目 录

[1 课题分析 1](#_Toc389724324)

[1.1 课题背景 1](#_Toc389724325)

[1.2 功能需求分析 1](#_Toc389724326)

[1.3 数据属性分析 2](#_Toc389724327)

[2 系统设计 3](#_Toc389724328)

[2.1 数据存储设计 3](#_Toc389724329)

[2.2 数据结构设计 3](#_Toc389724330)

[2.3 函数设计与说明 3](#_Toc389724331)

[2.4 主菜单设计 4](#_Toc389724332)

[2.5 文件清单与说明 4](#_Toc389724333)

[3 系统实现 4](#_Toc389724334)

3.1 新建用户登录与找回密码 4

[3.2 新增联系人信息 4](#_Toc389724335)

[3.3 修改联系人信息 5](#_Toc389724336)

[3.4 删除联系人信息 5](#_Toc389724337)

[3.5 联系人分组 5](#_Toc389724338)

[3.6 联系人排序与查询 5](#_Toc389724339)

[4 测试数据与测试过程记录 6](#_Toc389724340)

[4.1 登录界面测试 6](#_Toc389724341)

[4.2 系统功能测试 6](#_Toc389724342)

[5 课程设计遇到的问题及解决方法 6](#_Toc389724343)

[参考文献 7](#_Toc389724345)

# 1 课题分析

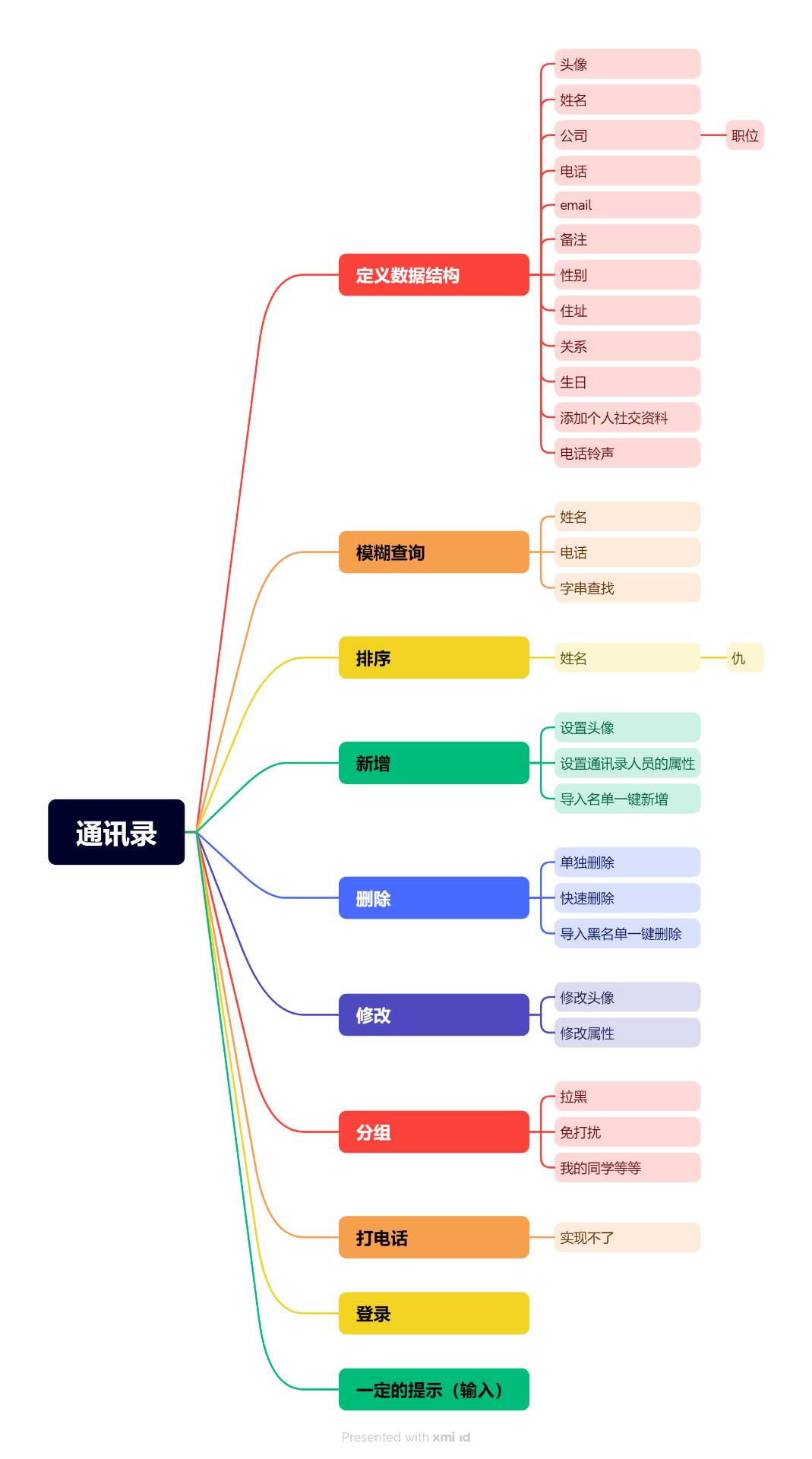
## 1.1 课题背景

本次课程设计我们小组选择设计通讯录，一方面是认为其贴近我们的生活，在生活中时常会使用到，另一方面也是因为我们对这个课题感兴趣，也想尽力设计出一个相对功能完善的通讯录。我们通过使用c、c++、qt来设计，为了实现登录、新建用户、新增联系人、删除、分组等功能，从而设计出一个完善的通讯录。

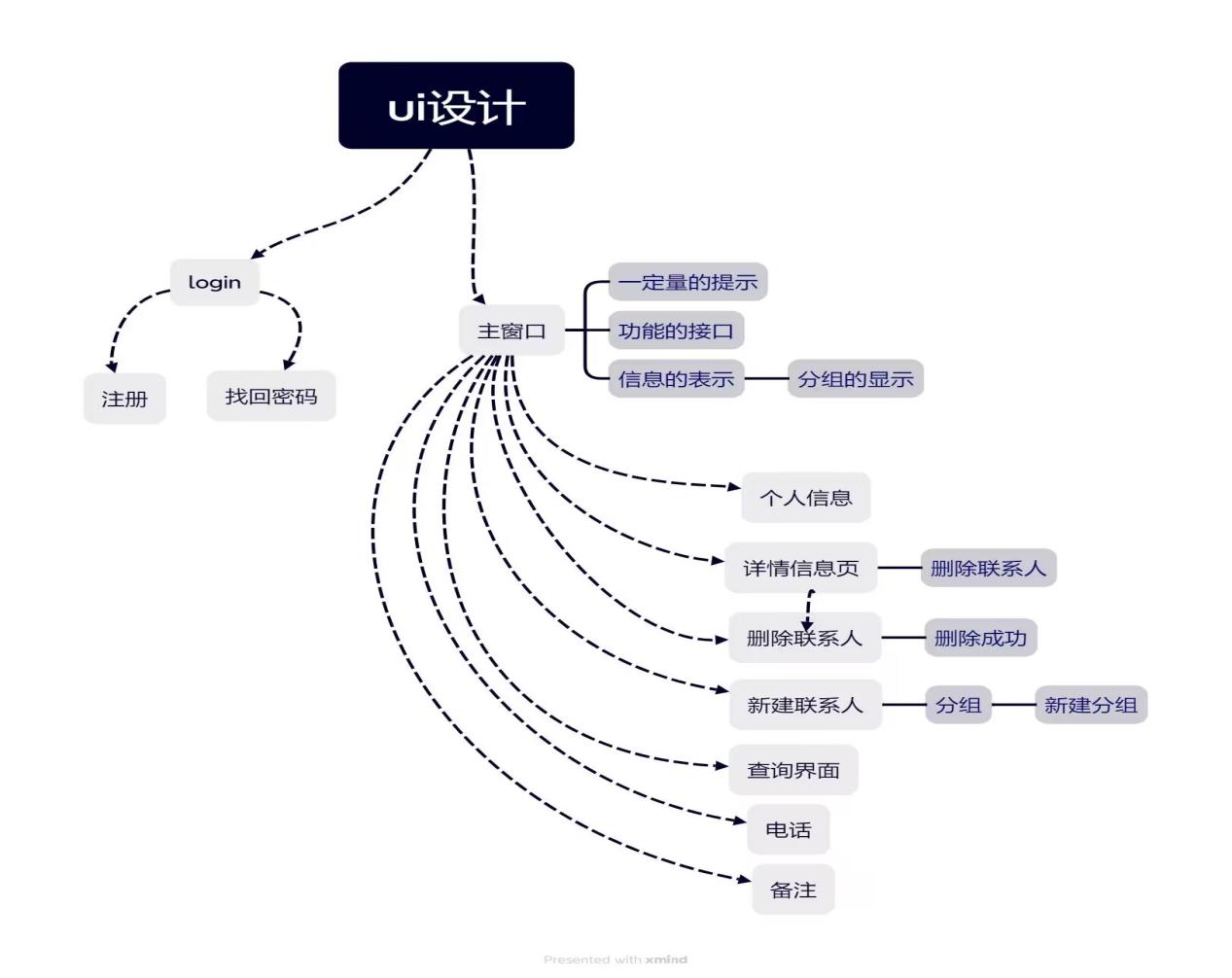
## 1.2 功能需求分析

如下为两个详细设计框架图。

1. 图1为系统功能模块图，我们经过参考以往通讯录的基本功能，有新增一个联系人，删除联系人，更改相关信息，对已有联系人进行排序，以及查询功能，并在此基础上增加了一些个性化的功能，如生日、头像、电话铃声等，为了便于用户自定义设置，为通讯录系统增添色彩。
2. 图2为UI窗口设计图，首先我们设计出了用户登录界面，使程序更具有隐私性，同时也是为了方便管理员对于用户的管理。然后是通讯录主界面窗口及一系列分支功能窗口。



**图1 系统功能模块图**

****

**图2 UI窗口设计图**

（3）下表为本次设计中使用的函数及功能

**表1 函数列表**

|  |  |  |
| --- | --- | --- |
| 序号 | 函 数 原 型 | 功 能 说 明 |
| 1 | Detailpage-on\_pushButton\_4\_clicked | 通过点击删除按钮，对删除目标的子项进行逐行读取确认，最后实现删除 |
| 2 | Detailpage-on\_pushButton\_2\_clicked | 点击分享联系人按钮，逐行读取信息，完成对联系人的分享 |
| 3 | Detailpage-on\_pushButton\_3\_clicked | 点击收藏按钮，逐行读取联系人信息，完成对联系人的收藏 |
| 4 | Newperson-on\_pushButton\_clicked | 完成新建联系人的信息，通过点击保存按钮，完成对联系人的保存 |
| 5 | Owninformation-on\_pushButton\_clicked | 完成对用户信息的完善，通过点击保存按钮，完成对用户信息的保存 |
| 6 | Delete::Delete(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Delete) | 通过管理员账号展示用户的账号信息，完成对用户信息的删除 |
| 7 | Detailpage::Detailpage(string name,string number,string Username,QWidget \*parent)  : QMainWindow(parent),ui(new Ui::Detailpage),conname(name),connumber(number)  , username(Username) | 储存联系人的姓名电话号码等 |
| 8 | Login::Login(QWidget \*parent)  : QMainWindow(parent)  , ui(new Ui::Login) | 完成用户的登录 |
| 9 | Newperson::Newperson(string username,QWidget \*parent)  : QMainWindow(parent),currentname(username)  , ui(new Ui::Newperson) | 完成新建联系人的添加 |
| 10 | Owninformation::Owninformation(string name,QWidget \*parent)  : QMainWindow(parent),username(name)  , ui(new Ui::Owninformation) | 展示用户的个人信息 |
| 11 | Queryinterface::Queryinterface(string name,QWidget \*parent)  : QMainWindow(parent),username(name)  , ui(new Ui::Queryinterface) | 展示出搜索结果 |
| 12 | Retrieverpassword::Retrieverpassword(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Retrieverpassword) | 完成对密码的重新设置 |
| 13 | Signup::Signup(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Signup) | 完成用户账号的创建 |

# 2 系统设计

## 2.1 数据存储设计

用户数据结构设计：采用链式存储方式进行储存，方便增减

联系人数据结构设计：采用线性表进行存储，更加便于查找和排序

## 2.2 函数设计与说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 函数 | 参数 | 返回值 | 功能描述 |
| 按钮相关函数 | | | | |
| 1 | Detailpage-on\_pushButton\_4\_clicked | 无 | 无 | 通过点击删除按钮，对删除目标的子项进行逐行读取确认，最后实现删除 |
| 2 | Detailpage-on\_pushButton\_2\_clicked | 无 | 无 | 点击分享联系人按钮，逐行读取信息，完成对联系人的分享 |
| 3 | Detailpage-on\_pushButton\_3\_clicked | 无 | 无 | 点击收藏按钮，逐行读取联系人信息，完成对联系人的收藏 |
| 4 | Newperson-on\_pushButton\_clicked | 无 | 无 | 完成新建联系人的信息，通过点击保存按钮，完成对联系人的保存 |
| 5 | Owninformation-on\_pushButton\_clicked | 无 | 无 | 完成对用户信息的完善，通过点击保存按钮，完成对用户信息的保存 |
| 序号 | 函数 | 参数 | 返回值 | 功能描述 |
| 构造函数 | | | | |
| 1 | Delete::Delete(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Delete) | Delete的父类 | Delete | 通过管理员账号展示用户的账号信息，完成对用户信息的删除 |
| 2 | Detailpage::Detailpage(string name,string number,string Username,QWidget \*parent)  : QMainWindow(parent),ui(new Ui::Detailpage),conname(name),connumber(number)  , username(Username) | name,number,Username,父类 | Detailpage | 储存联系人的姓名电话号码等 |
| 3 | Login::Login(QWidget \*parent)  : QMainWindow(parent)  , ui(new Ui::Login) | 父类 | Login | 完成用户的登录 |
| 4 | Newperson::Newperson(string username,QWidget \*parent)  : QMainWindow(parent),currentname(username)  , ui(new Ui::Newperson) | Username，父类 | Newperson | 完成新建联系人的添加 |
| 5 | Owninformation::Owninformation(string name,QWidget \*parent)  : QMainWindow(parent),username(name)  , ui(new Ui::Owninformation) | name，父类 | Owninformation | 展示用户的个人信息 |
| 6 | Queryinterface::Queryinterface(string name,QWidget \*parent)  : QMainWindow(parent),username(name)  , ui(new Ui::Queryinterface) | name，父类 | Queryinterface | 展示出搜索结果 |
| 7 | Retrieverpassword::Retrieverpassword(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Retrieverpassword) | 父类 | Retrieverpassword | 完成对密码的重新设置 |
| 8 | Signup::Signup(QWidget \*parent)  : QWidget(parent)  , ui(new Ui::Signup) | 父类 | Signup | 完成用户账号的创建 |

