**《数据结构与算法实验课》**

**综合应用报告**

**设计题目: 家谱管理系统**

**院 系: 信息学院**

**组 长: 徐晶**

**组 员: 曹默涵、赵佳豪**

**指导教师: 周琳**

**日 期: 2023.12.22**

**目录**

[1 实验内容与要求 3](#_Toc28931)

[1.1 需求分析 3](#_Toc26485)

[1.2 系统概述 3](#_Toc26485)

[2 实验环境 3](#_Toc7876)

[2.1 硬件环境 3](#_Toc27211)

[2.2 软件环境 3](#_Toc3580)

[3 分工及组内得分 4](#_Toc17223)

[4 系统分析及设计 4](#_Toc16359)

[4.1 概要设计 4](#_Toc27211)

[4.2 抽象数据类型 4](#_Toc3580)

[4.3 存储结构设计 6](#_Toc29641)

[4.4 功能设计](#_Toc27211) 7

[5 运行结果 8](#_Toc9376)

[6 总结 1](#_Toc9600)4

[6.1 数据结构与算法实验课-综合应用支撑的毕业要求指标点 14](#_Toc27211)

[6.2 心得 14](#_Toc27211)

[7 程序源代码 15](#_Toc30198)

# 题目

## 设计内容与要求

1.1需求分析

家谱是一种特殊的文献，就其内容而言，是中国五千年文明史中具有平民特色的文献，记载的是同宗共祖血缘集团世系人物和事迹等方面情况的历史图籍。家谱属珍贵的人文资料，对于历史学、民俗学、人口学、社会学和经济学的深入研究，均有其不可替代的独特功能。

谱：又称族谱、宗谱等。是一种以表谱形式，记载一个家族的世系繁衍及重要人物事迹的书。皇帝的家谱称玉牒，如新朝玉牒、皇宋玉牒。它以记载父系家族世系、人物为中心，由正史中的帝王本纪及王侯列传、年表等演变而来。

目前，很多家庭家谱丢失，家庭一些关系混乱，为了避免这一问题，制定家谱程序。家谱用于记录某家族历代家族成员的情况与关系，实现对一个家族所有的资料进行收集整理。支持对家谱的存储、更新、查询、统计等操作。并用计算机永久储存家族数据，方便随时调用。

1.2 系统概述

本系统面向家族成员，实现家谱的构建和查询，同时，提供修改，添加，数据更新等功能，以保证系统运行的稳定性

## 实验环境

2.1硬件环境

PC机内存：16g 硬盘：512g

2.2软件环境

Win10/11操作系统

C++17，图形化界面使用QT

具体版本：QT6.6.0

编译器：MinGW\_64\_bit

## 分工及组内得分

|  |  |  |  |
| --- | --- | --- | --- |
|  | 组长 | 组员一 | 组员二 |
| 姓名 | 徐晶 | 曹默涵 | 赵佳豪 |
| 学号 | 2022905415 | 2022901532 | 2020903623 |
| 组内得分 | 80 | 80 | 100 |
| 组员分工 | 实现对家谱成员的信息各个查询功能以及部分树的功能 | 实现对家谱成员的信息修改以及成员添加功能 | 对整个QT界面的编写，包括登陆界面和操作界面；以及实现各个类的定义、槽函数的链接；最后的整合和运行 |

