

# 吉林大学

## 软件学院

### 《C++课程设计》

### 实验报告

|      |          |     |  |
|------|----------|-----|--|
| 实验地点 | A108     | 机器号 |  |
| 班级   | 551801班  |     |  |
| 学号   | 55180123 |     |  |
| 姓名   | 荆煜       |     |  |

|   |            |
|---|------------|
| 实验项目  | 模拟即时通信系统实现 |
| <p>一、设计任务分析</p> <p>设计即时通信系统，其中包含三种账户类型，同一用户可使用三个不同账户。每个账户有独立的ID，昵称，住址等信息，但也可相互关联。每种账户都有类似但有些许不同的好友管理系统和群管理系统。可以满足及时网络通信。操作所改变的用户信息可以保存到文件中，再次打开时可以复现用户数据。有一定UI设计。</p>   |            |
| <p>二、设计方案</p> <p>设计关键词：网络通信；容器类；多线程；多态；输入检验；文件管理；</p> <p>主干类继承关系图如下：</p> <pre>classDiagram     class Base_JY     class QQ_JY     class Weibo_JY     class Wechat     class Date_JY     class Friendmanage_JY     class Groupmanage_JY      Base_JY &lt; -- QQ_JY     Base_JY &lt; -- Weibo_JY     Base_JY &lt; -- Wechat     Base_JY -- Date_JY     Base_JY -- Friendmanage_JY     Base_JY -- Groupmanage_JY</pre> <p>Date_JY类存放时间信息，Friendmanage_JY存放好友信息，Groupmanage_JY存放群信息，Base_JY存放三种账号共有的基础信息，QQ_JY，Weibo_JY，Wechat_JY继承Base_JY类。</p> <p>账户间好友关系和群关系实现原理如下：</p> <p>要求新建群和个人用户时，群和个人账户的ID从0开始顺序递增发放。</p> <p>Friendmanage_JY中包含二维容器类vector&lt;vector&lt;int&gt;&gt;friendlist；构成容纳某种程序所有好友关系的二维矩阵。矩阵的横纵坐标均为当前最大ID值。</p> |            |

其中，容器内容默认值为0，若为1则代表单方面发出好友申请。即：  
friendlist[a][b]=1表示id为a的账户向id为b的账户单方面发出好友申请。若  
friendlist[a][b]=1并且friendlist[b][a]=1，则表示a和b成功加为好友。

Groupmanage\_JY中包含二维容器类vector<vector<int>>grouplist；构成容纳该种程序所有群关系的二维矩阵。矩阵第一维表示群ID，第二维表示个人账户ID。矩阵大小为当前最大群ID\*当前最大个人ID。

其中，容器内容默认值为0。假设grouplist[a][b]=0，表示ID为b的个人账户不在ID为a的群众。可以用不同数字表示拥护b在群a中的等级。例如，在该程序的QQ普通群中，令rouplist[a][b]=1表示ID为b的个人账户申请加入ID为a的群，rouplist[a][b]=2表示ID为b的个人账户已在ID为a的群中，rouplist[a][b]=3表示ID为b的个人账户使ID为a的群的管理员，rouplist[a][b]=4表示ID为b的个人账户是ID为a的群的群主。根据数字不同，账户拥有不同权限，可以调用的函数也不同。

Base\_JY类及其成员Date\_JY，Friendmanage\_JY，Groupmanage\_JY类中，部分函数设为虚函数，以便在被QQ\_JY，Weibo\_JY，Wechat\_JY继承时实现多态。

本程序采取端到端的通信形式，未采取服务器-客户端的形式。在头文件”Network\_JY.h”中，编写了基于TCP/IP协议的网络通信功能，不仅支持文本传输，还支持文件传输。

在头文件”Multyple\_thread\_JY.h”和”Threadpool\_JY.h”中，编写了多线程功能，提供了多线程操作接口，以便实践在接入UI图形界面后，实现与菜单操作不冲突的实时接收信息功能。