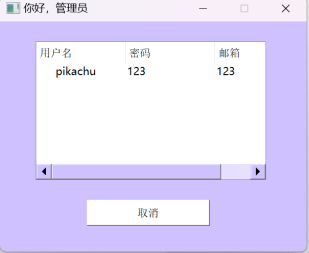
## 

## 2.4 菜单设计

2.4.1登录界面：程序一开始展现给用户的界面，提供了忘记密码，创建用户登录和退出以及眼睛（用来转变密码格式）的几个按钮，以及两个输入文本框用以接受用户输入的数据

## 

2.4.2管理员界面：当文本框接收到来自管理员的身份验证的时候，跳转至管理员界面。管理员界面是给管理员进行用户的管理的界面，可以用以查找当前用户，也可以删除当前用户。

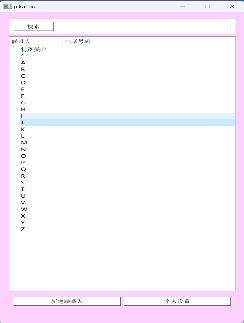


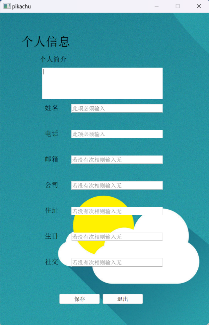
2.4.3注册界面：当用户在登陆界面按下创建用户界面的时候，跳转至此界面，此界面可以完成用户的新建操作，两次输入密码保证了用户密码的正确性，确定创建用户后，在文件中写入用户的信息。

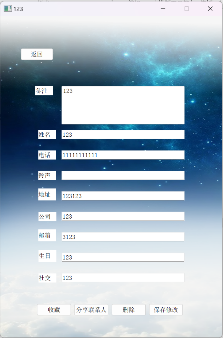


2.4.4忘记密码界面：当按下忘记密码按钮的时候跳转至此界面，增加了两个数学式子用于验证是否是用户要修改自己的密码，两个确认密码用于确认用户输入的新密码。



2.4.5主界面：当用户在登陆界面输入相应的信息后，跳转至此界面。此界面提供了搜索，个人设置，新建联系人，双击联系人可以跳转至联系人详情页的几个按钮，a-z index直观的展示每个联系人的拼音或者英文首字母位置，为用户检索提供了便捷，我们也实现了中英文混合排序，更加方便了用户的检索。  
2.4.6个人信息界面：个人信息界面，用于用户存储自己的个人信息，在之后如果用户想要添加联系人，可以展示给其他人看，方便用户添加联系人。

2.4.7联系人详情页界面：联系人详情页界面，提供了几个文本框用于展示联系人的详情信息，提供了收藏按钮用于用户对联系人进行分组，分享联系人可以导出联系人的信息至文件中，之后用户可以发送给其他人，提供了删除按钮用于删除联系人，最后是保存修改按钮，用于保存用户对于联系人信息的修改。

2.4.8新建联系人界面：在用户按下新建联系人按钮的时候跳转至新建联系人界面，几个有关联系人信息的文本框用户获取联系人的信息，并且在按下保存按钮之后查看电话号码的格式是否正确，再决定是否保存数据。

2.4.9搜索界面：当用户在搜索框中输入想要查询的字段，并且按下搜索按钮之后，会跳转至查询界面，查询的结果主要是展现联系人以及电话号两个信息，方便用户挑选用户需要的联系人，双击联系人信息，依然可以跳转至联系人详情页查看联系人，或者对联系人进行修改。

## 2.5 文件清单与说明

cnl.txt ：存储拼音的文件，用于中文字符串转换拼音字符串。

User.bat :储存用户信息文件，为保证安全性，采用bat

Usernew.bat ：储存新修改的用户信息，为保证安全，采用bat

connectsname.txt ：储存联系人信息文件，为方便读取，采用txt

# 3 系统实现

## 3.1 按钮功能实现

3.1.1删除联系人按钮

void Detailpage::on\_pushButton\_4\_clicked()//删除联系人功能实现

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

## 

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","删除成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

点击删除按钮之后通过对联系人信息的逐行读取，比对联系人的备注，姓名，电话号码，邮箱，公司，住址，生日，社交等信息，比对成功之后完成对联系人的删除，提示删除成功。自动关闭该界面，返回原界面。

3.1.2分享联系人按钮void Detailpage::on\_pushButton\_2\_clicked()

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

//cout<<currname<<endl<<currtext;

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

在联系人详情页界面，通过点击分享联系人按钮，通过对联系人信息读取，确认联系人之后，完成对联系人的分享。后提示分享成功。随后自动关闭该界面，返回原界面。

3.1.3收藏按钮实现

void Detailpage::on\_pushButton\_3\_clicked()

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

//cout<<currname<<endl<<currtext;

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

## 在联系人详情页界面，通过点击收藏联系人按钮，读取该联系人信息，确认联系人后完成联系人的收藏，提示收藏成功。将放于收藏子节点，随后自动关闭

3.1.4新建联系人按钮void Newperson::on\_pushButton\_clicked()

{

QString Usertext=ui->textEdit->toPlainText();

QString Name=ui->lineEdit\_3->text();

QString Number=ui->lineEdit\_4->text();

QString Email=ui->lineEdit\_5->text();

QString Company=ui->lineEdit\_6->text();

QString Address=ui->lineEdit\_7->text();

QString Birth=ui->lineEdit\_8->text();

QString Social=ui->lineEdit\_9->text();

string usertext=Usertext.toStdString();

string name=Name.toStdString();

string number=Number.toStdString();

string email=Email.toStdString();

string company=Company.toStdString();

string address=Address.toStdString();

string birth=Birth.toStdString();

string social=Social.toStdString();

ofstream out;

string file=currentname+"connecters.bat";

out.open(file,ios::app);//定位到文件的末尾

out<<usertext<<"\n"<<name<<"\n"<<number<<"\n"<<email<<"\n"<<company<<"\n"<<address<<"\n"<<birth<<"\n"<<social<<"\n";

out.close();

QMessageBox::information(this,"提示","新建联系人成功");

this->close();

MainWindow \*main=new MainWindow(currentname);

main->ini();

main->show();

}

通过点击主界面的新建联系人按钮后，自动跳转新建联系人页面，完成对新建联系人的信息输入后点击保存按钮，完成联系人新建，随后返回主界面。

3.1.5用户信息保存按钮

void Owninformation::on\_pushButton\_clicked()

{

if(Newperson::checknumber(ui->lineEdit\_number->text().toStdString()))

{

string file=username+"owninformation.bat";

string file\_new=username+"owninformationnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式不正确");

}

}

{

QMessageBox::information(this,"提示","电话号格式不正确");

}

}

用户完成个人信息界面信息的修改，通过点击保存按钮，完成对信息修改的保存，随后提示修改成功。

## 3.2 构造函数

3.2.1删除界面

Delete::Delete(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Delete)

{

ui->setupUi(this);

ui->treeWidget->setHeaderLabels(QStringList()<<"用户名"<<"密码"<<"邮箱");

ifstream in;

in.open("User.bat",ios::in);

string name,password,email;

QString now;

while(getline(in,name))

{

getline(in,password);

getline(in,email);

QTreeWidgetItem \* item = new QTreeWidgetItem(QStringList()<<now.fromStdString(name)<<now.fromStdString(password)<<now.fromStdString(email));

ui->treeWidget->addTopLevelItem(item);

}

in.close();

connect(ui->treeWidget, &QTreeWidget::itemDoubleClicked, this, &Delete::on\_treeWidget\_itemDoubleClick);

}

通过该函数返回Delete实例，设置一个删除界面UI，表头中展示用户名，密码，邮箱。随后打开储存用户信息的文件，显示出需要删除的用户信息，关闭该文件。通过绑定的删除按钮槽函数完成对联系人的删除。

3.2.2详情页界面

Detailpage::Detailpage(string name,string number,string Username,QWidget \*parent)

: QMainWindow(parent),conname(name),connumber(number),username(Username)

, ui(new Ui::Detailpage)

{

ui->setupUi(this);

QString name\_ccc=name\_ccc.fromStdString(conname);

this->setWindowTitle(name\_ccc);

string file=username+"connecters.bat";

ifstream in;

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

//cout<<currname<<endl<<currtext;

if(currname==conname&&currnumber==connumber)

{

ui->textEdit->setPlainText(Currtext.fromStdString(currtext));

ui->lineEdit\_address->setText(Currtext.fromStdString(curraddress));

ui->lineEdit\_birth->setText(Currtext.fromStdString(currbirth));

ui->lineEdit\_company->setText(Currtext.fromStdString(currcompany));

ui->lineEdit\_email->setText(Currtext.fromStdString(curremail));

ui->lineEdit\_name->setText(Currtext.fromStdString(currname));

ui->lineEdit\_number->setText(Currtext.fromStdString(currnumber));

ui->lineEdit\_social->setText(Currtext.fromStdString(currsocial));

}

}

}

## 通过该函数返回一个Detailpage实例，存储联系人的姓名，电话号码，用户名。设置一个详情页的UI界面，将信息转化为String类型进行处理。打开用户的联系人文件，储存联系人的备注，姓名，电话号码，邮箱，公司，住址，生日，社交等信息。随后进行逐行读取，直到结尾或相等。通过UI界面的Edit来写入联系人信息。

3.2.3登录界面

Login::Login(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::Login)

{

ui->setupUi(this

ui->lineEdit\_2->setPlaceholderText("请输入用户名");

ui->lineEdit->setPlaceholderText("请输入密码");

ui->lineEdit->setEchoMode(QLineEdit::Password); ui->toolButton->setIcon(QIcon(":\eyesopen.png"));

}

## 通过该函数返回一个Login实例。设置一个登录UI界面，分别在UI中的各个Edit中输入用户名和密码，点击显示密码图标实现密码可视化。

3.2.4新建联系人界面

Newperson::Newperson(string username,QWidget \*parent)

: QMainWindow(parent),currentname(username)

, ui(new Ui::Newperson)

{

ui->setupUi(this);

}

设置一个新建联系人的UI界面，通过该函数返回Newperson实例。

3.2.5个人信息界面

Owninformation::Owninformation(string name,QWidget \*parent)

: QMainWindow(parent),username(name)

, ui(new Ui::Owninformation)

{

ui->setupUi(this);

QString Username\_c=Username\_c.fromStdString(username);

this->setWindowTitle(Username\_c);

QString Username=Username.fromStdString(username);

Owninformation::setToolTip(Username);

ifstream in;

string file=username+"owninformation.bat";

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

ui->textEdit->setPlainText(Currtext.fromStdString(currtext));

ui->lineEdit\_address->setText(Currtext.fromStdString(curraddress));

ui->lineEdit\_birth->setText(Currtext.fromStdString(currbirth));

ui->lineEdit\_company->setText(Currtext.fromStdString(currcompany));

ui->lineEdit\_email->setText(Currtext.fromStdString(curremail));

ui->lineEdit\_name->setText(Currtext.fromStdString(currname));

ui->lineEdit\_number->setText(Currtext.fromStdString(currnumber));

ui->lineEdit\_social->setText(Currtext.fromStdString(currsocial));

}

}

创建一个个人信息UI界面，通过该函数返回Owninformation实例。设置该UI界面，将信息转化为String类型进行处理。打开储存用户信息的文件，通过UI界面中的Edit来输入用户个人信息，进行逐行读取，直到结尾或相等，将其储存起来。

3.2.6搜索界面

Queryinterface::Queryinterface(string name,QWidget \*parent)

: QMainWindow(parent),username(name)

, ui(new Ui::Queryinterface)

{

ui->setupUi(this);

connect(ui->treeWidget, &QTreeWidget::itemDoubleClicked, this, &Queryinterface::on\_treeWidget\_itemDoubleClicked);

}

设置一个搜索界面的UI，通过该函数返回Queryinterface实例。在界面的treeWidget中显示搜索结果，通过槽函数完成搜索。

3.2.7忘记密码界面

Retrieverpassword::Retrieverpassword(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Retrieverpassword)

{

ui->setupUi(this);

}

创建一个忘记密码UI界面，通过该函数返回一个Retrieverpassword实例。

3.2.8注册界面

Signup::Signup(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Signup)

{

ui->setupUi(this);

}

创建一个注册UI界面，通过该函数返回一个Signup实例。

# 4 测试数据与测试过程记录

## 4.1 菜单测试

4.1.1 主界面

## 4.1.2用户详情页



## 4.1.3联系人详情页

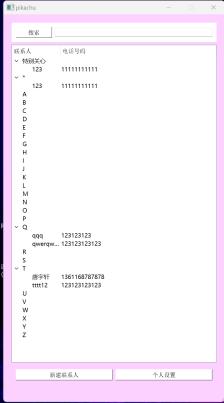
## 4.1.4新建联系人



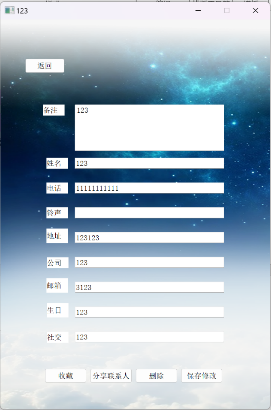
## 4.2 系统功能测试

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 功能名称 | 数据测试 | 预期结果 | 测试结果 | 测试结论 |
| 1 | 主界面 | 点击按钮，双击联系人 | 能够成功跳转至新建联系人或联系人详情页 | 成功跳转至新建联系人界面以及联系人详情页界面 | 通过 |
| 2 | 联系人详情页 | 显示联系人信息，修改联系人信息，按钮 | 显示联系人信息，可以成功修改信息，实现联系人收藏及分享 | 成功修改信息，提示分享成功，收藏成功 | 通过 |
| 3 | 用户详情页 | 显示用户信息，修改用户信息 | 显示所有用户信息，可以修改信息 | 提示修改成功，并且信息均为修改之后 | 通过 |
| 4 | 新建联系人 | 新建一个联系人 | 成功添加提个联系人 | 成功加入新联系人 | 通过 |

