面、游戏界面、复盘界面和帮助界面，开始界面有人人对战、帮助和退出选项，其中点击人人对战选项进入一个游戏界面，可以实现悔棋、复盘、更改头像等功能，对于复盘功能的实现，我们选择设立一个独立的窗口，并让其能完成上一次游戏的复刻。同时每个独立的窗口都有设置返回选项来完成界面间的跳转。帮助界面包括换页的选项框和返回的选项框，其中通过换页的选项框，即“上一页”、“下一页”可以进行帮助图片的更换。

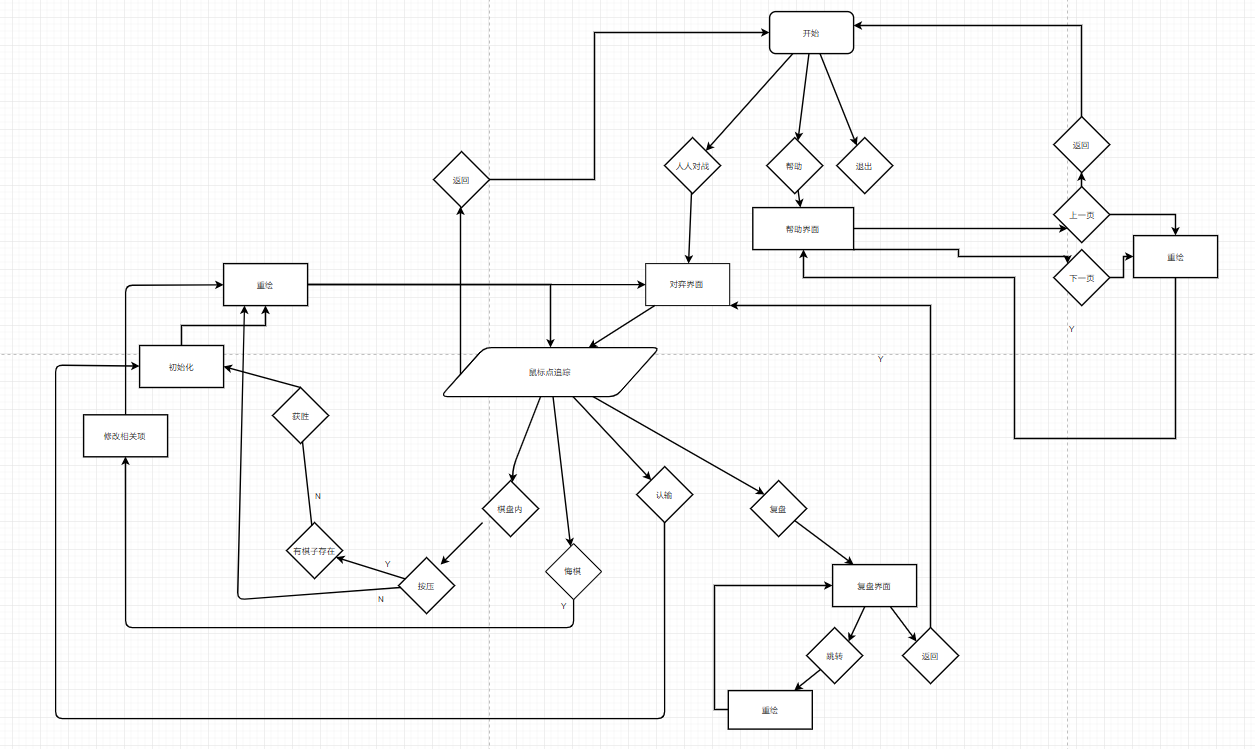
1. 系统框架图：



1. UML类图：



1. 程序流程图：



1. 重点难点分析及解决方案
2. 主界面的绘制

主界面是用户一开始进入的界面，对背景图片的选择自然是要很有讲究的，在我看来这个图片一定需要让用户本身感到舒适感，并且应该能够切合题目，所以我们选择了一个比较暖色的棋盘图片作为除了帮助界面的其他界面的背景图片；此外我们还要仔细规划标题字体的大小，菜单的颜色、大小、摆放的位置等等。当然在进行这些设计之前，首先肯定要先了解如何才能完成背景图片的加载操作，通过网上搜索和QT自带的帮助项目我们知道了如何使用资源库导入图片，并且在这之后对于背景图片本身，我直接在主窗口中对其进行了样式表的改写，这样就能使主界面被设置我所选择的背景图片，同时由于主窗口是其他所有子窗口的父窗口，所以其它窗口默认的背景图片就都是一开始的设计了，这对于系统本身的统一性是很有帮助的。对于这其中所设置的各个菜单跳转项，我们为了让界面显得整洁，我这边使用了一个管理单位来限制他们的大小宽度等，同时我们对样式表完成了较好的设置；为了能够成功地完成界面之间的相互跳转，我通过QT的信号与槽机制，使用了QT中内置的按钮类所具有的一些原有的信号和槽，同时通过设置点击信号和界面创建的槽，即在点击信号所对应的槽中通过动态指针设置新界面的产生和老界面的隐藏，最终我们成功实现了所需要的界面的跳转。