## 系统分析与设计

4.1概要设计

由于家谱是一颗树结构，而不是一颗二叉树，所以在存储时要转换成二叉树的形式，本程序只考虑了男性家族成员，实现和遍历时结构更加简单

4.2抽象数据类型

ADT Tree{

数据对象D：D是具有相同特性的数据元素的集合。

数据关系R：若D=Φ，则R=Φ，称Tree为空二叉树；

若D≠Φ，则R={H}，H是如下二元关系；

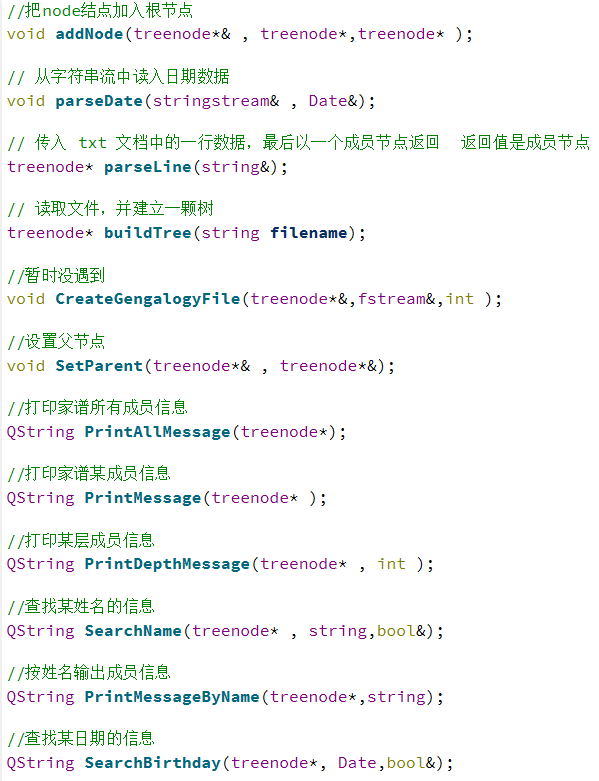
（1）在D中存在惟一的称为根的数据元素root，它在关系H下无前驱；

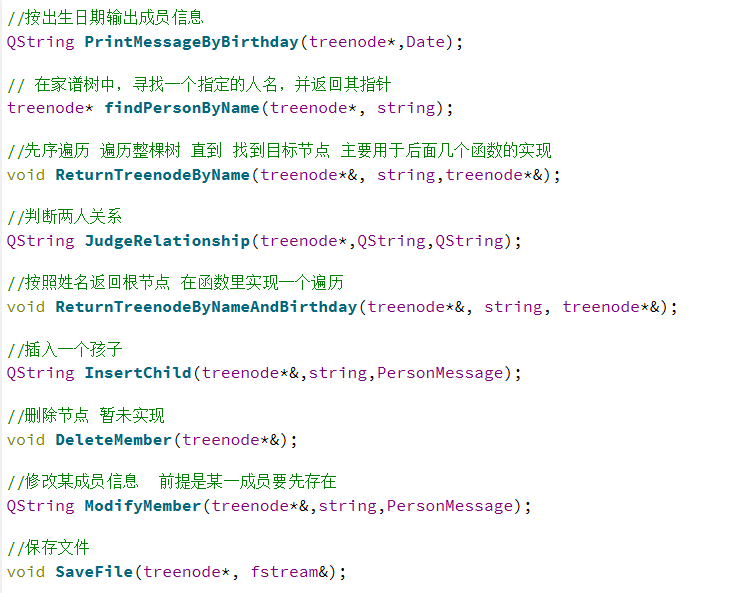
（2）若D-{root}≠Φ，则存在D-{root}={D1,Dr}，且D1∩Dr =Φ；

（3）若D1≠Φ，则D1中存在惟一的元素x1，<root,x1>∈H，且存在D1上的关系H1 ⊆H；若Dr≠Φ，则Dr中存在惟一的元素xr，<root,xr>∈H，且存在上的关系Hr ⊆H；H={<root,x1>,<root,xr>,H1,Hr}；