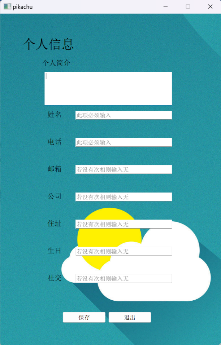
4.2.1主角面功能



4.2.2联系人详情页功能



4.2.3用户详情页功能

4.2.4新建联系人功能

# 5 课程设计遇到的问题及解决方法

（1）

问题内容：公开方法时代码报错

问题分析：属性有误

解决方法：修改函数，将属性公开。

（2）

问题内容：进行UI界面美化时，无法美化

问题分析：子类会继承父类属性，包括美化图片

解决方法：对父类添加#，使子类无法继承父类

问题内容：用户关闭软件后造成信息丢失

问题分析：没有很好的保存用户信息，用户数据导致信息丢失

解决方法：创建一个专门的文件，用于储存用户信息

（4）

问题内容：调用函数使报错

问题分析：C语言部分函数我们无法进行匹配

解决方法：对数据进行转型

# 参考文献

1. 算法竞赛入门经典/刘汝佳，北京：清华大学出版社2009.11
2. 数据结构：c语言/严蔚敏，吴伟民，北京：清华大学出版社1997.4
3. C++ Primer Plus中文版：第六版/（美）史蒂芬-普拉达；张海龙，袁国忠译北京：人民邮电出版社，2020.7
4. C程序设计教程与实验/吉顺如，北京：清华大学出版社，2022.1
5. Qt5.12实战 朱晨冰 / 李建英 北京：清华大学出版社

# 附：程序代码

头文件 cnlhelper.h

#ifndef CNLHELPER\_H

#define CNLHELPER\_H

#include <QObject>

class CNLHelper : public QObject

{

Q\_OBJECT

public:

explicit CNLHelper(QObject \*parent = nullptr);

private:

static CNLHelper \*curHelper;

QStringList listPY;

QStringList listJP;

public:

QString CNToPY(const QString &cnstr);//汉语转拼音

QString CNToEL(const QString &cnstr);//汉语转所有字首字母

QString CNToFL(const QString &cnstr);//汉语转第一个汉字首字母

};

#endif // CNLHELPER\_H

头文件 delete.h

#ifndef DELETE\_H

#define DELETE\_H

#include <QWidget>

#include <QTreeWidgetItem>

#include "Login.h"

namespace Ui {

class Delete;

}

class Delete : public QWidget

{

Q\_OBJECT

public:

explicit Delete(QWidget \*parent = nullptr);

~Delete();

private slots:

void on\_pushButton\_2\_clicked();

void on\_treeWidget\_itemDoubleClick(QTreeWidgetItem \*item, int column);

void removeItem(QTreeWidgetItem \*item);

private:

Ui::Delete \*ui;

};

void del(string name,string password,string email);

void ini(QTreeWidgetItem \*item);

#endif // DELETE\_H

头文件 detailpage.h

#ifndef DETAILPAGE\_H

#define DETAILPAGE\_H

#include <QMainWindow>

#pragma execution\_character\_set("UTF-8")//设置ui界面文字为中文

using namespace std;

namespace Ui {

class Detailpage;

}

class Detailpage : public QMainWindow

{

Q\_OBJECT

public:

Detailpage(string conname, string connumber,string Username,QWidget \*parent = nullptr);

~Detailpage();

private slots:

void on\_pushButton\_clicked();

void on\_pushButton\_4\_clicked();

void on\_pushButton\_6\_clicked();

void on\_pushButton\_2\_clicked();

void on\_pushButton\_3\_clicked();

private:

Ui::Detailpage \*ui;

string conname;

string connumber;

string username;

};

#endif // DETAILPAGE\_H

头文件 Login.h

#ifndef LOGIN\_H

#define LOGIN\_H

#include <QMainWindow>

#include<qstring.h>

#include<iostream>

#include <QMessageBox>

#include <cstring>

#include <string.h>

#include <fstream>

#include <cstring>

#include <qmainwindow.h>

#include "mainwindow.h"

#include "retrieverpassword.h"

#include <sstream>

using namespace std;

#pragma execution\_character\_set("UTF-8")//设置ui界面文字为中文

void getusers();

QT\_BEGIN\_NAMESPACE

namespace Ui {

class Login;

}

QT\_END\_NAMESPACE

class Login : public QMainWindow

{

Q\_OBJECT

public:

Login(QWidget \*parent = nullptr);

~Login();

Retrieverpassword retrie;

private slots:

void on\_pushButton\_2\_clicked();

void on\_pushButton\_clicked();

void on\_toolButton\_clicked();

void on\_pushButton\_3\_clicked();

void on\_pushButton\_4\_clicked();

private:

Ui::Login \*ui;

};

#endif // LOGIN\_H

头文件 mainwindow.h

#ifndef Z\_H

#define Z\_H

#include <QMainWindow>

#include <QTreeWidgetItem>

#include <stdio.h>

#include <cstdio>

#include <stdlib.h>

#include <stdint.h>

#include <stdio.h>

#include <ctype.h>

#include <string.h>

#include <QDebug>

#include <QCollator>

#include <QLocale>

using namespace std;

#pragma execution\_character\_set("GB2312")//设置ui界面文字为中文

QT\_BEGIN\_NAMESPACE

namespace Ui {

class MainWindow;

}

QT\_END\_NAMESPACE

typedef struct connecters

{

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

string check="";

struct connecters \*next;

}\*Connecters,connecters;

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

MainWindow(string name,QWidget \*parent = nullptr);

~MainWindow();

void on\_treeWidget\_itemDoubleClicked(QTreeWidgetItem \*item, int column);

void ini();

QString CNToPY(const QString &cnstr);

private slots:

void on\_pushButton\_5\_clicked();

void on\_pushButton\_4\_clicked();

void on\_pushButton\_clicked();

private:

Ui::MainWindow \*ui;

string username;

};

void QuickSort(int sum);

int IncludeChinese(const char \*str);

#endif // Z\_H

private:

Ui::MainWindow \*ui;

string username;

};

void QuickSort(int sum);

int IncludeChinese(const char \*str);

#endif // Z\_H

头文件 newperson.h

#ifndef NEWPERSON\_H

#define NEWPERSON\_H

#include <QMainWindow>

#include "Login.h"

namespace Ui {

class Newperson;

}

class Newperson : public QMainWindow

{

Q\_OBJECT

public:

Newperson(string username,QWidget \*parent = nullptr);

int static checknumber(string num);

~Newperson();

private slots:

void on\_pushButton\_clicked();

void on\_pushButton\_2\_clicked();

private:

Ui::Newperson \*ui;

string currentname;

string num;

};

#endif // NEWPERSON\_H

头文件 owninformation.h

#ifndef OWNINFORMATION\_H

#define OWNINFORMATION\_H

#include <QMainWindow>

using namespace std;

namespace Ui {

class Owninformation;

}

class Owninformation : public QMainWindow

{

Q\_OBJECT

public:

explicit Owninformation(string username,QWidget \*parent = nullptr);

~Owninformation();

private slots:

void on\_pushButton\_clicked();

void on\_pushButton\_2\_clicked();

private:

Ui::Owninformation \*ui;

string username;

};

#endif // OWNINFORMATION\_H

头文件 queryinterface.h

#ifndef QUERYINTERFACE\_H

#define QUERYINTERFACE\_H

#include <QMainWindow>

#include "Login.h"

using namespace std;

namespace Ui {

class Queryinterface;

}

class Queryinterface : public QMainWindow

{

Q\_OBJECT

public:

Queryinterface(string name,QWidget \*parent = nullptr);

~Queryinterface();

void on\_treeWidget\_itemDoubleClicked(QTreeWidgetItem \*item, int column);

Ui::Queryinterface \*ui;

private slots:

void on\_pushButton\_3\_clicked();

private:

string username;

};

#endif // QUERYINTERFACE\_H

头文件 retrieverpassword.h

#ifndef RETRIEVERPASSWORD\_H

#define RETRIEVERPASSWORD\_H

#include <QWidget>

using namespace std;

namespace Ui {

class Retrieverpassword;

}

class Retrieverpassword : public QWidget

{

Q\_OBJECT

public:

explicit Retrieverpassword(QWidget \*parent = nullptr);

~Retrieverpassword();

private slots:

void on\_pushButton\_clicked();

void on\_pushButton\_2\_clicked();

public:

Ui::Retrieverpassword \*ui;

};

int compare(string str);

void writeUser(string name,string email,string password);

int check(string namebase);

#endif // RETRIEVERPASSWORD\_H

头文件 signup.h

#ifndef SIGNUP\_H

#define SIGNUP\_H

#include <QWidget>

#include <stdlib.h>

using namespace std;

namespace Ui {

class Signup;

}

class Signup : public QWidget

{

Q\_OBJECT

public:

explicit Signup(QWidget \*parent = nullptr);

~Signup();

private slots:

void on\_pushButton\_clicked();

void on\_pushButton\_2\_clicked();

private:

Ui::Signup \*ui;

};

int checkusername(string name);

#endif // SIGNUP\_H

源文件 cnlhelper.cpp

#include "cnlhelper.h"

#include <QFile>

#include <QMutex>

CNLHelper \*CNLHelper::curHelper;

CNLHelper::CNLHelper(QObject \*parent) : QObject(parent)

{

//从配置文件读取拼音数组

QFile file(":/cnl.txt");

if (file.open(QFile::ReadOnly | QFile::Text)) {

QString str = file.readAll();

listPY = str.split(" ");

}

//加载简拼数组

listJP << "YDYQSXMWZSSXJBYMGCCZQPSSQBYCDSCDQLDYLYBSSJGYZZJJFKCCLZDHWDWZJLJPFYYNWJJTMYHZWZHFLZPPQHGSCYYYNJQYXXGJ";

listJP << "HHSDSJNKKTMOMLCRXYPSNQSECCQZGGLLYJLMYZZSECYKYYHQWJSSGGYXYZYJWWKDJHYCHMYXJTLXJYQBYXZLDWRDJRWYSRLDZJPC";

listJP << "BZJJBRCFTLECZSTZFXXZHTRQHYBDLYCZSSYMMRFMYQZPWWJJYFCRWFDFZQPYDDWYXKYJAWJFFXYPSFTZYHHYZYSWCJYXSCLCXXWZ";

listJP << "ZXNBGNNXBXLZSZSBSGPYSYZDHMDZBQBZCWDZZYYTZHBTSYYBZGNTNXQYWQSKBPHHLXGYBFMJEBJHHGQTJCYSXSTKZHLYCKGLYSMZ";

listJP << "XYALMELDCCXGZYRJXSDLTYZCQKCNNJWHJTZZCQLJSTSTBNXBTYXCEQXGKWJYFLZQLYHYXSPSFXLMPBYSXXXYDJCZYLLLSJXFHJXP";

listJP << "JBTFFYABYXBHZZBJYZLWLCZGGBTSSMDTJZXPTHYQTGLJSCQFZKJZJQNLZWLSLHDZBWJNCJZYZSQQYCQYRZCJJWYBRTWPYFTWEXCS";

listJP << "KDZCTBZHYZZYYJXZCFFZZMJYXXSDZZOTTBZLQWFCKSZSXFYRLNYJMBDTHJXSQQCCSBXYYTSYFBXDZTGBCNSLCYZZPSAZYZZSCJCS";

listJP << "HZQYDXLBPJLLMQXTYDZXSQJTZPXLCGLQTZWJBHCTSYJSFXYEJJTLBGXSXJMYJQQPFZASYJNTYDJXKJCDJSZCBARTDCLYJQMWNQNC";

listJP << "LLLKBYBZZSYHQQLTWLCCXTXLLZNTYLNEWYZYXCZXXGRKRMTCNDNJTSYYSSDQDGHSDBJGHRWRQLYBGLXHLGTGXBQJDZPYJSJYJCTM";

listJP << "RNYMGRZJCZGJMZMGXMPRYXKJNYMSGMZJYMKMFXMLDTGFBHCJHKYLPFMDXLQJJSMTQGZSJLQDLDGJYCALCMZCSDJLLNXDJFFFFJCZ";

listJP << "FMZFFPFKHKGDPSXKTACJDHHZDDCRRCFQYJKQCCWJDXHWJLYLLZGCFCQDSMLZPBJJPLSBCJGGDCKKDEZSQCCKJGCGKDJTJDLZYCXK";

listJP << "LQSCGJCLTFPCQCZGWPJDQYZJJBYJHSJDZWGFSJGZKQCCZLLPSPKJGQJHZZLJPLGJGJJTHJJYJZCZMLZLYQBGJWMLJKXZDZNJQSYZ";

listJP << "MLJLLJKYWXMKJLHSKJGBMCLYYMKXJQLBMLLKMDXXKWYXYSLMLPSJQQJQXYXFJTJDXMXXLLCXQBSYJBGWYMBGGBCYXPJYGPEPFGDJ";

listJP << "GBHBNSQJYZJKJKHXQFGQZKFHYGKHDKLLSDJQXPQYKYBNQSXQNSZSWHBSXWHXWBZZXDMNSJBSBKBBZKLYLXGWXDRWYQZMYWSJQLCJ";

listJP << "XXJXKJEQXSCYETLZHLYYYSDZPAQYZCMTLSHTZCFYZYXYLJSDCJQAGYSLCQLYYYSHMRQQKLDXZSCSSSYDYCJYSFSJBFRSSZQSBXXP";

listJP << "XJYSDRCKGJLGDKZJZBDKTCSYQPYHSTCLDJDHMXMCGXYZHJDDTMHLTXZXYLYMOHYJCLTYFBQQXPFBDFHHTKSQHZYYWCNXXCRWHOWG";

listJP << "YJLEGWDQCWGFJYCSNTMYTOLBYGWQWESJPWNMLRYDZSZTXYQPZGCWXHNGPYXSHMYQJXZTDPPBFYHZHTJYFDZWKGKZBLDNTSXHQEEG";

listJP << "ZZYLZMMZYJZGXZXKHKSTXNXXWYLYAPSTHXDWHZYMPXAGKYDXBHNHXKDPJNMYHYLPMGOCSLNZHKXXLPZZLBMLSFBHHGYGYYGGBHSC";