本程序的所有输入操作均进行了检验操作，增强程序鲁棒性。可以按需求选择输入的范围，并可检验并要求输入是否为string，int，float类型。

本程序的关键在于文件的反复读取与存储。在存放目录下，分别设立QQ，Weibo，Wechat三个文件夹中。因三者较为相似，以QQ为例说明。QQ下属group，person，source三个文件夹。group存放群信息，每个群以相对路径”./QQ/group/group\_id”(id替换为群的id)生成一个txt文件，存放该群的名称，类型等信息。同理，person文件夹下，为每个QQ账户创建一个文件，存放该用户的各种信息。在source文件夹下，有五个文件。qq\_friend.txt存放好友关系；qq\_group\_txt存放群关系；qq\_group\_max.txt存放当前最大群号以便在新建群时发放群号；qq\_id\_max.txt存放当前最大账户ID以便在新建账户时发放ID；qq\_number\_max.txt存放Groupmanage\_JY的辅助量。

### 三、详细设计

本程序采取分文件布局，将分文件分别阐述具体实现。QQ功能最为丰富，因篇幅问题，以QQ为例说明，微博和微信在相应虚函数中删除小幅修改部分功能即可。受篇幅限制，尽量省略代码。

“Limit\_JY.h”负责输入限制。float Limit\_JY::numberlimit(float min, float max, int type); 将值限制在min与max之间，type决定是否限制为整数，并返回符合要求的值。特别的，该程序解决了输入string时隐式转换为int导致蒙混过关或造成程序错误的问题。

“Date\_JY”存放时间类。成员int year;int month;int day;存放时间；  
void set\_system\_time();//读取设置为系统时间； void set\_time();// 手动设置时间；  
void print\_time();//输出时间； int get\_year();//返回年； int get\_month();//返回月  
int get\_day();//返回日；

“Base\_JY”存放基础信息类。成员变量有： Date\_JY creatday;存放创建时间（T龄）； string name;存放昵称； Date\_JY birthday;存放生日； string location;存放地址； int gender;存放性别； bool online;存放是否在线；

void get\_name();//设置昵称  
void get\_gender();//<sup>a</sup>设置性别  
void get\_location();//设置地址  
void get\_birthday();//设置生日  
virtual void show();//展示信息  
void get\_creatday();//自动设置创建日  
virtual void creat() ;//新建账户  
virtual void print\_friend() ;//输出好友  
virtual void read\_friend() ;//读取好友  
virtual void add\_friend() ;//添加好友  
virtual void delete\_friend() ;//删除好友

“Friendmanage\_JY.h”存放好友管理类。有私有成员vector<vector<int>>friendlist;以防好友关系被破坏；公有成员函数  
virtual void read\_friendlist();//读取好友列表

```
virtual void add_friend(long id);//添加好友
virtual void delete_friend(long id);//删除好友
virtual void search_friend(long id);//查找好友
virtual void watch_add(long id);//处理好友申请。
```

特别阐述读取操作函数部分代码：

```
char filename_id[30] = "./QQ/source/qq_id_max.txt";
    long max_id = 0;
    ifstream filein_id(filename_id, ios::in);
    if (!filein_id)
        cout << "error";
    filein_id >> max_id;
    filein_id.close();
```

//文件流读入当前最大账户ID

```
vector<int> temp_vector(max_id+1, 0);
friendlist.resize(max_id+1, temp_vector);
```

//调整容器类大小为ID\*ID的矩阵

添加，删除，申请等操作方法为：遍历筛选friendlist[a][b]，并修改其对应值。原理见第二部分设计方案。

查找操作：先遍历矩阵，找到所有friendlist[a][b]==1&&friendlist[b][a]==1的a和b，即a和b互为好友。按照程序规定的文件命名法生成相对路径名，读取该账户个人信息文件并输出到屏幕上。

```
char filename[30];
sprintf(filename, "./QQ/person/qq_%ld.txt", id);
//生成相对路径名
//在按照文件流正常读入即可
```

