程序设计实践报告

**五子棋游戏**

**课程名称 程序设计**

**实践形式 五子棋游戏制作**

**所在学院**

**年级班别**

**学 号**

**学生姓名**

**队 友 一人完成**

**指导教**师

年 月

**摘 要**

通过一学期的c++语言学习，已经掌握了数据类型、数组、指针，类等c++语言基础。根据课程实践的要求，通过已有的知识储备以及在课程过后个人的学习，完成了课程要求中的五子棋游戏的制作。由于程序使用了mysql数据库，所以程序无法实现打包发出，让别人测试。不过，若他人电脑中已配置了mysql数据库及环境，只需要修改数据库账号，密码及表名即可使用。

**关键词**：五子棋、QT、mysql

**目 录**

**[一、 总体设计 1](#_Toc18901_WPSOffice_Level1)**

[1.1界面设计 1](#_Toc20522_WPSOffice_Level2)

[1.2功能设计 1](#_Toc9071_WPSOffice_Level2)

[1.3代码实现 1](#_Toc21108_WPSOffice_Level2)

[1.4小结 1](#_Toc20250_WPSOffice_Level2)

**[二、详细设计 2](#_Toc20522_WPSOffice_Level1)**

[2.1界面设计详细讲解 2](#_Toc28970_WPSOffice_Level2)

[1）登陆界面 2](#_Toc20522_WPSOffice_Level3)

[2）注册界面 3](#_Toc9071_WPSOffice_Level3)

[3）游戏界面 4](#_Toc21108_WPSOffice_Level3)

[2.2 部分功能详解 5](#_Toc4777_WPSOffice_Level2)

[1) mysql数据库 5](#_Toc20250_WPSOffice_Level3)

[2) 背景设置 6](#_Toc28534_WPSOffice_Level3)

[3) 五子棋棋盘棋子绘画 7](#_Toc28970_WPSOffice_Level3)

[4）人机代码的编写 7](#_Toc4777_WPSOffice_Level3)

[5） 计时器 8](#_Toc3466_WPSOffice_Level3)

[6） Mainwindow的菜单栏的创建 8](#_Toc17722_WPSOffice_Level3)

[7） 设置（setting） 9](#_Toc4577_WPSOffice_Level3)

[8） 操作（operation） 1](#_Toc19131_WPSOffice_Level3)0

[2.3 总结： 1](#_Toc3466_WPSOffice_Level2)1

**[三、实验过程： 1](#_Toc9071_WPSOffice_Level1)1**

[3.1 配环境 1](#_Toc17722_WPSOffice_Level2)1

[3.2 编写代码 1](#_Toc4577_WPSOffice_Level2)2

[3.3无法解决的问题： 1](#_Toc19131_WPSOffice_Level2)2

[1） 个人信息无法显示： 1](#_Toc5616_WPSOffice_Level3)2

[2）排行榜无法显示问题： 1](#_Toc7285_WPSOffice_Level3)2

[3.4 总结： 1](#_Toc5616_WPSOffice_Level2)3

**[四、 关于未来： 1](#_Toc21108_WPSOffice_Level1)5**

**[五、 总结 1](#_Toc20250_WPSOffice_Level1)5**

**[六、参考文献 15](#_Toc28534_WPSOffice_Level1)**

**[七、致谢 15](#_Toc28970_WPSOffice_Level1)**

1. **总体设计**

程序框架

注册界面

登陆界面

功能

**游戏主页面**

模式

主窗口

暂停

悔棋

双人对战

退出

投降

人机对战

**1.1界面设计**

本程序有三个界面，分别为登陆界面，注册界面，游戏界面。在界面上，一切设计都以为用户提供更好的使用体验为原则而设计。本游戏界面均用QT设计，通过定义控件及槽函数组成。