listJP << "YAQTYWLXTZQCEZYDQDQMMHTKLLSZHLSJZWFYHQSWSCWLQAZYNYTLSXTHAZNKZZSZZLAXXZWWCTGQQTDDYZTCCHYQZFLXPSLZYGPZ";

listJP << "SZNGLNDQTBDLXGTCTAJDKYWNSYZLJHHZZCWNYYZYWMHYCHHYXHJKZWSXHZYXLYSKQYSPSLYZWMYPPKBYGLKZHTYXAXQSYSHXASMC";

listJP << "HKDSCRSWJPWXSGZJLWWSCHSJHSQNHCSEGNDAQTBAALZZMSSTDQJCJKTSCJAXPLGGXHHGXXZCXPDMMHLDGTYBYSJMXHMRCPXXJZCK";

listJP << "ZXSHMLQXXTTHXWZFKHCCZDYTCJYXQHLXDHYPJQXYLSYYDZOZJNYXQEZYSQYAYXWYPDGXDDXSPPYZNDLTWRHXYDXZZJHTCXMCZLHP";

listJP << "YYYYMHZLLHNXMYLLLMDCPPXHMXDKYCYRDLTXJCHHZZXZLCCLYLNZSHZJZZLNNRLWHYQSNJHXYNTTTKYJPYCHHYEGKCTTWLGQRLGG";

listJP << "TGTYGYHPYHYLQYQGCWYQKPYYYTTTTLHYHLLTYTTSPLKYZXGZWGPYDSSZZDQXSKCQNMJJZZBXYQMJRTFFBTKHZKBXLJJKDXJTLBWF";

listJP << "ZPPTKQTZTGPDGNTPJYFALQMKGXBDCLZFHZCLLLLADPMXDJHLCCLGYHDZFGYDDGCYYFGYDXKSSEBDHYKDKDKHNAXXYBPBYYHXZQGA";

listJP << "FFQYJXDMLJCSQZLLPCHBSXGJYNDYBYQSPZWJLZKSDDTACTBXZDYZYPJZQSJNKKTKNJDJGYYPGTLFYQKASDNTCYHBLWDZHBBYDWJR";

listJP << "YGKZYHEYYFJMSDTYFZJJHGCXPLXHLDWXXJKYTCYKSSSMTWCTTQZLPBSZDZWZXGZAGYKTYWXLHLSPBCLLOQMMZSSLCMBJCSZZKYDC";

listJP << "ZJGQQDSMCYTZQQLWZQZXSSFPTTFQMDDZDSHDTDWFHTDYZJYQJQKYPBDJYYXTLJHDRQXXXHAYDHRJLKLYTWHLLRLLRCXYLBWSRSZZ";

listJP << "SYMKZZHHKYHXKSMDSYDYCJPBZBSQLFCXXXNXKXWYWSDZYQOGGQMMYHCDZTTFJYYBGSTTTYBYKJDHKYXBELHTYPJQNFXFDYKZHQKZ";

listJP << "BYJTZBXHFDXKDASWTAWAJLDYJSFHBLDNNTNQJTJNCHXFJSRFWHZFMDRYJYJWZPDJKZYJYMPCYZNYNXFBYTFYFWYGDBNZZZDNYTXZ";

listJP << "EMMQBSQEHXFZMBMFLZZSRXYMJGSXWZJSPRYDJSJGXHJJGLJJYNZZJXHGXKYMLPYYYCXYTWQZSWHWLYRJLPXSLSXMFSWWKLCTNXNY";

listJP << "NPSJSZHDZEPTXMYYWXYYSYWLXJQZQXZDCLEEELMCPJPCLWBXSQHFWWTFFJTNQJHJQDXHWLBYZNFJLALKYYJLDXHHYCSTYYWNRJYX";

listJP << "YWTRMDRQHWQCMFJDYZMHMYYXJWMYZQZXTLMRSPWWCHAQBXYGZYPXYYRRCLMPYMGKSJSZYSRMYJSNXTPLNBAPPYPYLXYYZKYNLDZY";

listJP << "JZCZNNLMZHHARQMPGWQTZMXXMLLHGDZXYHXKYXYCJMFFYYHJFSBSSQLXXNDYCANNMTCJCYPRRNYTYQNYYMBMSXNDLYLYSLJRLXYS";

listJP << "XQMLLYZLZJJJKYZZCSFBZXXMSTBJGNXYZHLXNMCWSCYZYFZLXBRNNNYLBNRTGZQYSATSWRYHYJZMZDHZGZDWYBSSCSKXSYHYTXXG";

listJP << "CQGXZZSHYXJSCRHMKKBXCZJYJYMKQHZJFNBHMQHYSNJNZYBKNQMCLGQHWLZNZSWXKHLJHYYBQLBFCDSXDLDSPFZPSKJYZWZXZDDX";

listJP << "JSMMEGJSCSSMGCLXXKYYYLNYPWWWGYDKZJGGGZGGSYCKNJWNJPCXBJJTQTJWDSSPJXZXNZXUMELPXFSXTLLXCLJXJJLJZXCTPSWX";

listJP << "LYDHLYQRWHSYCSQYYBYAYWJJJQFWQCQQCJQGXALDBZZYJGKGXPLTZYFXJLTPADKYQHPMATLCPDCKBMTXYBHKLENXDLEEGQDYMSAW";

listJP << "HZMLJTWYGXLYQZLJEEYYBQQFFNLYXRDSCTGJGXYYNKLLYQKCCTLHJLQMKKZGCYYGLLLJDZGYDHZWXPYSJBZKDZGYZZHYWYFQYTYZ";

listJP << "SZYEZZLYMHJJHTSMQWYZLKYYWZCSRKQYTLTDXWCTYJKLWSQZWBDCQYNCJSRSZJLKCDCDTLZZZACQQZZDDXYPLXZBQJYLZLLLQDDZ";

listJP << "QJYJYJZYXNYYYNYJXKXDAZWYRDLJYYYRJLXLLDYXJCYWYWNQCCLDDNYYYNYCKCZHXXCCLGZQJGKWPPCQQJYSBZZXYJSQPXJPZBSB";

listJP << "DSFNSFPZXHDWZTDWPPTFLZZBZDMYYPQJRSDZSQZSQXBDGCPZSWDWCSQZGMDHZXMWWFYBPDGPHTMJTHZSMMBGZMBZJCFZWFZBBZMQ";

listJP << "CFMBDMCJXLGPNJBBXGYHYYJGPTZGZMQBQTCGYXJXLWZKYDPDYMGCFTPFXYZTZXDZXTGKMTYBBCLBJASKYTSSQYYMSZXFJEWLXLLS";

listJP << "ZBQJJJAKLYLXLYCCTSXMCWFKKKBSXLLLLJYXTYLTJYYTDPJHNHNNKBYQNFQYYZBYYESSESSGDYHFHWTCJBSDZZTFDMXHCNJZYMQW";

listJP << "SRYJDZJQPDQBBSTJGGFBKJBXTGQHNGWJXJGDLLTHZHHYYYYYYSXWTYYYCCBDBPYPZYCCZYJPZYWCBDLFWZCWJDXXHYHLHWZZXJTC";

listJP << "ZLCDPXUJCZZZLYXJJTXPHFXWPYWXZPTDZZBDZCYHJHMLXBQXSBYLRDTGJRRCTTTHYTCZWMXFYTWWZCWJWXJYWCSKYBZSCCTZQNHX";

listJP << "NWXXKHKFHTSWOCCJYBCMPZZYKBNNZPBZHHZDLSYDDYTYFJPXYNGFXBYQXCBHXCPSXTYZDMKYSNXSXLHKMZXLYHDHKWHXXSSKQYHH";

listJP << "CJYXGLHZXCSNHEKDTGZXQYPKDHEXTYKCNYMYYYPKQYYYKXZLTHJQTBYQHXBMYHSQCKWWYLLHCYYLNNEQXQWMCFBDCCMLJGGXDQKT";

listJP << "LXKGNQCDGZJWYJJLYHHQTTTNWCHMXCXWHWSZJYDJCCDBQCDGDNYXZTHCQRXCBHZTQCBXWGQWYYBXHMBYMYQTYEXMQKYAQYRGYZSL";

listJP << "FYKKQHYSSQYSHJGJCNXKZYCXSBXYXHYYLSTYCXQTHYSMGSCPMMGCCCCCMTZTASMGQZJHKLOSQYLSWTMXSYQKDZLJQQYPLSYCZTCQ";

listJP << "QPBBQJZCLPKHQZYYXXDTDDTSJCXFFLLCHQXMJLWCJCXTSPYCXNDTJSHJWXDQQJSKXYAMYLSJHMLALYKXCYYDMNMDQMXMCZNNCYBZ";

listJP << "KKYFLMCHCMLHXRCJJHSYLNMTJZGZGYWJXSRXCWJGJQHQZDQJDCJJZKJKGDZQGJJYJYLXZXXCDQHHHEYTMHLFSBDJSYYSHFYSTCZQ";

listJP << "LPBDRFRZTZYKYWHSZYQKWDQZRKMSYNBCRXQBJYFAZPZZEDZCJYWBCJWHYJBQSZYWRYSZPTDKZPFPBNZTKLQYHBBZPNPPTYZZYBQN";

listJP << "YDCPJMMCYCQMCYFZZDCMNLFPBPLNGQJTBTTNJZPZBBZNJKLJQYLNBZQHKSJZNGGQSZZKYXSHPZSNBCGZKDDZQANZHJKDRTLZLSWJ";

listJP << "LJZLYWTJNDJZJHXYAYNCBGTZCSSQMNJPJYTYSWXZFKWJQTKHTZPLBHSNJZSYZBWZZZZLSYLSBJHDWWQPSLMMFBJDWAQYZTCJTBNN";

listJP << "WZXQXCDSLQGDSDPDZHJTQQPSWLYYJZLGYXYZLCTCBJTKTYCZJTQKBSJLGMGZDMCSGPYNJZYQYYKNXRPWSZXMTNCSZZYXYBYHYZAX";

listJP << "YWQCJTLLCKJJTJHGDXDXYQYZZBYWDLWQCGLZGJGQRQZCZSSBCRPCSKYDZNXJSQGXSSJMYDNSTZTPBDLTKZWXQWQTZEXNQCZGWEZK";

listJP << "SSBYBRTSSSLCCGBPSZQSZLCCGLLLZXHZQTHCZMQGYZQZNMCOCSZJMMZSQPJYGQLJYJPPLDXRGZYXCCSXHSHGTZNLZWZKJCXTCFCJ";

listJP << "XLBMQBCZZWPQDNHXLJCTHYZLGYLNLSZZPCXDSCQQHJQKSXZPBAJYEMSMJTZDXLCJYRYYNWJBNGZZTMJXLTBSLYRZPYLSSCNXPHLL";

listJP << "HYLLQQZQLXYMRSYCXZLMMCZLTZSDWTJJLLNZGGQXPFSKYGYGHBFZPDKMWGHCXMSGDXJMCJZDYCABXJDLNBCDQYGSKYDQTXDJJYXM";

listJP << "SZQAZDZFSLQXYJSJZYLBTXXWXQQZBJZUFBBLYLWDSLJHXJYZJWTDJCZFQZQZZDZSXZZQLZCDZFJHYSPYMPQZMLPPLFFXJJNZZYLS";

listJP << "JEYQZFPFZKSYWJJJHRDJZZXTXXGLGHYDXCSKYSWMMZCWYBAZBJKSHFHJCXMHFQHYXXYZFTSJYZFXYXPZLCHMZMBXHZZSXYFYMNCW";

listJP << "DABAZLXKTCSHHXKXJJZJSTHYGXSXYYHHHJWXKZXSSBZZWHHHCWTZZZPJXSNXQQJGZYZYWLLCWXZFXXYXYHXMKYYSWSQMNLNAYCYS";

listJP << "PMJKHWCQHYLAJJMZXHMMCNZHBHXCLXTJPLTXYJHDYYLTTXFSZHYXXSJBJYAYRSMXYPLCKDUYHLXRLNLLSTYZYYQYGYHHSCCSMZCT";

listJP << "ZQXKYQFPYYRPFFLKQUNTSZLLZMWWTCQQYZWTLLMLMPWMBZSSTZRBPDDTLQJJBXZCSRZQQYGWCSXFWZLXCCRSZDZMCYGGDZQSGTJS";

listJP << "WLJMYMMZYHFBJDGYXCCPSHXNZCSBSJYJGJMPPWAFFYFNXHYZXZYLREMZGZCYZSSZDLLJCSQFNXZKPTXZGXJJGFMYYYSNBTYLBNLH";

listJP << "PFZDCYFBMGQRRSSSZXYSGTZRNYDZZCDGPJAFJFZKNZBLCZSZPSGCYCJSZLMLRSZBZZLDLSLLYSXSQZQLYXZLSKKBRXBRBZCYCXZZ";

listJP << "ZEEYFGKLZLYYHGZSGZLFJHGTGWKRAAJYZKZQTSSHJJXDCYZUYJLZYRZDQQHGJZXSSZBYKJPBFRTJXLLFQWJHYLQTYMBLPZDXTZYG";

listJP << "BDHZZRBGXHWNJTJXLKSCFSMWLSDQYSJTXKZSCFWJLBXFTZLLJZLLQBLSQMQQCGCZFPBPHZCZJLPYYGGDTGWDCFCZQYYYQYSSCLXZ";

listJP << "SKLZZZGFFCQNWGLHQYZJJCZLQZZYJPJZZBPDCCMHJGXDQDGDLZQMFGPSYTSDYFWWDJZJYSXYYCZCYHZWPBYKXRYLYBHKJKSFXTZJ";

listJP << "MMCKHLLTNYYMSYXYZPYJQYCSYCWMTJJKQYRHLLQXPSGTLYYCLJSCPXJYZFNMLRGJJTYZBXYZMSJYJHHFZQMSYXRSZCWTLRTQZSST";

listJP << "KXGQKGSPTGCZNJSJCQCXHMXGGZTQYDJKZDLBZSXJLHYQGGGTHQSZPYHJHHGYYGKGGCWJZZYLCZLXQSFTGZSLLLMLJSKCTBLLZZSZ";

listJP << "MMNYTPZSXQHJCJYQXYZXZQZCPSHKZZYSXCDFGMWQRLLQXRFZTLYSTCTMJCXJJXHJNXTNRZTZFQYHQGLLGCXSZSJDJLJCYDSJTLNY";