“Groupmanage\_JY.h”存储群管理相关操作。私有成员变量long groupid;存放群ID；vector<vector<int> >grouplist;存放群好友列表；公有成员变量int type;存放群类型。

公有成员函数：

```
void apply_zu(long id);//新建讨论组
virtual void apply_group(long id);//新建群
virtual void read_grouplist();//读取群信息
virtual void save_grouplist();//保存群信息
virtual void add_number(long id,long groupid);//添加群成员
```

```
virtual void apply(long id);//申请入群
virtual void delete_number(long id, long groupid);//删除群成员
virtual void search_number(long id, long groupid);//查找群成员
virtual void watch_add(long id, long groupid);//处理申请信息
virtual void set(long id, long groupid);//设置群信息
特别阐述read_group();
```

因为进行群操作时才会读写群信息，不涉及群操作时，群信息不会读写刷新。若新建账号，会导致账户数目增加，导致二维向量的第二维变大。若依然按照最大群ID\*最大账户ID来读取，会导致数据错位。因而设置”./QQ/source/qq\_number\_max.txt”文件，存放刷新前的第二维ID大小。保存同理，通向保存该数值。

查找功能类似好友查找功能，同样按照一定的命名规则读取数据。

删除，申请，处理等功能为简单的便利修改，此处不做赘述。

“QQ\_JY.h”存放QQ账户基础信息。包含私有成员变量： long id;//存放ID  
公有成员变量： Friendmanage\_JY qq\_friendmanage;//存放好友信息；  
Groupmanage\_JY qq\_groupmanage;//存放群信息； int connect\_wechat;是否关联微信； int connect\_weibo;是否关联微博，因微博和QQ共享ID，所以不需单独存储。 long connect\_wechat\_id;关联的微信ID； string key;密码

QQ\_JY(long qid);//构造函数，负责从文件中读取ID为qid的账户信息建立实例对象。

QQ\_JY();//构造函数，负责新建账户并实例化对象。

void creat();//新建账户录入基础信息

long return\_id() {return id;}//返回私有成员

void show\_all();//展示个人信息

特别阐述：新建账号时，读取”./QQ/source/qq\_id\_max.txt”，得到当前最大ID，+1变为本次新建账号的ID，并刷新文件，录入个人信息。

特别的，sprintf(filename, ”./QQ/person/qq\_%ld.txt”, qid);按照该格式，生成新账号的个人信息文件，并将基础信息写入文件。

“Application\_JY.h”用来存放操作界面。有公有成员函数

void how\_sign\_in();//选择登录或新建账户

void login();//登录

```
long creat_app();//新建账户  
void menu_qq(long id);//操作菜单
```

特别的：若在how\_sigh\_in()中选择登录，将会检索该账号是否关联微博和微信账号，过关联，可无需密码一键登录。

在menu\_qq中调用QQ类及基类和类成员的各个函数，已实现不同操作。  
如关联其他账号，新建群，网络通讯聊天等功能。

“Network\_JY.h”包含网络通讯功能。“Mulityple\_thread\_JY.h”负责构建进程条件，“Threadpool\_JY.h”负责构建进程池。因较为复杂，本处不做赘述。

#### 四、总结与体会

本次课程设计，提高了面向对象程序设计能力，自主学习了部分网络通信、多线程、基于Qt图形界面设计的知识。

不足和改进。因wintogo系统崩溃，造成部分文件丢失，使得基于Qt的图形界面无法如期实现。

因时间问题，通信功能没有修改为服务器-客户端模式，必须双方上线才能接受信息。

因无图形界面，在控制台中进行操作多线程意义不大，频繁接收信息可能会导致占用操作菜单，导致无法正常操作。在图形界面中才能体现多线程操作的优势。