1. 棋盘的绘制

棋盘是五子棋游戏的灵魂，可以说没有棋盘就没有五子棋，对于五子棋棋盘的大小也需要根据界面的大小进行适当的调整，或者说他们两者之间是相互制约的。其中棋盘可以直接利用QT中所自带的画图功能来进行绘制，当然也可以从网上截取图片像制作背景图片一样先把图片放入资源库里，再把图片放在一个固定的标签单位上去，在适当的时候进行显示即可，但是会产生很多显示的BUG，而且实际上并不有利于后续的调试功能。所以最终我采用了自己绘制的方法，即改写了窗口所自带的绘画函数这一虚函数，来进行对棋盘的绘制，具体方案先要了解棋盘本身的构造，即十五乘以十五的黑线围成，其次要了解本身窗口的x轴，y轴，在适当的位置进行绘画函数的调用，然后我们利用了QT自带的调色板类及其自带的相关函数来进行黑线的勾画，同时我们利用了填色函数来进行黄色底盘的绘制。

1. 帮助菜单的绘制

帮助菜单主要是要帮助用户了解游戏机制，了解界面内容，从而增加玩家的游戏体验感。在我看来，帮助菜单要做到简洁明了，要做到直观，所以对于所需的图片我们进行了PS操作，并且用箭头和文字进行了介绍；此外，为了让图片的大小足够大，我们需要设置翻页的操作，并在适当的地方设置翻页的选择框，通过“上一页”、“下一页”来进行帮助提示图片的转换。首先和背景图片一样先在资源库中加载帮助菜单所需的图片，在这之后我设置了两个按钮类来实现页面的转化，具体来说就是根据页数来进行隐藏与显现的操作，之后要根据页数来对页数转换框来进行显示和隐藏的操作，从而达到界面友善的效果。

1. 调试难点及解决方法
2. 棋盘的绘制的调试

经过实践，如果一开始就进行棋盘的绘制，那么它的调试需要对每一步进行控制，这样无疑加大了我的任务量，降低了我的任务效率，所以我选择在进行总体绘制之前先进行各个板块的绘制，因为各单位之间的联系其实并不是很密切，所以我决定最后再将每个板块放在一起构建即可。

对于各个板块图形的绘制，由于QT自带的各类函数，如果在调试时使用step into，就会进入未知领域，常常会变得不知所措。还好调试的过程虽然复杂，但错误点不易出现，对于出现的错误点，我们决定使用QT中丰富的函数来进行相应的替代，避其锋芒，增加了编写效率，也成功的完成了棋盘绘制的实现。

2.对于界面图片的加载

由于QT中不同的类对象具有不同的成员函数，所以在进行图片的加和覆盖、转换时经常会出现图形样子扭曲、大小不正确的问题,而对于调试而言这方面基本很难找出错误的原因，只能找出错误点，对于这个问题，我们通过增加修改大小的代码并在它和实际代码之间设置断点可以较好的知道哪里出现了问题，从而进行因地制宜的解决。具体来说，我们先在新建立的项目里进行图像加载的测试，再复制到源代码中进行界面图片的加载。同时为了保障程序的稳定性，我们测试了一次导入多个图片的效果，最终发现有时会出现程序崩溃的问题，经过断点的设置及不断的调试，我们最终知道是内存溢出了，此后我们选择对图片进行单独加载而不是集体加载的方式，最终成功解决了部分内存溢出问题。