listJP << "XHSZXCGJZYQPYLFHDJSBPCCZHJJJQZJQDYBSSLLCMYTTMQTBHJQNNYGKYRQYQMZGCJKPDCGMYZHQLLSLLCLMHOLZGDYYFZSLJCQZ";

listJP << "LYLZQJESHNYLLJXGJXLYSYYYXNBZLJSSZCQQCJYLLZLTJYLLZLLBNYLGQCHXYYXOXCXQKYJXXXYKLXSXXYQXCYKQXQCSGYXXYQXY";

listJP << "GYTQOHXHXPYXXXULCYEYCHZZCBWQBBWJQZSCSZSSLZYLKDESJZWMYMCYTSDSXXSCJPQQSQYLYYZYCMDJDZYWCBTJSYDJKCYDDJLB";

listJP << "DJJSODZYSYXQQYXDHHGQQYQHDYXWGMMMAJDYBBBPPBCMUUPLJZSMTXERXJMHQNUTPJDCBSSMSSSTKJTSSMMTRCPLZSZMLQDSDMJM";

listJP << "QPNQDXCFYNBFSDQXYXHYAYKQYDDLQYYYSSZBYDSLNTFQTZQPZMCHDHCZCWFDXTMYQSPHQYYXSRGJCWTJTZZQMGWJJTJHTQJBBHWZ";

listJP << "PXXHYQFXXQYWYYHYSCDYDHHQMNMTMWCPBSZPPZZGLMZFOLLCFWHMMSJZTTDHZZYFFYTZZGZYSKYJXQYJZQBHMBZZLYGHGFMSHPZF";

listJP << "ZSNCLPBQSNJXZSLXXFPMTYJYGBXLLDLXPZJYZJYHHZCYWHJYLSJEXFSZZYWXKZJLUYDTMLYMQJPWXYHXSKTQJEZRPXXZHHMHWQPW";

listJP << "QLYJJQJJZSZCPHJLCHHNXJLQWZJHBMZYXBDHHYPZLHLHLGFWLCHYYTLHJXCJMSCPXSTKPNHQXSRTYXXTESYJCTLSSLSTDLLLWWYH";

listJP << "DHRJZSFGXTSYCZYNYHTDHWJSLHTZDQDJZXXQHGYLTZPHCSQFCLNJTCLZPFSTPDYNYLGMJLLYCQHYSSHCHYLHQYQTMZYPBYWRFQYK";

listJP << "QSYSLZDQJMPXYYSSRHZJNYWTQDFZBWWTWWRXCWHGYHXMKMYYYQMSMZHNGCEPMLQQMTCWCTMMPXJPJJHFXYYZSXZHTYBMSTSYJTTQ";

listJP << "QQYYLHYNPYQZLCYZHZWSMYLKFJXLWGXYPJYTYSYXYMZCKTTWLKSMZSYLMPWLZWXWQZSSAQSYXYRHSSNTSRAPXCPWCMGDXHXZDZYF";

listJP << "JHGZTTSBJHGYZSZYSMYCLLLXBTYXHBBZJKSSDMALXHYCFYGMQYPJYCQXJLLLJGSLZGQLYCJCCZOTYXMTMTTLLWTGPXYMZMKLPSZZ";

listJP << "ZXHKQYSXCTYJZYHXSHYXZKXLZWPSQPYHJWPJPWXQQYLXSDHMRSLZZYZWTTCYXYSZZSHBSCCSTPLWSSCJCHNLCGCHSSPHYLHFHHXJ";

listJP << "SXYLLNYLSZDHZXYLSXLWZYKCLDYAXZCMDDYSPJTQJZLNWQPSSSWCTSTSZLBLNXSMNYYMJQBQHRZWTYYDCHQLXKPZWBGQYBKFCMZW";

listJP << "PZLLYYLSZYDWHXPSBCMLJBSCGBHXLQHYRLJXYSWXWXZSLDFHLSLYNJLZYFLYJYCDRJLFSYZFSLLCQYQFGJYHYXZLYLMSTDJCYHBZ";

listJP << "LLNWLXXYGYYHSMGDHXXHHLZZJZXCZZZCYQZFNGWPYLCPKPYYPMCLQKDGXZGGWQBDXZZKZFBXXLZXJTPJPTTBYTSZZDWSLCHZHSLT";

listJP << "YXHQLHYXXXYYZYSWTXZKHLXZXZPYHGCHKCFSYHUTJRLXFJXPTZTWHPLYXFCRHXSHXKYXXYHZQDXQWULHYHMJTBFLKHTXCWHJFWJC";

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "XXJXKJEQXSCYETLZHLYYYSDZPAQYZCMTLSHTZCFYZYXYLJSDCJQAGYSLCQLYYYSHMRQQKLDXZSCSSSYDYCJYSFSJBFRSSZQSBXXP";

listJP << "XJYSDRCKGJLGDKZJZBDKTCSYQPYHSTCLDJDHMXMCGXYZHJDDTMHLTXZXYLYMOHYJCLTYFBQQXPFBDFHHTKSQHZYYWCNXXCRWHOWG";

listJP << "YJLEGWDQCWGFJYCSNTMYTOLBYGWQWESJPWNMLRYDZSZTXYQPZGCWXHNGPYXSHMYQJXZTDPPBFYHZHTJYFDZWKGKZBLDNTSXHQEEG";

listJP << "ZZYLZMMZYJZGXZXKHKSTXNXXWYLYAPSTHXDWHZYMPXAGKYDXBHNHXKDPJNMYHYLPMGOCSLNZHKXXLPZZLBMLSFBHHGYGYYGGBHSC";

listJP << "YAQTYWLXTZQCEZYDQDQMMHTKLLSZHLSJZWFYHQSWSCWLQAZYNYTLSXTHAZNKZZSZZLAXXZWWCTGQQTDDYZTCCHYQZFLXPSLZYGPZ";

listJP << "SZNGLNDQTBDLXGTCTAJDKYWNSYZLJHHZZCWNYYZYWMHYCHHYXHJKZWSXHZYXLYSKQYSPSLYZWMYPPKBYGLKZHTYXAXQSYSHXASMC";

listJP << "HKDSCRSWJPWXSGZJLWWSCHSJHSQNHCSEGNDAQTBAALZZMSSTDQJCJKTSCJAXPLGGXHHGXXZCXPDMMHLDGTYBYSJMXHMRCPXXJZCK";

listJP << "ZXSHMLQXXTTHXWZFKHCCZDYTCJYXQHLXDHYPJQXYLSYYDZOZJNYXQEZYSQYAYXWYPDGXDDXSPPYZNDLTWRHXYDXZZJHTCXMCZLHP";

listJP << "YYYYMHZLLHNXMYLLLMDCPPXHMXDKYCYRDLTXJCHHZZXZLCCLYLNZSHZJZZLNNRLWHYQSNJHXYNTTTKYJPYCHHYEGKCTTWLGQRLGG";

listJP << "TGTYGYHPYHYLQYQGCWYQKPYYYTTTTLHYHLLTYTTSPLKYZXGZWGPYDSSZZDQXSKCQNMJJZZBXYQMJRTFFBTKHZKBXLJJKDXJTLBWF";

listJP << "ZPPTKQTZTGPDGNTPJYFALQMKGXBDCLZFHZCLLLLADPMXDJHLCCLGYHDZFGYDDGCYYFGYDXKSSEBDHYKDKDKHNAXXYBPBYYHXZQGA";

listJP << "FFQYJXDMLJCSQZLLPCHBSXGJYNDYBYQSPZWJLZKSDDTACTBXZDYZYPJZQSJNKKTKNJDJGYYPGTLFYQKASDNTCYHBLWDZHBBYDWJR";

listJP << "YGKZYHEYYFJMSDTYFZJJHGCXPLXHLDWXXJKYTCYKSSSMTWCTTQZLPBSZDZWZXGZAGYKTYWXLHLSPBCLLOQMMZSSLCMBJCSZZKYDC";

listJP << "ZJGQQDSMCYTZQQLWZQZXSSFPTTFQMDDZDSHDTDWFHTDYZJYQJQKYPBDJYYXTLJHDRQXXXHAYDHRJLKLYTWHLLRLLRCXYLBWSRSZZ";

listJP << "SYMKZZHHKYHXKSMDSYDYCJPBZBSQLFCXXXNXKXWYWSDZYQOGGQMMYHCDZTTFJYYBGSTTTYBYKJDHKYXBELHTYPJQNFXFDYKZHQKZ";

listJP << "BYJTZBXHFDXKDASWTAWAJLDYJSFHBLDNNTNQJTJNCHXFJSRFWHZFMDRYJYJWZPDJKZYJYMPCYZNYNXFBYTFYFWYGDBNZZZDNYTXZ";

listJP << "EMMQBSQEHXFZMBMFLZZSRXYMJGSXWZJSPRYDJSJGXHJJGLJJYNZZJXHGXKYMLPYYYCXYTWQZSWHWLYRJLPXSLSXMFSWWKLCTNXNY";

listJP << "NPSJSZHDZEPTXMYYWXYYSYWLXJQZQXZDCLEEELMCPJPCLWBXSQHFWWTFFJTNQJHJQDXHWLBYZNFJLALKYYJLDXHHYCSTYYWNRJYX";

listJP << "YWTRMDRQHWQCMFJDYZMHMYYXJWMYZQZXTLMRSPWWCHAQBXYGZYPXYYRRCLMPYMGKSJSZYSRMYJSNXTPLNBAPPYPYLXYYZKYNLDZY";

listJP << "JZCZNNLMZHHARQMPGWQTZMXXMLLHGDZXYHXKYXYCJMFFYYHJFSBSSQLXXNDYCANNMTCJCYPRRNYTYQNYYMBMSXNDLYLYSLJRLXYS";

listJP << "XQMLLYZLZJJJKYZZCSFBZXXMSTBJGNXYZHLXNMCWSCYZYFZLXBRNNNYLBNRTGZQYSATSWRYHYJZMZDHZGZDWYBSSCSKXSYHYTXXG";

listJP << "CQGXZZSHYXJSCRHMKKBXCZJYJYMKQHZJFNBHMQHYSNJNZYBKNQMCLGQHWLZNZSWXKHLJHYYBQLBFCDSXDLDSPFZPSKJYZWZXZDDX";

listJP << "JSMMEGJSCSSMGCLXXKYYYLNYPWWWGYDKZJGGGZGGSYCKNJWNJPCXBJJTQTJWDSSPJXZXNZXUMELPXFSXTLLXCLJXJJLJZXCTPSWX";

listJP << "LYDHLYQRWHSYCSQYYBYAYWJJJQFWQCQQCJQGXALDBZZYJGKGXPLTZYFXJLTPADKYQHPMATLCPDCKBMTXYBHKLENXDLEEGQDYMSAW";

listJP << "HZMLJTWYGXLYQZLJEEYYBQQFFNLYXRDSCTGJGXYYNKLLYQKCCTLHJLQMKKZGCYYGLLLJDZGYDHZWXPYSJBZKDZGYZZHYWYFQYTYZ";

listJP << "SZYEZZLYMHJJHTSMQWYZLKYYWZCSRKQYTLTDXWCTYJKLWSQZWBDCQYNCJSRSZJLKCDCDTLZZZACQQZZDDXYPLXZBQJYLZLLLQDDZ";

listJP << "QJYJYJZYXNYYYNYJXKXDAZWYRDLJYYYRJLXLLDYXJCYWYWNQCCLDDNYYYNYCKCZHXXCCLGZQJGKWPPCQQJYSBZZXYJSQPXJPZBSB";

listJP << "DSFNSFPZXHDWZTDWPPTFLZZBZDMYYPQJRSDZSQZSQXBDGCPZSWDWCSQZGMDHZXMWWFYBPDGPHTMJTHZSMMBGZMBZJCFZWFZBBZMQ";

listJP << "CFMBDMCJXLGPNJBBXGYHYYJGPTZGZMQBQTCGYXJXLWZKYDPDYMGCFTPFXYZTZXDZXTGKMTYBBCLBJASKYTSSQYYMSZXFJEWLXLLS";

listJP << "ZBQJJJAKLYLXLYCCTSXMCWFKKKBSXLLLLJYXTYLTJYYTDPJHNHNNKBYQNFQYYZBYYESSESSGDYHFHWTCJBSDZZTFDMXHCNJZYMQW";

listJP << "SRYJDZJQPDQBBSTJGGFBKJBXTGQHNGWJXJGDLLTHZHHYYYYYYSXWTYYYCCBDBPYPZYCCZYJPZYWCBDLFWZCWJDXXHYHLHWZZXJTC";

listJP << "ZLCDPXUJCZZZLYXJJTXPHFXWPYWXZPTDZZBDZCYHJHMLXBQXSBYLRDTGJRRCTTTHYTCZWMXFYTWWZCWJWXJYWCSKYBZSCCTZQNHX";

listJP << "NWXXKHKFHTSWOCCJYBCMPZZYKBNNZPBZHHZDLSYDDYTYFJPXYNGFXBYQXCBHXCPSXTYZDMKYSNXSXLHKMZXLYHDHKWHXXSSKQYHH";

listJP << "CJYXGLHZXCSNHEKDTGZXQYPKDHEXTYKCNYMYYYPKQYYYKXZLTHJQTBYQHXBMYHSQCKWWYLLHCYYLNNEQXQWMCFBDCCMLJGGXDQKT";

listJP << "LXKGNQCDGZJWYJJLYHHQTTTNWCHMXCXWHWSZJYDJCCDBQCDGDNYXZTHCQRXCBHZTQCBXWGQWYYBXHMBYMYQTYEXMQKYAQYRGYZSL";

listJP << "FYKKQHYSSQYSHJGJCNXKZYCXSBXYXHYYLSTYCXQTHYSMGSCPMMGCCCCCMTZTASMGQZJHKLOSQYLSWTMXSYQKDZLJQQYPLSYCZTCQ";

listJP << "QPBBQJZCLPKHQZYYXXDTDDTSJCXFFLLCHQXMJLWCJCXTSPYCXNDTJSHJWXDQQJSKXYAMYLSJHMLALYKXCYYDMNMDQMXMCZNNCYBZ";

listJP << "KKYFLMCHCMLHXRCJJHSYLNMTJZGZGYWJXSRXCWJGJQHQZDQJDCJJZKJKGDZQGJJYJYLXZXXCDQHHHEYTMHLFSBDJSYYSHFYSTCZQ";