**1.2功能设计**

本程序可以实现基本的五子棋的双人对战，人机对战，计时，撤回，投降，认输，暂停，换背景等功能，还有数据库支持的登陆，注册，玩家信息系统等功能

**1.3代码实现**

在代码的实现上，基础界面的代码为QT中的自己手写槽函数、控件和类，手动摆放控件位置来实现的，并为使用ui界面。本五子棋的代码核心是窗口类的切换及游戏框架类的应用，以此来实现较多的基本功能。

**1.4小结**

该五子棋程序的功能基本实现，通过精美简洁的界面设计和使用方式，给用户提供更好的使用体验。可惜的是本程序基于自己的mysql数据库实现，没法打包，但是若其他人想要使用这个代码，可以更改其中的数据库账户密码，或者创造个.db的静态数据库来储存。即可使用本程序的代码。

**二、详细设计**

**2.1界面设计详细讲解**

**1）登陆界面**

****

登陆界面为一个c++类，属于widget类。背景由两张图片组成。可在此界面中输入账号密码，账号密码将会在数据库中搜索，若与数据库中的数据一致，即可登陆到游戏界面中。

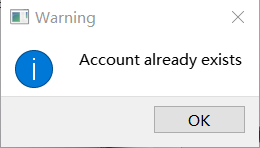
**2）注册界面**

****

注册界面为一个c++类，属于widget类。可以输入账号及注册的密码。

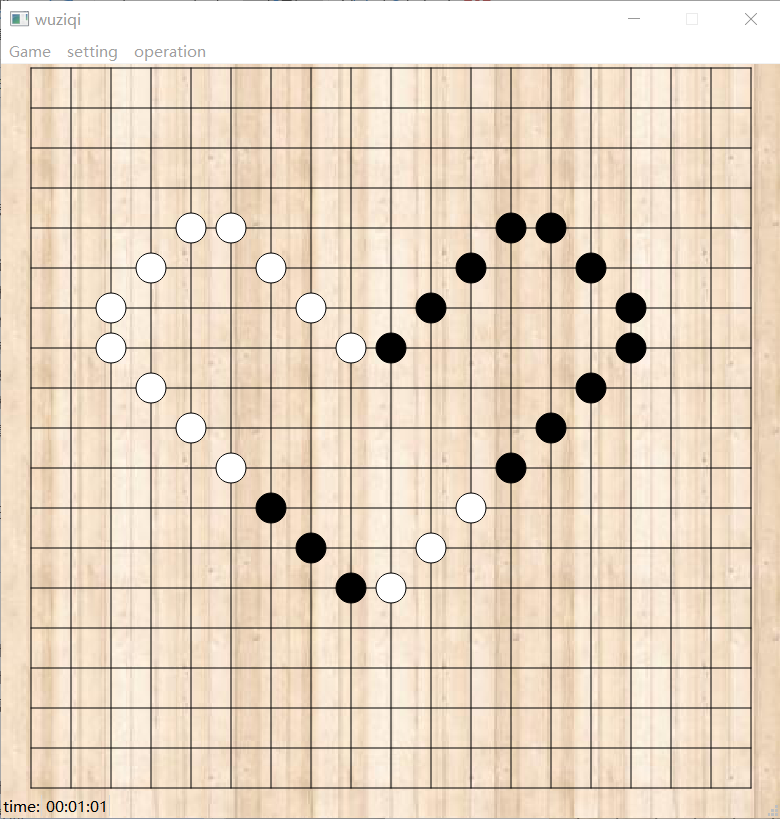
注册时会搜素mysql数据库的已存在账号。

当注册账号已存在时：



同时，注册账号密码不可为空

**3）游戏界面**



注册界面由两个类组成，属于mainwindow类。一个为界面绘画类，一个为下棋类（包括人工AI下棋）

由于程序总是出bug，让我在显示数据库数据时出了问题，导致个人信息界面和排行榜界面没能显示出来，试过很多种方法还是不行，只能暂时放弃这两个功能。

**2.2 部分功能详解**

**1) mysql数据库**

首先是连接数据库，这个在很多个.cpp里都用到了。

代码如下：

qDebug()<<QSqlDatabase::drivers();

