家谱管理系统

#### 软件工程20级2班 2053300 胡锦晖

# 项目概述

## 1.1 项目背景

家谱是一种以表谱形式，记载一个以血缘关系为主体的家族世袭繁衍和重要任务事迹的 特殊图书体裁。家谱是中国特有的文化遗产，是中华民族的三大文献(国史，地志，族谱) 之一，属于珍贵的人文资料，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。

## 1.2 项目目标

本项目的目标是通过树形结构实现对一个家谱管理的简单模拟，主要功能有：

1. 建立家谱
2. 完善家谱
3. 添加家庭成员
4. 解散局部家庭
5. 更改家庭成员姓名

对于每个功能的具体操作将在【2.3功能实现】模块进行介绍。

# 实现思路

## 2.1 数据的存储结构

对于家谱来说，如果将每一位家庭成员抽象为一个结点，那么家谱描述的其实是一个结点与另一个或多个结点之间的关系，而树这种数据结构恰好可以方便地描述这种结点关系。

而对于树的多种存储结构，考虑到孩子兄弟表示法这种将一般树转化为二叉树的存储结构可以方便地实现查询、遍历等操作，最终的树结点设计如下：

需要注意的是文本

低可信度描述已自动生成，这里的树结点使用了模板Type，在使用时需要用string进行具体化。

## 文本 描述已自动生成2.2 类的设计及关系

## 2.3 功能实现

### 2.3.1 菜单

文本

描述已自动生成由于每一个家谱均为一个单例，所以我们考虑将菜单显示在FamilyTree的构造函数中实现，同时进行数据的初始化，具体如下：

### 2.3.2 Tree类中各个方法的实现

由于使用的是孩子兄弟表示法，可以对于这样一个二叉链表的结构，可以采用递归的方式轻易实现查找和删除操作。

1. **文本

   描述已自动生成查找操作**：找到以pNode为根结点的树中值为key的结点并返回指向该结点的指针，若未找到则返回nullptr。
2. **插入操作**：给parent结点添加一个值为key的孩子。插入操作不需递归：当parent无孩子时，直接插入到firstChild中去；当parent有孩子时，遍历nextSibling直到文本

   描述已自动生成nextSibling为空（此时相当于遍历到了parent的最后一个孩子）时进行插入。
3. 文本

   描述已自动生成**删除操作**：删除操作借助了递归实现，具体思路为：若该结点没有孩子，则直接删除该结点，否则递归删除以该结点首孩子为根结点的子树。

### 2.3.3 完善家谱

完善家谱与下面【2.3.4添加家庭成员】功能十分相似，区别在于，完善家谱是对于一个已经存在于家谱中的家庭成员进行信息的录入，支持一次性录入多个子女的信息，而添加家庭成员只是对于一个已经存在于家谱中的家庭成员进行单个子女的添加。

### 2.3.4 添加家庭成员和解散家庭

这两个功能的主体是【2.3.2 Tree类各个方法的实现】中的insert方法和remove方法，在其基础上添加了一些对于输入合法性的判断以及输出的提示性语句，如图所示：

### 文本 描述已自动生成2.3.5 更改家庭成员的姓名

该功能的主体是【2.3.2 Tree类各个方法的实现】中的find方法，在找到需要修改名称的成员结点之后，根据新输入的信息对其进行相应修改即可：

### 文本 描述已自动生成2.3.6 格式化输出一位成员的家庭

文本

描述已自动生成该功能主要是要求查找一个结点的孩子。由于我们使用孩子兄弟法模拟一棵普通的树，一个结点的孩子即为它的长子firstChild以及firstChild的所有兄弟nextSilbing，循环寻找右兄弟nextSibling直到其为空。