listJP << "LPBDRFRZTZYKYWHSZYQKWDQZRKMSYNBCRXQBJYFAZPZZEDZCJYWBCJWHYJBQSZYWRYSZPTDKZPFPBNZTKLQYHBBZPNPPTYZZYBQN";

listJP << "YDCPJMMCYCQMCYFZZDCMNLFPBPLNGQJTBTTNJZPZBBZNJKLJQYLNBZQHKSJZNGGQSZZKYXSHPZSNBCGZKDDZQANZHJKDRTLZLSWJ";

listJP << "LJZLYWTJNDJZJHXYAYNCBGTZCSSQMNJPJYTYSWXZFKWJQTKHTZPLBHSNJZSYZBWZZZZLSYLSBJHDWWQPSLMMFBJDWAQYZTCJTBNN";

listJP << "WZXQXCDSLQGDSDPDZHJTQQPSWLYYJZLGYXYZLCTCBJTKTYCZJTQKBSJLGMGZDMCSGPYNJZYQYYKNXRPWSZXMTNCSZZYXYBYHYZAX";

listJP << "YWQCJTLLCKJJTJHGDXDXYQYZZBYWDLWQCGLZGJGQRQZCZSSBCRPCSKYDZNXJSQGXSSJMYDNSTZTPBDLTKZWXQWQTZEXNQCZGWEZK";

listJP << "SSBYBRTSSSLCCGBPSZQSZLCCGLLLZXHZQTHCZMQGYZQZNMCOCSZJMMZSQPJYGQLJYJPPLDXRGZYXCCSXHSHGTZNLZWZKJCXTCFCJ";

listJP << "XLBMQBCZZWPQDNHXLJCTHYZLGYLNLSZZPCXDSCQQHJQKSXZPBAJYEMSMJTZDXLCJYRYYNWJBNGZZTMJXLTBSLYRZPYLSSCNXPHLL";

listJP << "HYLLQQZQLXYMRSYCXZLMMCZLTZSDWTJJLLNZGGQXPFSKYGYGHBFZPDKMWGHCXMSGDXJMCJZDYCABXJDLNBCDQYGSKYDQTXDJJYXM";

listJP << "SZQAZDZFSLQXYJSJZYLBTXXWXQQZBJZUFBBLYLWDSLJHXJYZJWTDJCZFQZQZZDZSXZZQLZCDZFJHYSPYMPQZMLPPLFFXJJNZZYLS";

listJP << "JEYQZFPFZKSYWJJJHRDJZZXTXXGLGHYDXCSKYSWMMZCWYBAZBJKSHFHJCXMHFQHYXXYZFTSJYZFXYXPZLCHMZMBXHZZSXYFYMNCW";

listJP << "DABAZLXKTCSHHXKXJJZJSTHYGXSXYYHHHJWXKZXSSBZZWHHHCWTZZZPJXSNXQQJGZYZYWLLCWXZFXXYXYHXMKYYSWSQMNLNAYCYS";

listJP << "PMJKHWCQHYLAJJMZXHMMCNZHBHXCLXTJPLTXYJHDYYLTTXFSZHYXXSJBJYAYRSMXYPLCKDUYHLXRLNLLSTYZYYQYGYHHSCCSMZCT";

listJP << "ZQXKYQFPYYRPFFLKQUNTSZLLZMWWTCQQYZWTLLMLMPWMBZSSTZRBPDDTLQJJBXZCSRZQQYGWCSXFWZLXCCRSZDZMCYGGDZQSGTJS";

listJP << "WLJMYMMZYHFBJDGYXCCPSHXNZCSBSJYJGJMPPWAFFYFNXHYZXZYLREMZGZCYZSSZDLLJCSQFNXZKPTXZGXJJGFMYYYSNBTYLBNLH";

listJP << "PFZDCYFBMGQRRSSSZXYSGTZRNYDZZCDGPJAFJFZKNZBLCZSZPSGCYCJSZLMLRSZBZZLDLSLLYSXSQZQLYXZLSKKBRXBRBZCYCXZZ";

listJP << "ZEEYFGKLZLYYHGZSGZLFJHGTGWKRAAJYZKZQTSSHJJXDCYZUYJLZYRZDQQHGJZXSSZBYKJPBFRTJXLLFQWJHYLQTYMBLPZDXTZYG";

listJP << "BDHZZRBGXHWNJTJXLKSCFSMWLSDQYSJTXKZSCFWJLBXFTZLLJZLLQBLSQMQQCGCZFPBPHZCZJLPYYGGDTGWDCFCZQYYYQYSSCLXZ";

listJP << "SKLZZZGFFCQNWGLHQYZJJCZLQZZYJPJZZBPDCCMHJGXDQDGDLZQMFGPSYTSDYFWWDJZJYSXYYCZCYHZWPBYKXRYLYBHKJKSFXTZJ";

listJP << "MMCKHLLTNYYMSYXYZPYJQYCSYCWMTJJKQYRHLLQXPSGTLYYCLJSCPXJYZFNMLRGJJTYZBXYZMSJYJHHFZQMSYXRSZCWTLRTQZSST";

listJP << "KXGQKGSPTGCZNJSJCQCXHMXGGZTQYDJKZDLBZSXJLHYQGGGTHQSZPYHJHHGYYGKGGCWJZZYLCZLXQSFTGZSLLLMLJSKCTBLLZZSZ";

listJP << "MMNYTPZSXQHJCJYQXYZXZQZCPSHKZZYSXCDFGMWQRLLQXRFZTLYSTCTMJCXJJXHJNXTNRZTZFQYHQGLLGCXSZSJDJLJCYDSJTLNY";

listJP << "XHSZXCGJZYQPYLFHDJSBPCCZHJJJQZJQDYBSSLLCMYTTMQTBHJQNNYGKYRQYQMZGCJKPDCGMYZHQLLSLLCLMHOLZGDYYFZSLJCQZ";

listJP << "LYLZQJESHNYLLJXGJXLYSYYYXNBZLJSSZCQQCJYLLZLTJYLLZLLBNYLGQCHXYYXOXCXQKYJXXXYKLXSXXYQXCYKQXQCSGYXXYQXY";

listJP << "GYTQOHXHXPYXXXULCYEYCHZZCBWQBBWJQZSCSZSSLZYLKDESJZWMYMCYTSDSXXSCJPQQSQYLYYZYCMDJDZYWCBTJSYDJKCYDDJLB";

listJP << "DJJSODZYSYXQQYXDHHGQQYQHDYXWGMMMAJDYBBBPPBCMUUPLJZSMTXERXJMHQNUTPJDCBSSMSSSTKJTSSMMTRCPLZSZMLQDSDMJM";

listJP << "QPNQDXCFYNBFSDQXYXHYAYKQYDDLQYYYSSZBYDSLNTFQTZQPZMCHDHCZCWFDXTMYQSPHQYYXSRGJCWTJTZZQMGWJJTJHTQJBBHWZ";

listJP << "PXXHYQFXXQYWYYHYSCDYDHHQMNMTMWCPBSZPPZZGLMZFOLLCFWHMMSJZTTDHZZYFFYTZZGZYSKYJXQYJZQBHMBZZLYGHGFMSHPZF";

listJP << "ZSNCLPBQSNJXZSLXXFPMTYJYGBXLLDLXPZJYZJYHHZCYWHJYLSJEXFSZZYWXKZJLUYDTMLYMQJPWXYHXSKTQJEZRPXXZHHMHWQPW";

listJP << "QLYJJQJJZSZCPHJLCHHNXJLQWZJHBMZYXBDHHYPZLHLHLGFWLCHYYTLHJXCJMSCPXSTKPNHQXSRTYXXTESYJCTLSSLSTDLLLWWYH";

listJP << "DHRJZSFGXTSYCZYNYHTDHWJSLHTZDQDJZXXQHGYLTZPHCSQFCLNJTCLZPFSTPDYNYLGMJLLYCQHYSSHCHYLHQYQTMZYPBYWRFQYK";

listJP << "QSYSLZDQJMPXYYSSRHZJNYWTQDFZBWWTWWRXCWHGYHXMKMYYYQMSMZHNGCEPMLQQMTCWCTMMPXJPJJHFXYYZSXZHTYBMSTSYJTTQ";

listJP << "QQYYLHYNPYQZLCYZHZWSMYLKFJXLWGXYPJYTYSYXYMZCKTTWLKSMZSYLMPWLZWXWQZSSAQSYXYRHSSNTSRAPXCPWCMGDXHXZDZYF";

listJP << "JHGZTTSBJHGYZSZYSMYCLLLXBTYXHBBZJKSSDMALXHYCFYGMQYPJYCQXJLLLJGSLZGQLYCJCCZOTYXMTMTTLLWTGPXYMZMKLPSZZ";

listJP << "ZXHKQYSXCTYJZYHXSHYXZKXLZWPSQPYHJWPJPWXQQYLXSDHMRSLZZYZWTTCYXYSZZSHBSCCSTPLWSSCJCHNLCGCHSSPHYLHFHHXJ";

listJP << "SXYLLNYLSZDHZXYLSXLWZYKCLDYAXZCMDDYSPJTQJZLNWQPSSSWCTSTSZLBLNXSMNYYMJQBQHRZWTYYDCHQLXKPZWBGQYBKFCMZW";

listJP << "PZLLYYLSZYDWHXPSBCMLJBSCGBHXLQHYRLJXYSWXWXZSLDFHLSLYNJLZYFLYJYCDRJLFSYZFSLLCQYQFGJYHYXZLYLMSTDJCYHBZ";

listJP << "LLNWLXXYGYYHSMGDHXXHHLZZJZXCZZZCYQZFNGWPYLCPKPYYPMCLQKDGXZGGWQBDXZZKZFBXXLZXJTPJPTTBYTSZZDWSLCHZHSLT";

listJP << "YXHQLHYXXXYYZYSWTXZKHLXZXZPYHGCHKCFSYHUTJRLXFJXPTZTWHPLYXFCRHXSHXKYXXYHZQDXQWULHYHMJTBFLKHTXCWHJFWJC";

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "JZCZNNLMZHHARQMPGWQTZMXXMLLHGDZXYHXKYXYCJMFFYYHJFSBSSQLXXNDYCANNMTCJCYPRRNYTYQNYYMBMSXNDLYLYSLJRLXYS";

listJP << "XQMLLYZLZJJJKYZZCSFBZXXMSTBJGNXYZHLXNMCWSCYZYFZLXBRNNNYLBNRTGZQYSATSWRYHYJZMZDHZGZDWYBSSCSKXSYHYTXXG";

listJP << "CQGXZZSHYXJSCRHMKKBXCZJYJYMKQHZJFNBHMQHYSNJNZYBKNQMCLGQHWLZNZSWXKHLJHYYBQLBFCDSXDLDSPFZPSKJYZWZXZDDX";

listJP << "JSMMEGJSCSSMGCLXXKYYYLNYPWWWGYDKZJGGGZGGSYCKNJWNJPCXBJJTQTJWDSSPJXZXNZXUMELPXFSXTLLXCLJXJJLJZXCTPSWX";

listJP << "LYDHLYQRWHSYCSQYYBYAYWJJJQFWQCQQCJQGXALDBZZYJGKGXPLTZYFXJLTPADKYQHPMATLCPDCKBMTXYBHKLENXDLEEGQDYMSAW";

listJP << "HZMLJTWYGXLYQZLJEEYYBQQFFNLYXRDSCTGJGXYYNKLLYQKCCTLHJLQMKKZGCYYGLLLJDZGYDHZWXPYSJBZKDZGYZZHYWYFQYTYZ";

listJP << "SZYEZZLYMHJJHTSMQWYZLKYYWZCSRKQYTLTDXWCTYJKLWSQZWBDCQYNCJSRSZJLKCDCDTLZZZACQQZZDDXYPLXZBQJYLZLLLQDDZ";

listJP << "QJYJYJZYXNYYYNYJXKXDAZWYRDLJYYYRJLXLLDYXJCYWYWNQCCLDDNYYYNYCKCZHXXCCLGZQJGKWPPCQQJYSBZZXYJSQPXJPZBSB";

listJP << "DSFNSFPZXHDWZTDWPPTFLZZBZDMYYPQJRSDZSQZSQXBDGCPZSWDWCSQZGMDHZXMWWFYBPDGPHTMJTHZSMMBGZMBZJCFZWFZBBZMQ";

listJP << "CFMBDMCJXLGPNJBBXGYHYYJGPTZGZMQBQTCGYXJXLWZKYDPDYMGCFTPFXYZTZXDZXTGKMTYBBCLBJASKYTSSQYYMSZXFJEWLXLLS";

listJP << "ZBQJJJAKLYLXLYCCTSXMCWFKKKBSXLLLLJYXTYLTJYYTDPJHNHNNKBYQNFQYYZBYYESSESSGDYHFHWTCJBSDZZTFDMXHCNJZYMQW";

listJP << "SRYJDZJQPDQBBSTJGGFBKJBXTGQHNGWJXJGDLLTHZHHYYYYYYSXWTYYYCCBDBPYPZYCCZYJPZYWCBDLFWZCWJDXXHYHLHWZZXJTC";

listJP << "ZLCDPXUJCZZZLYXJJTXPHFXWPYWXZPTDZZBDZCYHJHMLXBQXSBYLRDTGJRRCTTTHYTCZWMXFYTWWZCWJWXJYWCSKYBZSCCTZQNHX";

listJP << "NWXXKHKFHTSWOCCJYBCMPZZYKBNNZPBZHHZDLSYDDYTYFJPXYNGFXBYQXCBHXCPSXTYZDMKYSNXSXLHKMZXLYHDHKWHXXSSKQYHH";

listJP << "CJYXGLHZXCSNHEKDTGZXQYPKDHEXTYKCNYMYYYPKQYYYKXZLTHJQTBYQHXBMYHSQCKWWYLLHCYYLNNEQXQWMCFBDCCMLJGGXDQKT";

listJP << "LXKGNQCDGZJWYJJLYHHQTTTNWCHMXCXWHWSZJYDJCCDBQCDGDNYXZTHCQRXCBHZTQCBXWGQWYYBXHMBYMYQTYEXMQKYAQYRGYZSL";

listJP << "FYKKQHYSSQYSHJGJCNXKZYCXSBXYXHYYLSTYCXQTHYSMGSCPMMGCCCCCMTZTASMGQZJHKLOSQYLSWTMXSYQKDZLJQQYPLSYCZTCQ";