//添加MYSQL数据库

QSqlDatabase db=QSqlDatabase::addDatabase("QMYSQL"); //数据库对象

//连接数据库

db.setHostName("127.0.0.1"); //数据库服务器IP

db.setUserName("root"); //数据库用户名

db.setPassword("holdworld000312"); //数据库密码

db.setDatabaseName("slimes"); //使用哪个数据库

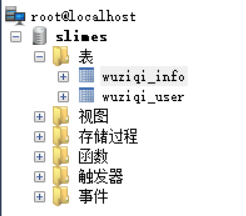
if(!db.open()){

QMessageBox::information(this, "连接结果", "连接数据库失败!");

return;

}

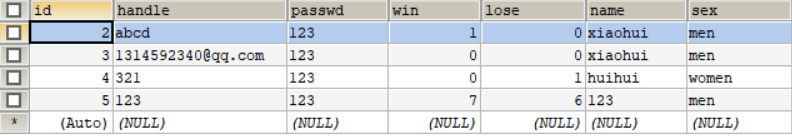
下面是我的mysql中的数据库。（这里数据库可视化我用SQLyog）



数据库中有两个表。

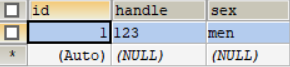
一个是wuziqi\_info

用来存取注册后的个人信息。



其中包括id，账号，密码，赢的局数，输的局数，姓名，性别。

另一个是我用来存储登陆的账户，以便于反馈给各各界面是哪个用户在玩游戏。



中间涉及到的数据库操作有：

【1】 查看登陆账号密码是否正确

QSqlTableModel model;

model.*setTable*("wuziqi\_info");

model.*setFilter*(tr("handle = '%1' and passwd = '%2'").arg(handle).arg(passwd));

model.*select*();

【2】注册时存入新用户的信息

QString str=QString("insert into wuziqi\_info(handle, passwd, win, lose, name, sex) values('%1','%2', 0 , 0 ,'%3','%4')").arg(handle).arg(passwd1).arg(name).arg(sex);

QSqlQuery query;

query.exec(str);

【3】在游戏输赢时获取用户的输赢局数，然后决定哪个+1，再修改表中的数据

QString win;

//这里就举例一个

QString sq=QStringLiteral("select \*from wuziqi\_info where handle='%1' ").arg(NAME);

QSqlQuery sqlquery=QSqlQuery(db);

sqlquery.exec(sq);

while (sqlquery.next()){

win.append(sqlquery.value(3).toString());

}

//这里就举例一个

QSqlQuery query;

query.prepare("update wuziqi\_info set win=? where handle=?");

query.addBindValue(newwin);

query.addBindValue(NAME);

query.exec();

1. **背景设置**

用画笔Qpainter 在***paintEvent***(QPaintEvent \*event)，获取图片名称（图片存储于与此代码统一位置中，即可实现背景画图）

//QPainter p(this);

QPainter p; //创建画家对象

p.begin(this); //指定当前窗口为绘图设备

//绘图操作

//画背景图

if(color == 0)

p.drawPixmap(0,0,width(),height(), QPixmap("../Image/u=3967820435,4109070133&fm=26&gp=0.jpg"));

**3) 五子棋棋盘棋子绘画**

用到***paintEvent***(QPaintEvent \*event)这个函数，可以实时更新页面。

用painter.drawLine画棋盘的格子。

画棋子（颜色，粉刷）：

brush.setColor(Qt::white);

painter.setBrush(brush);

painter.drawEllipse(kBoardMargin + kBlockSize \* j - kRadius, kBoardMargin + kBlockSize \* i - kRadius, kRadius \* 2, kRadius \* 2);