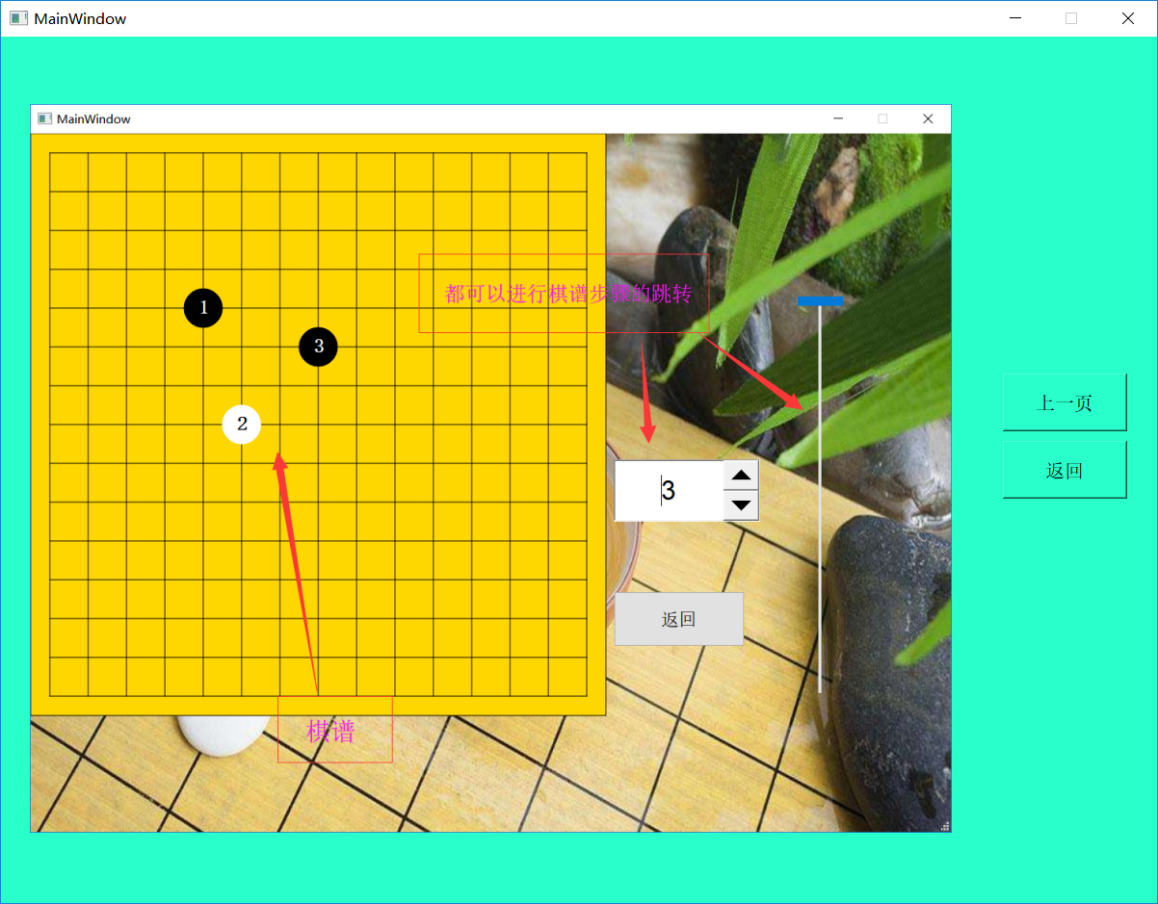
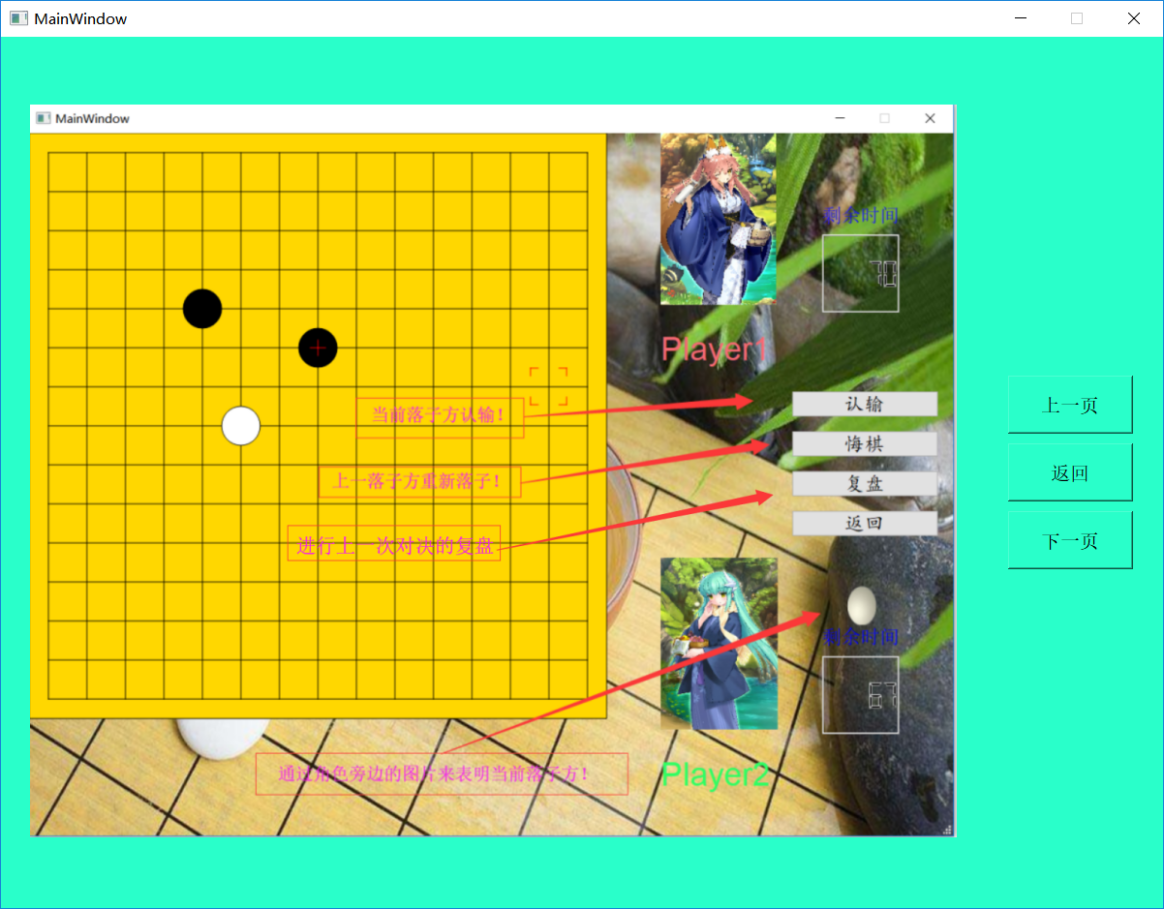
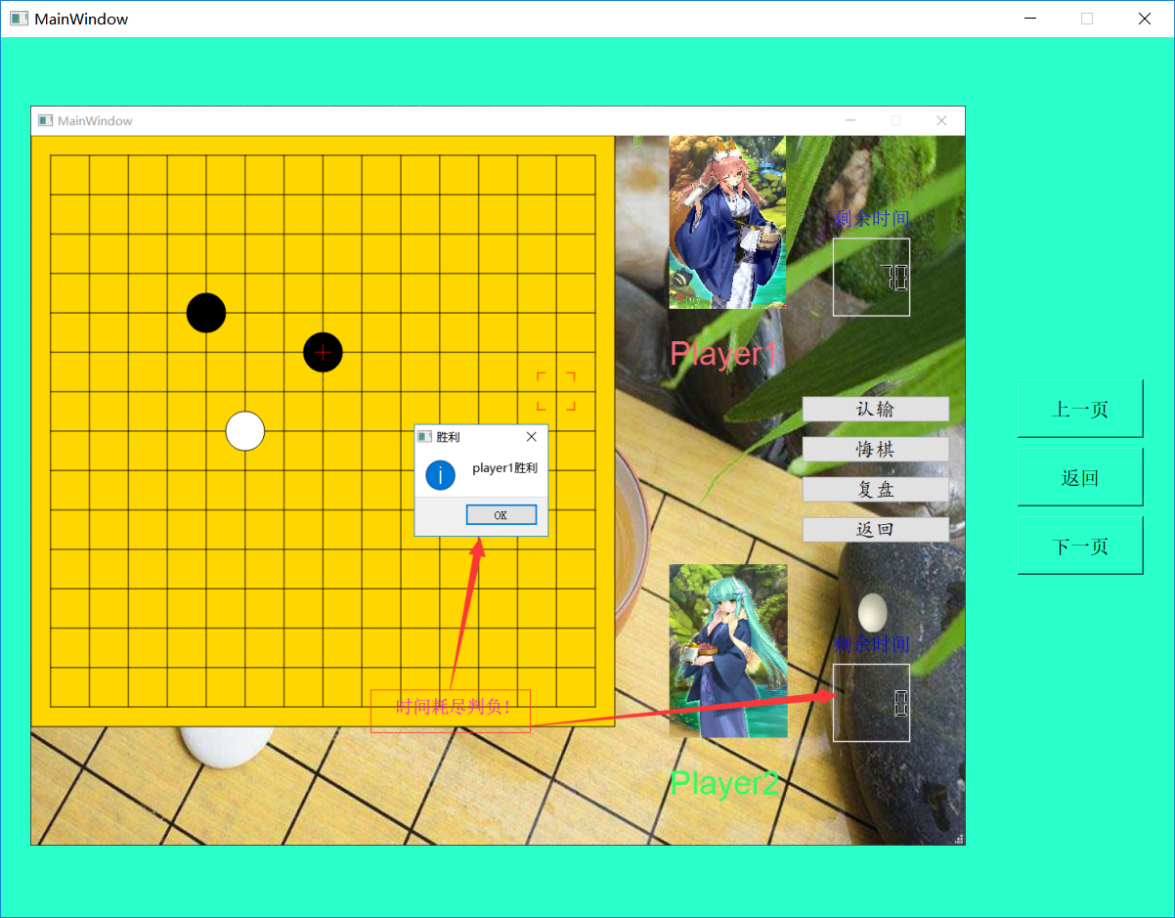