### 2.3.7 错误处理

本项目的错误处理主要是操作码输入错误以及重名/不存在错误两部分。

重名/不存在错误处理策略：

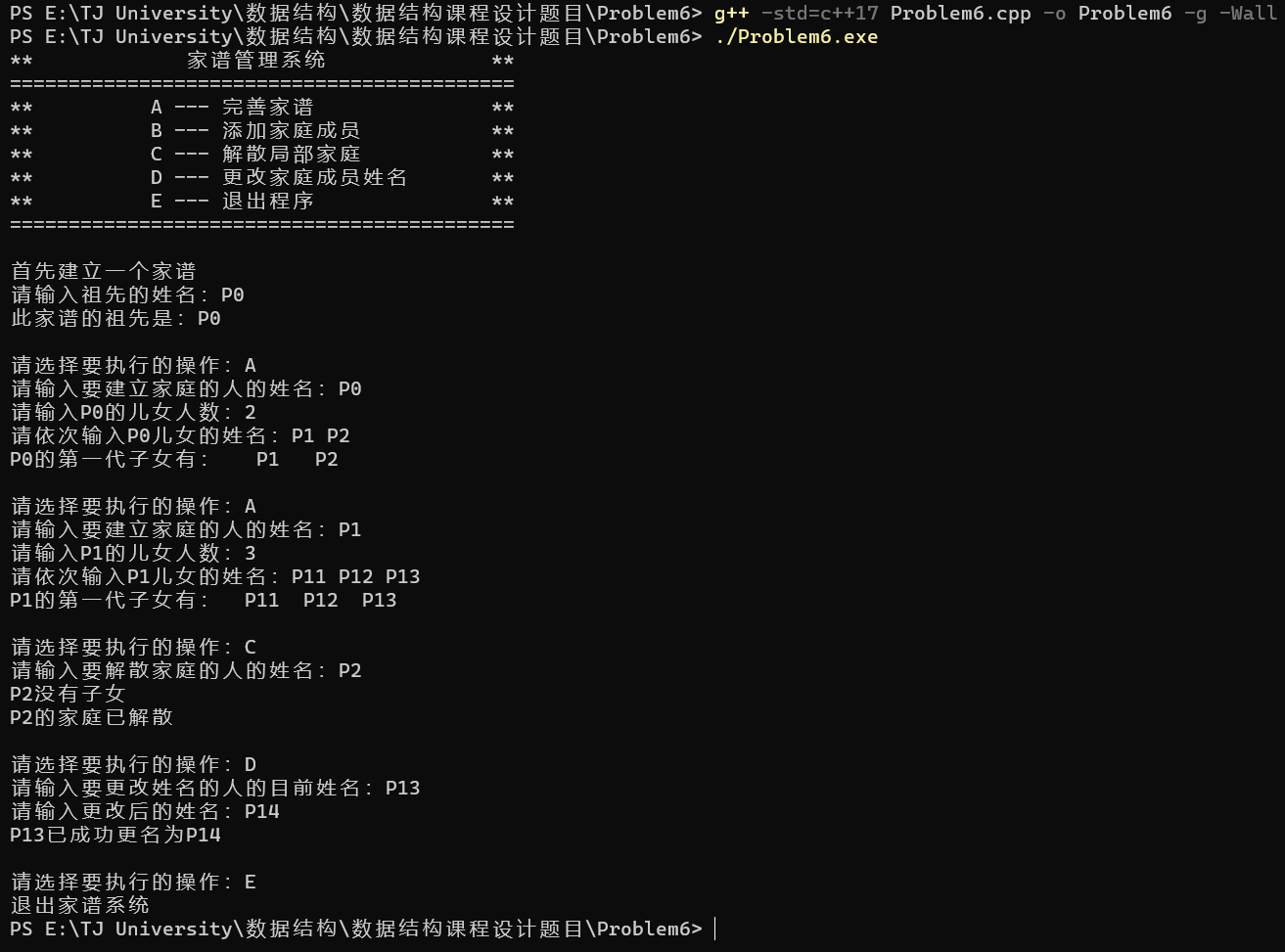
根据【2.3.2 Tree类各个方法的实现】中的find方法进行查找检验，根据返回值的不同判断是否重名/不存在。

操作码输入错误处理策略：

 只有输入为0-5的有效操作才可进行后续操作，否则将会清空缓冲区并要求重新输入。

# 四、测试结果

## 4.1 Windows平台

操作系统：Windows11 专业版 21H2

编译器：g++ (GCC) 8.1.0

C++标准：c++17

## 4.2 Linux平台

操作系统：CentOS Linux release 8.5.2111

编译器：g++ (GCC) 8.5.0-4

C++标准：c++17

## 4.3 Mac平台

操作系统：macOS Monterey 12.0.1

编译器：clang++ 13.0.0

C++标准：std=c++17

图形用户界面, 文本

描述已自动生成