用***mouseMoveEvent***(QMouseEvent \*event)通过鼠标的hover确定落子的标记

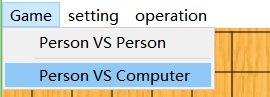
还有用一个update()函数，每次都重绘，即时时更新。

程序中用了一个数组game->gameMapVec[clickPosRow][clickPosCol]为0的时

候没有棋子，为1的时候有棋子。

**4）人机代码的编写**

人机代码我放在一个GameModel类中，有个初始话评分系统，从各种角度对每个格子的分数进行加分。这里我用了暴力搜索每个点的四个方向哈哈，其实如果要降低复杂度可以用dfs来优化。然后就两个量personNum（botNum），emptyNum用来记录自己连子的个数，和对方连子的个数。最后分别给黑子进行统一打分。分数高的位置优先考虑下子。



（在这里选择人机或非人机操作）

目前还有缺陷，就是计分时，没有考虑到优先取胜策略，只是在防守，不过，给宿友们测试了一下，发现人机还是很厉害的，成功将宿友的耐心消磨哈哈。当然可以通过修改每个落点的得分，来安排人机的攻守风格。

部分代码：

for (int i = 1; i <= 4; i++){

if (row + i \* y > 0 && row + i \* y < kBoardSizeNum &&col + i \* x > 0 && col + i \* x < kBoardSizeNum &&gameMapVec[row + i \* y][col + i \* x] == 1) // 玩家的子{

personNum++;

}

else if (row + i \* y > 0 && row + i \* y < kBoardSizeNum &&col + i \* x > 0 && col + i \* x < kBoardSizeNum &&gameMapVec[row + i \* y][col + i \* x] == 0) // 空白位{

emptyNum++;

break;

}

else// 出边界

break;

if (personNum == 1) // 杀二

scoreMapVec[row][col] += 10;

else if (personNum == 2) // 杀三

{

if (emptyNum == 1)

scoreMapVec[row][col] += 30;

else if (emptyNum == 2)

scoreMapVec[row][col] += 40;

}

else if (personNum == 3) // 杀四

{// 量变空位不一样，优先级不一样

if (emptyNum == 1)

scoreMapVec[row][col] += 60;

else if (emptyNum == 2)

scoreMapVec[row][col] += 110;

}

else if (personNum == 4) // 杀五

scoreMapVec[row][col] += 10100;

}

由于棋盘不大，就225个格子，每个格子八个方向扫一遍，225\*8\*5=9000

所以每次操作最复杂的时间复杂度才O(1000)只需要0.00001秒，这里就不节省时间复杂度啦。

1. **计时器**

写个状态栏，将计时器放于状态栏中。用到time\_update()函数，可以时时更新时间

currentTimeLabel = new QLabel; // 创建QLabel控件

QStatusBar \*sBar = statusBar();

sBar->addWidget(new QLabel("time:",this));

sBar->addWidget(currentTimeLabel);

timer = new QTimer(this);

time = new QTime(0,0,0);

timer->start(1000); //每隔1000ms发送timeout的信号

connect(timer, SIGNAL(timeout()),this,SLOT(time\_update()));

void MainWindow::**time\_update**()

{

//time->addSecs(1);

\*time=time->addSecs(1);

//QDateTime current\_time = new QDateTime(0,0,0);

QString timestr = time->toString("hh:mm:ss"); //设置显示的格式

currentTimeLabel->setText(timestr); //设置label的文本内容为时间

}

1561465122(1)

（本计时器可以在游戏结束或认输时重计，在暂停时停止）

1. **Mainwindow的菜单栏的创建**

举例：

// 添加游戏菜单

QMenu \*gameMenu = menuBar()->addMenu(tr("Game")); // menuBar默认是存在的，直接加菜单就可以了

QAction \*actionPVP = new QAction("Person VS Person", this);

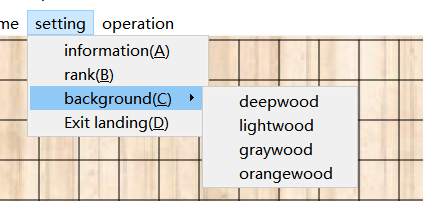
connect(actionPVP, SIGNAL(triggered()), this,SLOT(initPVPGame()));

gameMenu->addAction(actionPVP);