（4）(D1,{H1})是一棵符合本定义的二叉树，称为根的左子树；(Dr,{Hr})是一棵符合本定义的二叉树，称为根的右子树。

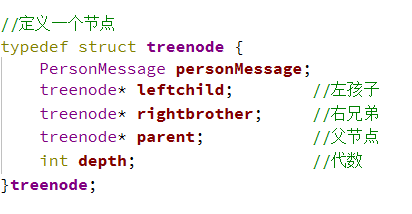
基本操作 P：





4.3存储结构设计

本系统定义三个结构体分别用来保存日期、成员信息、成员关系，如下图所示：



4.4功能设计

本程序的各个功能模块涉及若干函数，如下表所示：

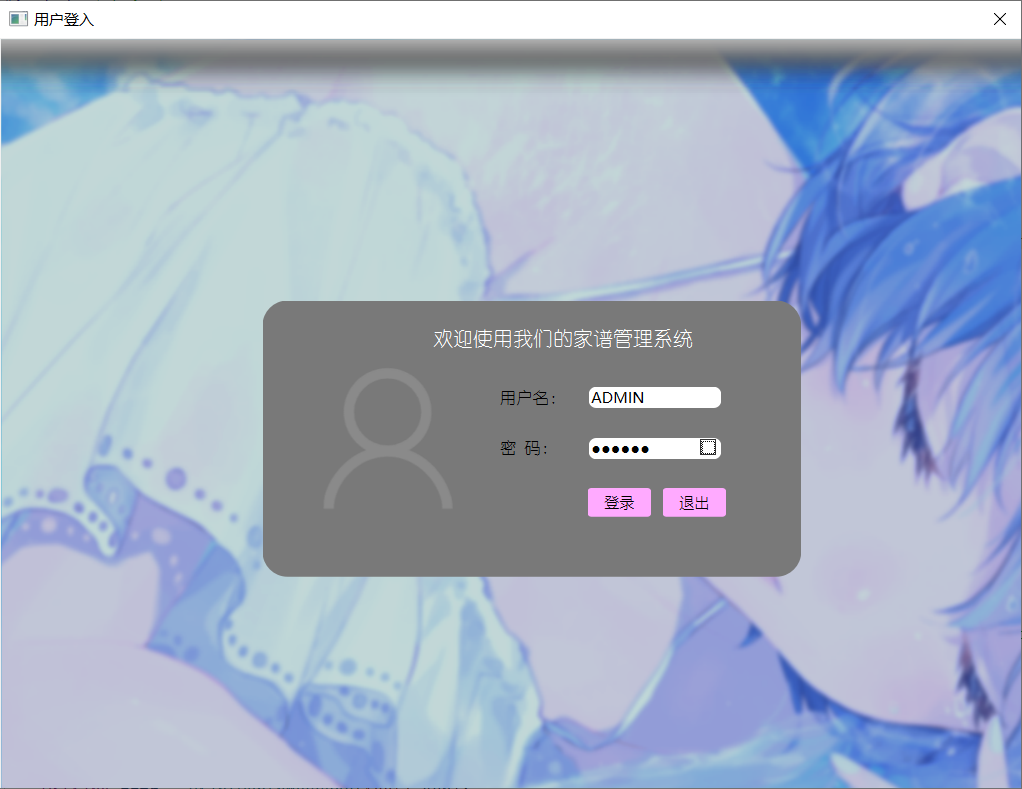
|  |  |
| --- | --- |
| 函数名称 | 功能简述 |
| void addNode(treenode\*& , treenode\*,treenode\* ); | 把node结点加入根节点 |
| void parseDate(stringstream& , Date&); | 从字符串流中读入日期数据 |
| treenode\* parseLine(string&); | 传入 txt 文档中的一行数据，最后以一个成员节点返回 返回值是成员节点 |
| treenode\* buildTree(string filename); | 读取文件，并建立一颗树 |
| void SetParent(treenode\*& , treenode\*&); | 设置父节点 |
| QString PrintAllMessage(treenode\*); | 打印家谱所有成员信息 |
| QString PrintMessage(treenode\* ); | 打印家谱某成员信息 |
| QString PrintDepthMessage(treenode\* , int ); | 打印某层成员信息 |
| QString SearchName(treenode\* , string,bool&); | 查找某姓名的信息 |
| QString PrintMessageByName(treenode\*,string); | 按姓名输出成员信息 |
| QString SearchBirthday(treenode\*, Date,bool&); | 查找某日期的信息 |
| QString PrintMessageByBirthday(treenode\*,Date); | 按出生日期输出成员信息 |
| treenode\* findPersonByName(treenode\*, string); | 在家谱树中，寻找一个指定的人名，并返回其指针 |
| voidReturnTreenodeByName(treenode\*&, string,treenode\*&); | 先序遍历 遍历整棵树 直到 找到目标节点 主要用于后面几个函数的实现 |
| QStringJudgeRelationship(treenode\*,QString,QString); | 判断两人关系 |
| voidReturnTreenodeByNameAndBirthday(treenode\*&, string, treenode\*&); | 按照姓名返回根节点 在函数里实现一个遍历 |
| QString InsertChild(treenode\*&,string,PersonMessage); | 插入一个孩子 |
| QString ModifyMember(treenode\*&,string,PersonMessage); | 修改某成员信息 前提是某一成员要先存在 |
| void SaveFile(treenode\*, fstream&); | 保存文件 |