listJP << "QPBBQJZCLPKHQZYYXXDTDDTSJCXFFLLCHQXMJLWCJCXTSPYCXNDTJSHJWXDQQJSKXYAMYLSJHMLALYKXCYYDMNMDQMXMCZNNCYBZ";

listJP << "KKYFLMCHCMLHXRCJJHSYLNMTJZGZGYWJXSRXCWJGJQHQZDQJDCJJZKJKGDZQGJJYJYLXZXXCDQHHHEYTMHLFSBDJSYYSHFYSTCZQ";

listJP << "LPBDRFRZTZYKYWHSZYQKWDQZRKMSYNBCRXQBJYFAZPZZEDZCJYWBCJWHYJBQSZYWRYSZPTDKZPFPBNZTKLQYHBBZPNPPTYZZYBQN";

listJP << "YDCPJMMCYCQMCYFZZDCMNLFPBPLNGQJTBTTNJZPZBBZNJKLJQYLNBZQHKSJZNGGQSZZKYXSHPZSNBCGZKDDZQANZHJKDRTLZLSWJ";

listJP << "LJZLYWTJNDJZJHXYAYNCBGTZCSSQMNJPJYTYSWXZFKWJQTKHTZPLBHSNJZSYZBWZZZZLSYLSBJHDWWQPSLMMFBJDWAQYZTCJTBNN";

listJP << "WZXQXCDSLQGDSDPDZHJTQQPSWLYYJZLGYXYZLCTCBJTKTYCZJTQKBSJLGMGZDMCSGPYNJZYQYYKNXRPWSZXMTNCSZZYXYBYHYZAX";

listJP << "YWQCJTLLCKJJTJHGDXDXYQYZZBYWDLWQCGLZGJGQRQZCZSSBCRPCSKYDZNXJSQGXSSJMYDNSTZTPBDLTKZWXQWQTZEXNQCZGWEZK";

listJP << "SSBYBRTSSSLCCGBPSZQSZLCCGLLLZXHZQTHCZMQGYZQZNMCOCSZJMMZSQPJYGQLJYJPPLDXRGZYXCCSXHSHGTZNLZWZKJCXTCFCJ";

listJP << "XLBMQBCZZWPQDNHXLJCTHYZLGYLNLSZZPCXDSCQQHJQKSXZPBAJYEMSMJTZDXLCJYRYYNWJBNGZZTMJXLTBSLYRZPYLSSCNXPHLL";

listJP << "HYLLQQZQLXYMRSYCXZLMMCZLTZSDWTJJLLNZGGQXPFSKYGYGHBFZPDKMWGHCXMSGDXJMCJZDYCABXJDLNBCDQYGSKYDQTXDJJYXM";

listJP << "SZQAZDZFSLQXYJSJZYLBTXXWXQQZBJZUFBBLYLWDSLJHXJYZJWTDJCZFQZQZZDZSXZZQLZCDZFJHYSPYMPQZMLPPLFFXJJNZZYLS";

listJP << "JEYQZFPFZKSYWJJJHRDJZZXTXXGLGHYDXCSKYSWMMZCWYBAZBJKSHFHJCXMHFQHYXXYZFTSJYZFXYXPZLCHMZMBXHZZSXYFYMNCW";

listJP << "DABAZLXKTCSHHXKXJJZJSTHYGXSXYYHHHJWXKZXSSBZZWHHHCWTZZZPJXSNXQQJGZYZYWLLCWXZFXXYXYHXMKYYSWSQMNLNAYCYS";

listJP << "PMJKHWCQHYLAJJMZXHMMCNZHBHXCLXTJPLTXYJHDYYLTTXFSZHYXXSJBJYAYRSMXYPLCKDUYHLXRLNLLSTYZYYQYGYHHSCCSMZCT";

listJP << "ZQXKYQFPYYRPFFLKQUNTSZLLZMWWTCQQYZWTLLMLMPWMBZSSTZRBPDDTLQJJBXZCSRZQQYGWCSXFWZLXCCRSZDZMCYGGDZQSGTJS";

listJP << "WLJMYMMZYHFBJDGYXCCPSHXNZCSBSJYJGJMPPWAFFYFNXHYZXZYLREMZGZCYZSSZDLLJCSQFNXZKPTXZGXJJGFMYYYSNBTYLBNLH";

listJP << "PFZDCYFBMGQRRSSSZXYSGTZRNYDZZCDGPJAFJFZKNZBLCZSZPSGCYCJSZLMLRSZBZZLDLSLLYSXSQZQLYXZLSKKBRXBRBZCYCXZZ";

listJP << "ZEEYFGKLZLYYHGZSGZLFJHGTGWKRAAJYZKZQTSSHJJXDCYZUYJLZYRZDQQHGJZXSSZBYKJPBFRTJXLLFQWJHYLQTYMBLPZDXTZYG";

listJP << "BDHZZRBGXHWNJTJXLKSCFSMWLSDQYSJTXKZSCFWJLBXFTZLLJZLLQBLSQMQQCGCZFPBPHZCZJLPYYGGDTGWDCFCZQYYYQYSSCLXZ";

listJP << "SKLZZZGFFCQNWGLHQYZJJCZLQZZYJPJZZBPDCCMHJGXDQDGDLZQMFGPSYTSDYFWWDJZJYSXYYCZCYHZWPBYKXRYLYBHKJKSFXTZJ";

listJP << "MMCKHLLTNYYMSYXYZPYJQYCSYCWMTJJKQYRHLLQXPSGTLYYCLJSCPXJYZFNMLRGJJTYZBXYZMSJYJHHFZQMSYXRSZCWTLRTQZSST";

listJP << "KXGQKGSPTGCZNJSJCQCXHMXGGZTQYDJKZDLBZSXJLHYQGGGTHQSZPYHJHHGYYGKGGCWJZZYLCZLXQSFTGZSLLLMLJSKCTBLLZZSZ";

listJP << "MMNYTPZSXQHJCJYQXYZXZQZCPSHKZZYSXCDFGMWQRLLQXRFZTLYSTCTMJCXJJXHJNXTNRZTZFQYHQGLLGCXSZSJDJLJCYDSJTLNY";

listJP << "XHSZXCGJZYQPYLFHDJSBPCCZHJJJQZJQDYBSSLLCMYTTMQTBHJQNNYGKYRQYQMZGCJKPDCGMYZHQLLSLLCLMHOLZGDYYFZSLJCQZ";

listJP << "LYLZQJESHNYLLJXGJXLYSYYYXNBZLJSSZCQQCJYLLZLTJYLLZLLBNYLGQCHXYYXOXCXQKYJXXXYKLXSXXYQXCYKQXQCSGYXXYQXY";

listJP << "GYTQOHXHXPYXXXULCYEYCHZZCBWQBBWJQZSCSZSSLZYLKDESJZWMYMCYTSDSXXSCJPQQSQYLYYZYCMDJDZYWCBTJSYDJKCYDDJLB";

listJP << "DJJSODZYSYXQQYXDHHGQQYQHDYXWGMMMAJDYBBBPPBCMUUPLJZSMTXERXJMHQNUTPJDCBSSMSSSTKJTSSMMTRCPLZSZMLQDSDMJM";

listJP << "QPNQDXCFYNBFSDQXYXHYAYKQYDDLQYYYSSZBYDSLNTFQTZQPZMCHDHCZCWFDXTMYQSPHQYYXSRGJCWTJTZZQMGWJJTJHTQJBBHWZ";

listJP << "PXXHYQFXXQYWYYHYSCDYDHHQMNMTMWCPBSZPPZZGLMZFOLLCFWHMMSJZTTDHZZYFFYTZZGZYSKYJXQYJZQBHMBZZLYGHGFMSHPZF";

listJP << "ZSNCLPBQSNJXZSLXXFPMTYJYGBXLLDLXPZJYZJYHHZCYWHJYLSJEXFSZZYWXKZJLUYDTMLYMQJPWXYHXSKTQJEZRPXXZHHMHWQPW";

listJP << "QLYJJQJJZSZCPHJLCHHNXJLQWZJHBMZYXBDHHYPZLHLHLGFWLCHYYTLHJXCJMSCPXSTKPNHQXSRTYXXTESYJCTLSSLSTDLLLWWYH";

listJP << "DHRJZSFGXTSYCZYNYHTDHWJSLHTZDQDJZXXQHGYLTZPHCSQFCLNJTCLZPFSTPDYNYLGMJLLYCQHYSSHCHYLHQYQTMZYPBYWRFQYK";

listJP << "QSYSLZDQJMPXYYSSRHZJNYWTQDFZBWWTWWRXCWHGYHXMKMYYYQMSMZHNGCEPMLQQMTCWCTMMPXJPJJHFXYYZSXZHTYBMSTSYJTTQ";

listJP << "QQYYLHYNPYQZLCYZHZWSMYLKFJXLWGXYPJYTYSYXYMZCKTTWLKSMZSYLMPWLZWXWQZSSAQSYXYRHSSNTSRAPXCPWCMGDXHXZDZYF";

listJP << "JHGZTTSBJHGYZSZYSMYCLLLXBTYXHBBZJKSSDMALXHYCFYGMQYPJYCQXJLLLJGSLZGQLYCJCCZOTYXMTMTTLLWTGPXYMZMKLPSZZ";

listJP << "ZXHKQYSXCTYJZYHXSHYXZKXLZWPSQPYHJWPJPWXQQYLXSDHMRSLZZYZWTTCYXYSZZSHBSCCSTPLWSSCJCHNLCGCHSSPHYLHFHHXJ";

listJP << "SXYLLNYLSZDHZXYLSXLWZYKCLDYAXZCMDDYSPJTQJZLNWQPSSSWCTSTSZLBLNXSMNYYMJQBQHRZWTYYDCHQLXKPZWBGQYBKFCMZW";

listJP << "PZLLYYLSZYDWHXPSBCMLJBSCGBHXLQHYRLJXYSWXWXZSLDFHLSLYNJLZYFLYJYCDRJLFSYZFSLLCQYQFGJYHYXZLYLMSTDJCYHBZ";

listJP << "LLNWLXXYGYYHSMGDHXXHHLZZJZXCZZZCYQZFNGWPYLCPKPYYPMCLQKDGXZGGWQBDXZZKZFBXXLZXJTPJPTTBYTSZZDWSLCHZHSLT";

listJP << "YXHQLHYXXXYYZYSWTXZKHLXZXZPYHGCHKCFSYHUTJRLXFJXPTZTWHPLYXFCRHXSHXKYXXYHZQDXQWULHYHMJTBFLKHTXCWHJFWJC";

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "LPBDRFRZTZYKYWHSZYQKWDQZRKMSYNBCRXQBJYFAZPZZEDZCJYWBCJWHYJBQSZYWRYSZPTDKZPFPBNZTKLQYHBBZPNPPTYZZYBQN";

listJP << "YDCPJMMCYCQMCYFZZDCMNLFPBPLNGQJTBTTNJZPZBBZNJKLJQYLNBZQHKSJZNGGQSZZKYXSHPZSNBCGZKDDZQANZHJKDRTLZLSWJ";

listJP << "LJZLYWTJNDJZJHXYAYNCBGTZCSSQMNJPJYTYSWXZFKWJQTKHTZPLBHSNJZSYZBWZZZZLSYLSBJHDWWQPSLMMFBJDWAQYZTCJTBNN";

listJP << "WZXQXCDSLQGDSDPDZHJTQQPSWLYYJZLGYXYZLCTCBJTKTYCZJTQKBSJLGMGZDMCSGPYNJZYQYYKNXRPWSZXMTNCSZZYXYBYHYZAX";

listJP << "YWQCJTLLCKJJTJHGDXDXYQYZZBYWDLWQCGLZGJGQRQZCZSSBCRPCSKYDZNXJSQGXSSJMYDNSTZTPBDLTKZWXQWQTZEXNQCZGWEZK";

listJP << "SSBYBRTSSSLCCGBPSZQSZLCCGLLLZXHZQTHCZMQGYZQZNMCOCSZJMMZSQPJYGQLJYJPPLDXRGZYXCCSXHSHGTZNLZWZKJCXTCFCJ";

listJP << "XLBMQBCZZWPQDNHXLJCTHYZLGYLNLSZZPCXDSCQQHJQKSXZPBAJYEMSMJTZDXLCJYRYYNWJBNGZZTMJXLTBSLYRZPYLSSCNXPHLL";

listJP << "HYLLQQZQLXYMRSYCXZLMMCZLTZSDWTJJLLNZGGQXPFSKYGYGHBFZPDKMWGHCXMSGDXJMCJZDYCABXJDLNBCDQYGSKYDQTXDJJYXM";

listJP << "SZQAZDZFSLQXYJSJZYLBTXXWXQQZBJZUFBBLYLWDSLJHXJYZJWTDJCZFQZQZZDZSXZZQLZCDZFJHYSPYMPQZMLPPLFFXJJNZZYLS";

listJP << "JEYQZFPFZKSYWJJJHRDJZZXTXXGLGHYDXCSKYSWMMZCWYBAZBJKSHFHJCXMHFQHYXXYZFTSJYZFXYXPZLCHMZMBXHZZSXYFYMNCW";

listJP << "DABAZLXKTCSHHXKXJJZJSTHYGXSXYYHHHJWXKZXSSBZZWHHHCWTZZZPJXSNXQQJGZYZYWLLCWXZFXXYXYHXMKYYSWSQMNLNAYCYS";

listJP << "PMJKHWCQHYLAJJMZXHMMCNZHBHXCLXTJPLTXYJHDYYLTTXFSZHYXXSJBJYAYRSMXYPLCKDUYHLXRLNLLSTYZYYQYGYHHSCCSMZCT";

listJP << "ZQXKYQFPYYRPFFLKQUNTSZLLZMWWTCQQYZWTLLMLMPWMBZSSTZRBPDDTLQJJBXZCSRZQQYGWCSXFWZLXCCRSZDZMCYGGDZQSGTJS";

listJP << "WLJMYMMZYHFBJDGYXCCPSHXNZCSBSJYJGJMPPWAFFYFNXHYZXZYLREMZGZCYZSSZDLLJCSQFNXZKPTXZGXJJGFMYYYSNBTYLBNLH";

listJP << "PFZDCYFBMGQRRSSSZXYSGTZRNYDZZCDGPJAFJFZKNZBLCZSZPSGCYCJSZLMLRSZBZZLDLSLLYSXSQZQLYXZLSKKBRXBRBZCYCXZZ";