1. **设置（setting）**

Information 和rank 都还没完成，这里先不介绍。

**【1】换背景（Background）：**



四种背景可供选择。

Deepwood：



Lightwood：



Graywood：



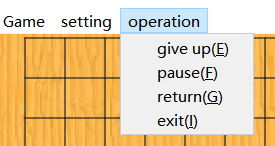
Orangewood：



【2】**退出登陆（Exit landing）:**

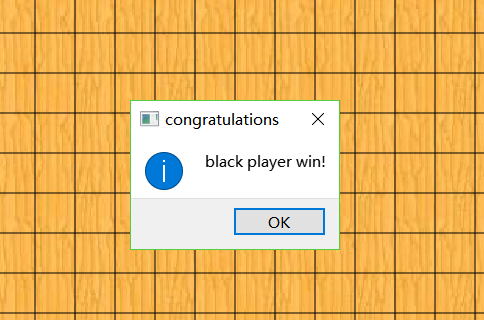
这是退出登陆的按钮，可以通过这个回到登陆界面，换另一个账号继续玩本游戏。

1. **操作（operation）**



【1】**投降（give up）**

投降后自己的输的局数在数据库中的记录会+1，同时计时器重新计时。（自己为白方）

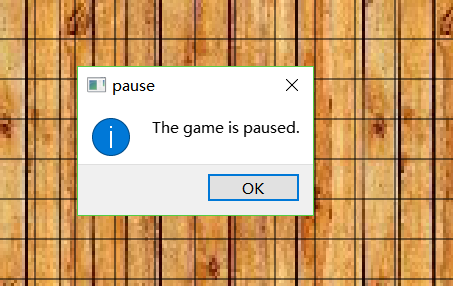


【2】**暂停（pause）**

暂停后，玩家将无法下棋，计时器也会随之暂停。

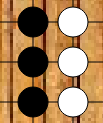
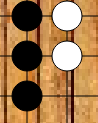
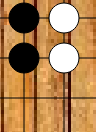
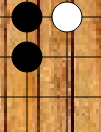
**总结：**

很多功能都是慢慢想到的，然后再加上去，



【3】**悔棋（return）**

这里我用了个二维数组来存储每次放棋后的操作，若点击撤回，则now的位置变为0，然后now-1，到上一个状态。

**【4】退出登陆（exit）**

退出游戏。

**2.3 总结：**

这次的代码大部分是边学边完成的，所以框架较凌乱，有一些功能无法实现。但基础功能还是实现啦，由于一开始写程序框架的时候没有使用ui去做，而是自己手写控件和槽函数，所以界面比较简陋。但是基本功能都差不多实现啦。

**三、实验过程：**

**3.1 配环境**

本来想着用visual studio 2017连接mysql数据库来写程序的。但是我配了一个多星期的环境加上静态库，还是会报出一堆错误，所以果断放弃visual studio 2017 ，改用QT5.7+mysql，完成程序的编写。

本人先写了游戏框架及规则，在编写完mainwindow界面后才逐渐添加其他界面。在游戏编写的过程中，也是遇到了许许多多的问题，比如可以在网格外放棋子等。在观看众多博客后，才慢慢的将bug修改完。

之后，我才开始编写界面类，先写了登陆界面，这时候，我开始尝试数据库的连接，在我的电脑的path路径中加入mysql.dll静态数据库的路径后，总算成功连接上了数据库。之后便学了一系列的数据添加，读取，搜索。才成功实现了登陆界面及注册界面的编写。但这时候我面临了一个问题。