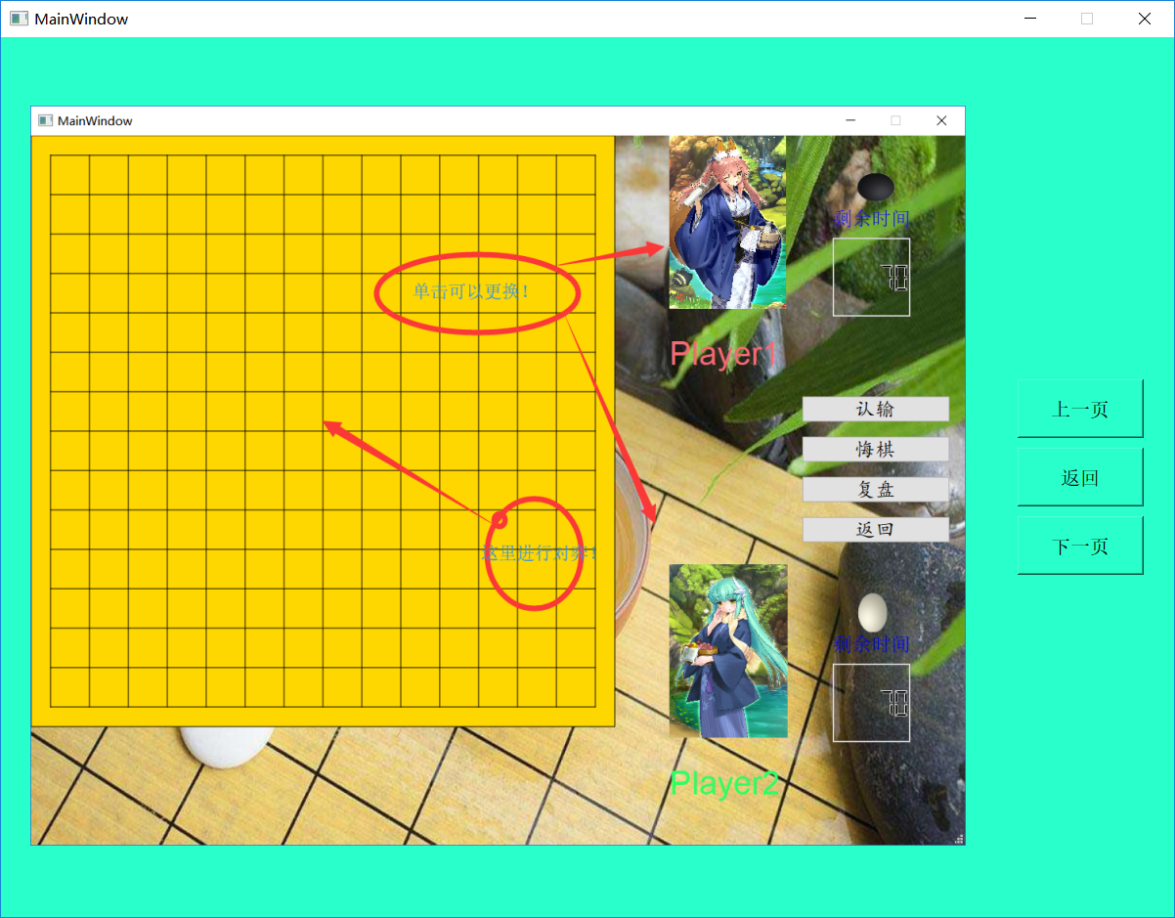
此外，我们最后选择在工程中添加CONFIG += resources\_big这一行代码，即让程序不限制图片加载的大小，不但最终成功完成了目标所需，还对复盘功能的实现提供了帮助，因为根据代码来说，复盘功能的实现是需要预先加载二百五十五的图片项，所以如果能够不限制程序内存的大小可以很好的加载各个板块，也就是说我们利用内存换取了加载速度。