表4-1程序相关函数

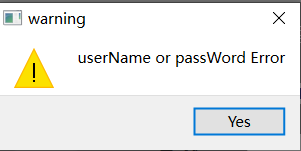
|  |  |
| --- | --- |
| 包含库 | 在本程序中作用 |
| #include <QWidget> | qt官方库，用来实现窗口可视化界面的一系列操作，包括各个功能按钮，窗体的样式表设计等 |
| #include <QInputDialog> | 输入对话框 用来弹出提示信息窗口 |
| #include <string> | C++字符串相关操作 |
| #include <fstream> | 文件流 用来操作文件 |
| #include <sstream> | <sstream> 定义了三个类：istringstream、ostringstream 和 stringstream，分别用来进行流的输入、输出和输入输出操作；同时可以用来进行类型转换 |
| #include <QString> | qt 中字符串相关函数 |
| #include <QTextStream> | QTextStream 可以逐行读取 方便处理数据 |
| #include <QDebug> | qt命令行里的输入输出，便于调试 |
| #include<QVBoxLayout> | qt的布局相关库 |

表4-2程序用到的库

## 运行结果

（1）进入系统首先是登陆界面，登录界面实现了输入用户名和密码（密码支持隐藏），在密码输入错误时会报错，只有用户名和密码输入正确后才能登录，如下图所示：

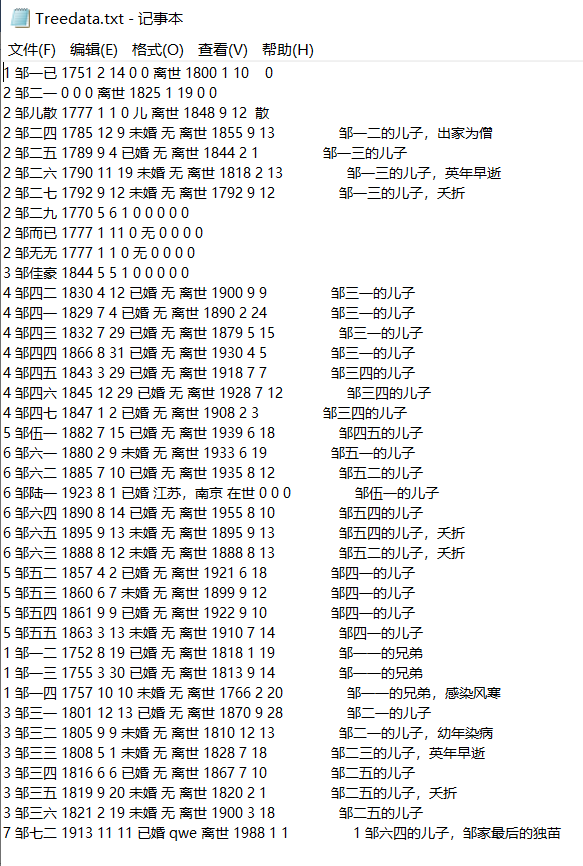
密码输入错误时报错，关闭报错后可重新输入：



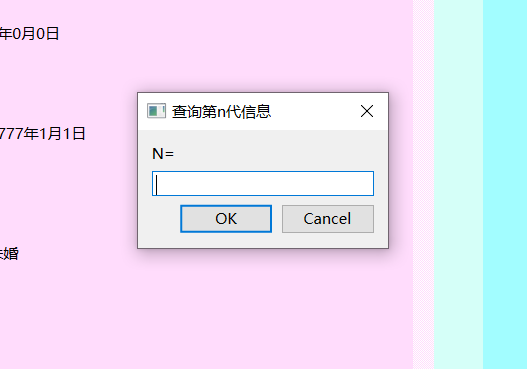
（2）登陆成功后进入程序主界面;程序的主要功能的实现就在这个界面，包括查询第n代人信息、按照姓名查询成员信息、按照出生日期查询成员信息、查询两人关系、为某成员添加孩子、修改某成员信息、输出所有成员的信息、返回和退出。如下图所示：

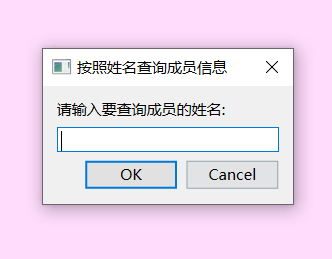


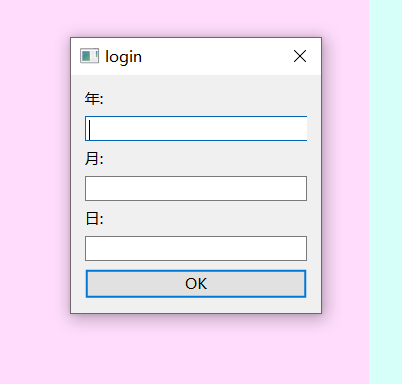
对应的txt后缀文件：

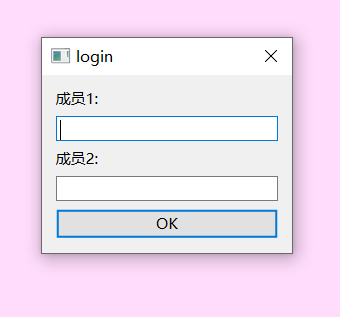


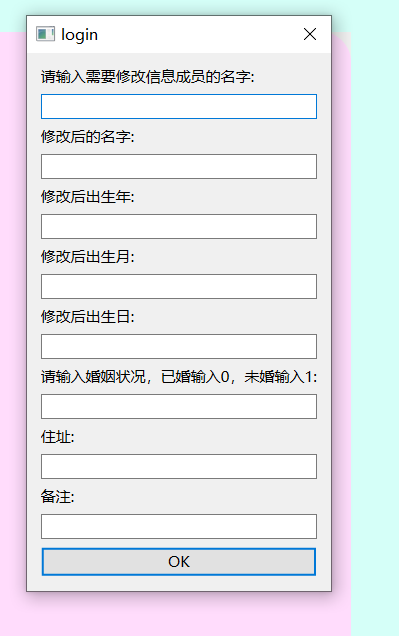
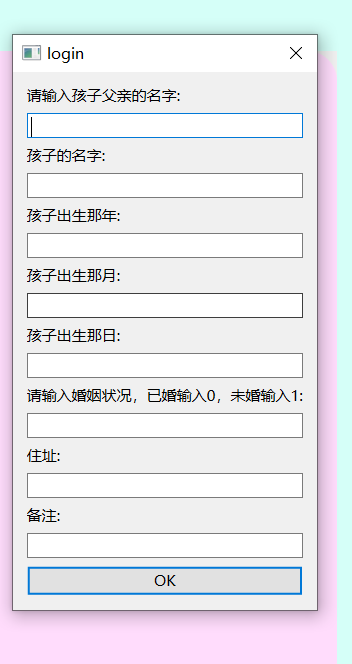
对应的各个按钮实现如下各图所示：