**3.2 编写代码**

因为，其实从一开始，界面就全部存在，所以，每个界面拥有的都是在那一时刻的数据，无法更新到最新，就比如我想要的个人信息查询，排行榜。。。（这两个只是显示上除了问题，获取最新的数据我已经解决了），而且每个界面都是，独立的，无法知道是哪个用户在进行游戏。当时的第一做法便是，运用信号和槽。通过登陆界面，放送信号给其他窗口。但是却在主窗口传信息给子窗口时出现了问题。而且窗口众多，导致代码越写越乱。所以我便创造了个数据库的表，用来记录，此时在线的玩家的信息。最后就是每个窗口获得最新的信息的问题了。这里，因为要让窗口显示，肯定要先按一个按钮，所以，我在按钮里增加信息的更新。这样子，这个问题就被解决啦。编写这个代码中间还遇到了大大小小的问题，有些解决了，有些仍无法解决。

**3.3无法解决的问题：**

1. 个人信息无法显示：



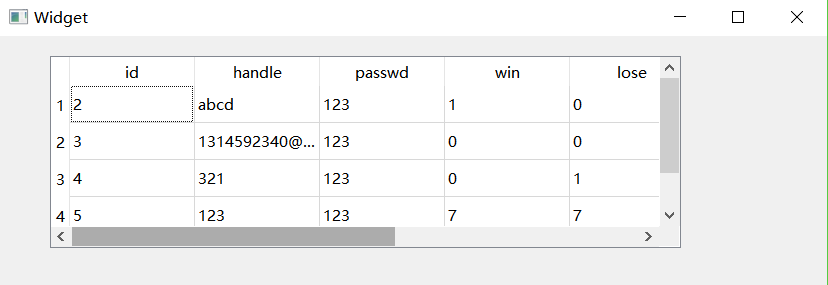


如图，我已经将数据库中的用户信息获取，并用qDebug输出，但是我尝试了好几种方法，还是不能将其显示在窗口上。现在只能根据男女，给用户安排了个头像哈哈。

2）排行榜无法显示问题：



因为排行榜很多数据，所以我想要将数据库的数据直接排序，然后再可视化。但是这时候又出现问题了，也许因为不是主窗口，所以，我可视化的表格里，都是空白的。但是，当我重新创个项目，去尝试这个代码时，又可以显示。



。

3.4 总结：

编写代码中遇到好多问题，大部分得以解决，可是还有几个至今都还无法解决，不过，等有一定的时间后还是应该将其解决，加上以后，我希望可以给该游戏连上区域网，实现玩家的对战。

1. **关于未来：**

虽然这只是一个课设，但是我的完成程度还让我很不满意，还有很多的功能无法实现，还有实现玩家联网对抗等等，慢慢的我会在进行改进，希望可以不断完善，最后将其变成一个app应用。所以在即将到来的暑假中，我会花一定的时间去考虑如何实现以上功能，从而更好的完善该程序的功能。当然我还想加入围棋等各种棋类，慢慢的去更多了解界面设计的其他知识点。让自己学到的知识更多。

1. **总结**

之前社团考核用过c语言去写了一个部员管理系统，还有写过成绩管理系统，不过，当时的代码，还是只能在cmd窗口上运行，文件只能用txt保存，用链表来存数据，但是这次课设，成功脱离了小黑框，有了比较美观的界面，有很大的成就感。当然，主要还是得满足对客户的需求，所以我请了挺多同学来测试我的程序的，反响还行，人机也挺给力的哈哈。不过还是有很多不妥的地方需要我慢慢的改进，慢慢的去满足用户的需求。

通过这次课设的学习，我还是学到了好多，比如类，基类，派生类，继承，信号与槽等。对于界面设计，虽然不是主方向，但是，以后一定很多用得上的时候，所以这次的课设让我收获很大很大。

六、**参考文献**

【1】哔哩哔哩视频：妙韵含的QT开发全套视频

**七、致谢**

在这里，我要感谢\*\*老师给予我的指导和帮助，没有老师的指明方向，我现在肯定还是一头雾水的。这学期的c++程序设计课也接近尾声了，感谢老师传授我们的知识，还有对我的关注。