3.帮助菜单绘制的调试

对于帮助菜单的绘制的调试，在我看来，是和棋盘的绘制同理的，由于我们要先对每个板块进行独立的配置，再整体构建在一起，但与此不同的是，这里不能简单的将各个模块堆砌在一起，而是要通过信号与槽机制达到有机的统一，从而实现翻页更换图片的功能。所以对于帮助菜单的调试，我们追踪了每一次信号的来源，并在适当的地方设置断点，查询信号是否发送成功，以及槽是否接受了信号。具体来说，每次构建标签类的单位时我们都设置了断点进行追踪，并在显示窗口中实时显示页数，接着我通过一步一步的调试来确认图片的显示的成功与否，最后再实际运行中通过不断的点击来检测程序是否会因此而产生内存溢出等问题。

1. 系统交互界面及结果展示：





1. 总结或体会：

首先，请允许我在叙述总结或体会前先感谢一下对我们进行鼓励、鞭策，在我们遇到困难的时刻慷慨解囊的老师、助教、同学们，没有他们的帮助我想我很难单独完成本次的课程设计任务。

在我看来，我们成功达到了本课程的规定的要求，甚至我觉得我们在某些地方还有了额外的拓展项，此外组员之间的合作与交流也越来越默契，可以说这是一次较为成功的程序设计了。

本次课程设计加强了我的编程能力，通过对QT的使用，我学会了如何用代码进行图形的绘制，如何完成一个界面的绘制和设计，如何实现QT中信号与槽的机制，。由于我本身代码能力其实较弱，相关函数的实现达不到，但是经过其他组员和老师、同学的帮助，最终我还是成功的实现了组长所规划给我的任务。

具体来说，我知道了只在课堂上学到知识是不够的，只有经过实践后才能知道自己到底有没有懂得知识本身。在一开始我还没有重视课程设计，但在组长给我分配任务我去实现时才发现其实这次课程设计并没有想象中那么简单，对于图形界面的绘制等等都是我的第一次，我一开始完全没有如何绘制一个完整的图形界面的概念，因为之前一直都习惯于控制台的相关显示，并不知道该怎么去绘制出一个图形界面。之前的课程设计中有设计过花的绘制，但是那时是通过函数进行等于0的判断画出来的。所以本次进行棋盘绘制确实让我在这一方面进步了很多很多！具体来讲：第一QT给我提供了很多便捷的函数和类，让我很方便的就能画线，画方块，填色，加载图片等。但是有利有弊，由于QT具有非常多的关于图形绘制界面的类与对象，所以对QT界面管理的学习的过程不但非常难，而且非常多、非常枯燥，不过经过长时间的学习和实践后，特别是在和各位组员的讨论后，还是较为成功的把所要求的界面画出来了。在实际画出来之后还是有非常大的成就感的。现在每次看着自己亲手制作的十分漂亮的界面，就会更有了学习的动力，更希望能去深入了解相关的内容了。所以毫无疑问，本次课程设计极大地提高了我对学习的兴趣!第二对于图片的把握也有了很好的掌控，我第一次试着去调整各个模块应该所处的位置，第一次去一次次地调试画图函数，选择我所认为的最好的颜色搭配，选择我所认为的最优的线条粗细，在我看来，这都是十分有趣的。

而且，除此之外，在组里大家的讨论之后，我还学到了很多很多，如如何高效地进行STL容器使用，如何进行策略树的构建，什么是QT特别的信号与槽机制等等，让我一个人的时间通过大家的努力成倍增加了，我想这就是常说的头脑风暴的魅力，一个人有时在程序编完后自己操作时的成就感真是很大!此外，通过对棋盘的绘制和对各个界面的设计，我还增加了一些美术上的启迪——对称性，简洁美等等，获益匪浅。

针对本次课程设计中所暴露的代码能力较弱，实践能力不足的问题，我决定在以后的学习中，加强实践，在实践中提高代码能力，弥补自身不足。