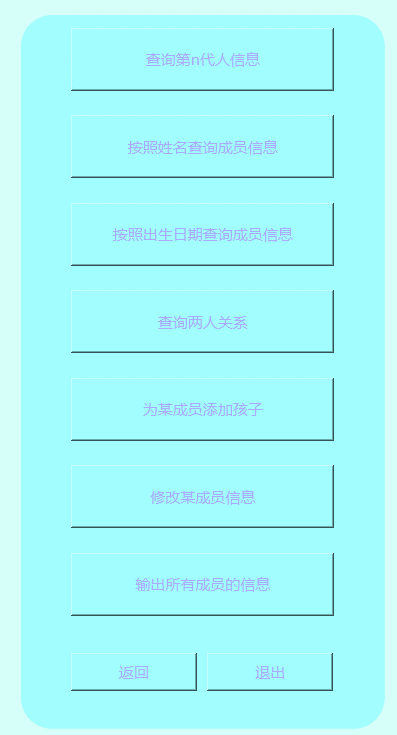












## 总结

**6.1 数据结构与算法实验课-综合应用支撑的毕业要求指标点**

数据结构与算法实验课-综合应用支撑以下四项毕业要求指标点：

指标点2-4：能够从工程科学的角度，结合文献查阅及研究，对计算机领域复杂工程问题进行系统分析，并获得有效结论。

指标点7-1：知晓和理解环境保护和可持续发展的理念和内涵，正确认识计算机科学技术的发展与环境和可持续发展的关系。

指标点9-3：能够理解个人在团队中的角色划分，且胜任相应的角色职责。

指标点12-1：能在社会发展的大背景下，理解终身学习的必要性，具有自主学习和终身学习的意识。

**6.2心得**

赵佳豪：实践是检验真理的唯一标准，两年前面向对象我用的qt5，做的是一个很简陋的音乐播放器，而这个程序我用的QT6,当兵两年，计算机技术进步很快，两年前GO，vue，这些我听都没听过，而现在这些都已经成为新时代下企业招聘的考量因素。这两年AI的发展迅猛，甚至有时候也会让我对未来产生迷茫，让我产生“上这大学有啥用”的念头，不过学习一些底层的东西总是好的，只有学了数据结构和算法，未来工作时才能有思路去优化一些框架，减少程序的运行时间，数据结构和算法是基础，不论学习什么语言，这都是绕不过的一关，与其痛苦挣扎，不如痛痛快快的投入时间和精力去研究。以下是本课设期间，我的主要参考：

QT官方帮助文档

慕晨 安毅生主编 《数据结构与算法C++实现》

严蔚敏.《数据结构（C语言版）》

程杰 《大话数据结构》

csdn

GitHub

曹默涵：基于二叉树的家谱管理系统的开发过程中，我深刻认识到了二叉树在数据结构中的重要性和应用价值。通过合理地利用二叉树的特点和相关算法，我能够更高效地处理家族成员的信息和关系，并提供快速的查询和展示功能。这个项目不仅加深了我对二叉树的理解，也提升了我的数据结构和算法设计能力。

参考文献：

慕晨 安毅生主编 《数据结构与算法C++实现》

CSDN

徐晶：在该课设的设计过程中，我通过翻阅资料、自己实践的过程更深刻地理解了数据结构中的树这一部分的内容。例如，在利用二叉树算法实现家族信息查询功能的过程中，对树的遍历和构造有了更清楚的认知。提高了自己的动手编程实现能力，加深了对数据结构的理论知识理解。

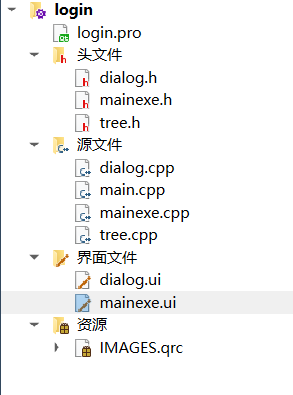
参考文献：

慕晨 安毅生主编 《数据结构与算法C++实现》

CSDN

## 程序源代码

本程序包含的文件如下：



源代码已上传：