listJP << "ZEEYFGKLZLYYHGZSGZLFJHGTGWKRAAJYZKZQTSSHJJXDCYZUYJLZYRZDQQHGJZXSSZBYKJPBFRTJXLLFQWJHYLQTYMBLPZDXTZYG";

listJP << "BDHZZRBGXHWNJTJXLKSCFSMWLSDQYSJTXKZSCFWJLBXFTZLLJZLLQBLSQMQQCGCZFPBPHZCZJLPYYGGDTGWDCFCZQYYYQYSSCLXZ";

listJP << "SKLZZZGFFCQNWGLHQYZJJCZLQZZYJPJZZBPDCCMHJGXDQDGDLZQMFGPSYTSDYFWWDJZJYSXYYCZCYHZWPBYKXRYLYBHKJKSFXTZJ";

listJP << "MMCKHLLTNYYMSYXYZPYJQYCSYCWMTJJKQYRHLLQXPSGTLYYCLJSCPXJYZFNMLRGJJTYZBXYZMSJYJHHFZQMSYXRSZCWTLRTQZSST";

listJP << "KXGQKGSPTGCZNJSJCQCXHMXGGZTQYDJKZDLBZSXJLHYQGGGTHQSZPYHJHHGYYGKGGCWJZZYLCZLXQSFTGZSLLLMLJSKCTBLLZZSZ";

listJP << "MMNYTPZSXQHJCJYQXYZXZQZCPSHKZZYSXCDFGMWQRLLQXRFZTLYSTCTMJCXJJXHJNXTNRZTZFQYHQGLLGCXSZSJDJLJCYDSJTLNY";

listJP << "XHSZXCGJZYQPYLFHDJSBPCCZHJJJQZJQDYBSSLLCMYTTMQTBHJQNNYGKYRQYQMZGCJKPDCGMYZHQLLSLLCLMHOLZGDYYFZSLJCQZ";

listJP << "LYLZQJESHNYLLJXGJXLYSYYYXNBZLJSSZCQQCJYLLZLTJYLLZLLBNYLGQCHXYYXOXCXQKYJXXXYKLXSXXYQXCYKQXQCSGYXXYQXY";

listJP << "GYTQOHXHXPYXXXULCYEYCHZZCBWQBBWJQZSCSZSSLZYLKDESJZWMYMCYTSDSXXSCJPQQSQYLYYZYCMDJDZYWCBTJSYDJKCYDDJLB";

listJP << "DJJSODZYSYXQQYXDHHGQQYQHDYXWGMMMAJDYBBBPPBCMUUPLJZSMTXERXJMHQNUTPJDCBSSMSSSTKJTSSMMTRCPLZSZMLQDSDMJM";

listJP << "QPNQDXCFYNBFSDQXYXHYAYKQYDDLQYYYSSZBYDSLNTFQTZQPZMCHDHCZCWFDXTMYQSPHQYYXSRGJCWTJTZZQMGWJJTJHTQJBBHWZ";

listJP << "PXXHYQFXXQYWYYHYSCDYDHHQMNMTMWCPBSZPPZZGLMZFOLLCFWHMMSJZTTDHZZYFFYTZZGZYSKYJXQYJZQBHMBZZLYGHGFMSHPZF";

listJP << "ZSNCLPBQSNJXZSLXXFPMTYJYGBXLLDLXPZJYZJYHHZCYWHJYLSJEXFSZZYWXKZJLUYDTMLYMQJPWXYHXSKTQJEZRPXXZHHMHWQPW";

listJP << "QLYJJQJJZSZCPHJLCHHNXJLQWZJHBMZYXBDHHYPZLHLHLGFWLCHYYTLHJXCJMSCPXSTKPNHQXSRTYXXTESYJCTLSSLSTDLLLWWYH";

listJP << "DHRJZSFGXTSYCZYNYHTDHWJSLHTZDQDJZXXQHGYLTZPHCSQFCLNJTCLZPFSTPDYNYLGMJLLYCQHYSSHCHYLHQYQTMZYPBYWRFQYK";

listJP << "QSYSLZDQJMPXYYSSRHZJNYWTQDFZBWWTWWRXCWHGYHXMKMYYYQMSMZHNGCEPMLQQMTCWCTMMPXJPJJHFXYYZSXZHTYBMSTSYJTTQ";

listJP << "QQYYLHYNPYQZLCYZHZWSMYLKFJXLWGXYPJYTYSYXYMZCKTTWLKSMZSYLMPWLZWXWQZSSAQSYXYRHSSNTSRAPXCPWCMGDXHXZDZYF";

listJP << "JHGZTTSBJHGYZSZYSMYCLLLXBTYXHBBZJKSSDMALXHYCFYGMQYPJYCQXJLLLJGSLZGQLYCJCCZOTYXMTMTTLLWTGPXYMZMKLPSZZ";

listJP << "ZXHKQYSXCTYJZYHXSHYXZKXLZWPSQPYHJWPJPWXQQYLXSDHMRSLZZYZWTTCYXYSZZSHBSCCSTPLWSSCJCHNLCGCHSSPHYLHFHHXJ";

listJP << "SXYLLNYLSZDHZXYLSXLWZYKCLDYAXZCMDDYSPJTQJZLNWQPSSSWCTSTSZLBLNXSMNYYMJQBQHRZWTYYDCHQLXKPZWBGQYBKFCMZW";

listJP << "PZLLYYLSZYDWHXPSBCMLJBSCGBHXLQHYRLJXYSWXWXZSLDFHLSLYNJLZYFLYJYCDRJLFSYZFSLLCQYQFGJYHYXZLYLMSTDJCYHBZ";

listJP << "LLNWLXXYGYYHSMGDHXXHHLZZJZXCZZZCYQZFNGWPYLCPKPYYPMCLQKDGXZGGWQBDXZZKZFBXXLZXJTPJPTTBYTSZZDWSLCHZHSLT";

listJP << "YXHQLHYXXXYYZYSWTXZKHLXZXZPYHGCHKCFSYHUTJRLXFJXPTZTWHPLYXFCRHXSHXKYXXYHZQDXQWULHYHMJTBFLKHTXCWHJFWJC";

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "KXGQKGSPTGCZNJSJCQCXHMXGGZTQYDJKZDLBZSXJLHYQGGGTHQSZPYHJHHGYYGKGGCWJZZYLCZLXQSFTGZSLLLMLJSKCTBLLZZSZ";

listJP << "MMNYTPZSXQHJCJYQXYZXZQZCPSHKZZYSXCDFGMWQRLLQXRFZTLYSTCTMJCXJJXHJNXTNRZTZFQYHQGLLGCXSZSJDJLJCYDSJTLNY";

listJP << "XHSZXCGJZYQPYLFHDJSBPCCZHJJJQZJQDYBSSLLCMYTTMQTBHJQNNYGKYRQYQMZGCJKPDCGMYZHQLLSLLCLMHOLZGDYYFZSLJCQZ";

listJP << "LYLZQJESHNYLLJXGJXLYSYYYXNBZLJSSZCQQCJYLLZLTJYLLZLLBNYLGQCHXYYXOXCXQKYJXXXYKLXSXXYQXCYKQXQCSGYXXYQXY";

listJP << "GYTQOHXHXPYXXXULCYEYCHZZCBWQBBWJQZSCSZSSLZYLKDESJZWMYMCYTSDSXXSCJPQQSQYLYYZYCMDJDZYWCBTJSYDJKCYDDJLB";

listJP << "DJJSODZYSYXQQYXDHHGQQYQHDYXWGMMMAJDYBBBPPBCMUUPLJZSMTXERXJMHQNUTPJDCBSSMSSSTKJTSSMMTRCPLZSZMLQDSDMJM";

listJP << "QPNQDXCFYNBFSDQXYXHYAYKQYDDLQYYYSSZBYDSLNTFQTZQPZMCHDHCZCWFDXTMYQSPHQYYXSRGJCWTJTZZQMGWJJTJHTQJBBHWZ";

listJP << "PXXHYQFXXQYWYYHYSCDYDHHQMNMTMWCPBSZPPZZGLMZFOLLCFWHMMSJZTTDHZZYFFYTZZGZYSKYJXQYJZQBHMBZZLYGHGFMSHPZF";

listJP << "ZSNCLPBQSNJXZSLXXFPMTYJYGBXLLDLXPZJYZJYHHZCYWHJYLSJEXFSZZYWXKZJLUYDTMLYMQJPWXYHXSKTQJEZRPXXZHHMHWQPW";

listJP << "QLYJJQJJZSZCPHJLCHHNXJLQWZJHBMZYXBDHHYPZLHLHLGFWLCHYYTLHJXCJMSCPXSTKPNHQXSRTYXXTESYJCTLSSLSTDLLLWWYH";

listJP << "DHRJZSFGXTSYCZYNYHTDHWJSLHTZDQDJZXXQHGYLTZPHCSQFCLNJTCLZPFSTPDYNYLGMJLLYCQHYSSHCHYLHQYQTMZYPBYWRFQYK";

listJP << "QSYSLZDQJMPXYYSSRHZJNYWTQDFZBWWTWWRXCWHGYHXMKMYYYQMSMZHNGCEPMLQQMTCWCTMMPXJPJJHFXYYZSXZHTYBMSTSYJTTQ";

listJP << "QQYYLHYNPYQZLCYZHZWSMYLKFJXLWGXYPJYTYSYXYMZCKTTWLKSMZSYLMPWLZWXWQZSSAQSYXYRHSSNTSRAPXCPWCMGDXHXZDZYF";

listJP << "JHGZTTSBJHGYZSZYSMYCLLLXBTYXHBBZJKSSDMALXHYCFYGMQYPJYCQXJLLLJGSLZGQLYCJCCZOTYXMTMTTLLWTGPXYMZMKLPSZZ";

listJP << "ZXHKQYSXCTYJZYHXSHYXZKXLZWPSQPYHJWPJPWXQQYLXSDHMRSLZZYZWTTCYXYSZZSHBSCCSTPLWSSCJCHNLCGCHSSPHYLHFHHXJ";

listJP << "SXYLLNYLSZDHZXYLSXLWZYKCLDYAXZCMDDYSPJTQJZLNWQPSSSWCTSTSZLBLNXSMNYYMJQBQHRZWTYYDCHQLXKPZWBGQYBKFCMZW";

listJP << "PZLLYYLSZYDWHXPSBCMLJBSCGBHXLQHYRLJXYSWXWXZSLDFHLSLYNJLZYFLYJYCDRJLFSYZFSLLCQYQFGJYHYXZLYLMSTDJCYHBZ";

listJP << "LLNWLXXYGYYHSMGDHXXHHLZZJZXCZZZCYQZFNGWPYLCPKPYYPMCLQKDGXZGGWQBDXZZKZFBXXLZXJTPJPTTBYTSZZDWSLCHZHSLT";

listJP << "YXHQLHYXXXYYZYSWTXZKHLXZXZPYHGCHKCFSYHUTJRLXFJXPTZTWHPLYXFCRHXSHXKYXXYHZQDXQWULHYHMJTBFLKHTXCWHJFWJC";

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "FPQRYQXCYYYQYGRPYWSGSUNGWCHKZDXYFLXXHJJBYZWTSXXNCYJJYMSWZJQRMHXZWFQSYLZJZGBHYNSLBGTTCSYBYXXWXYHXYYXN";

listJP << "SQYXMQYWRGYQLXBBZLJSYLPSYTJZYHYZAWLRORJMKSCZJXXXYXCHDYXRYXXJDTSQFXLYLTSFFYXLMTYJMJUYYYXLTZCSXQZQHZXL";

listJP << "YYXZHDNBRXXXJCTYHLBRLMBRLLAXKYLLLJLYXXLYCRYLCJTGJCMTLZLLCYZZPZPCYAWHJJFYBDYYZSMPCKZDQYQPBPCJPDCYZMDP";

listJP << "BCYYDYCNNPLMTMLRMFMMGWYZBSJGYGSMZQQQZTXMKQWGXLLPJGZBQCDJJJFPKJKCXBLJMSWMDTQJXLDLPPBXCWRCQFBFQJCZAHZG";

listJP << "MYKPHYYHZYKNDKZMBPJYXPXYHLFPNYYGXJDBKXNXHJMZJXSTRSTLDXSKZYSYBZXJLXYSLBZYSLHXJPFXPQNBYLLJQKYGZMCYZZYM";

listJP << "CCSLCLHZFWFWYXZMWSXTYNXJHPYYMCYSPMHYSMYDYSHQYZCHMJJMZCAAGCFJBBHPLYZYLXXSDJGXDHKXXTXXNBHRMLYJSLTXMRHN";

listJP << "LXQJXYZLLYSWQGDLBJHDCGJYQYCMHWFMJYBMBYJYJWYMDPWHXQLDYGPDFXXBCGJSPCKRSSYZJMSLBZZJFLJJJLGXZGYXYXLSZQYX";

listJP << "BEXYXHGCXBPLDYHWETTWWCJMBTXCHXYQXLLXFLYXLLJLSSFWDPZSMYJCLMWYTCZPCHQEKCQBWLCQYDPLQPPQZQFJQDJHYMMCXTXD";

listJP << "RMJWRHXCJZYLQXDYYNHYYHRSLSRSYWWZJYMTLTLLGTQCJZYABTCKZCJYCCQLJZQXALMZYHYWLWDXZXQDLLQSHGPJFJLJHJABCQZD";

listJP << "JGTKHSSTCYJLPSWZLXZXRWGLDLZRLZXTGSLLLLZLYXXWGDZYGBDPHZPBRLWSXQBPFDWOFMWHLYPCBJCCLDMBZPBZZLCYQXLDOMZB";

listJP << "LZWPDWYYGDSTTHCSQSCCRSSSYSLFYBFNTYJSZDFNDPDHDZZMBBLSLCMYFFGTJJQWFTMTPJWFNLBZCMMJTGBDZLQLPYFHYYMJYLSD";

listJP << "CHDZJWJCCTLJCLDTLJJCPDDSQDSSZYBNDBJLGGJZXSXNLYCYBJXQYCBYLZCFZPPGKCXZDZFZTJJFJSJXZBNZYJQTTYJYHTYCZHYM";

listJP << "DJXTTMPXSPLZCDWSLSHXYPZGTFMLCJTYCBPMGDKWYCYZCDSZZYHFLYCTYGWHKJYYLSJCXGYWJCBLLCSNDDBTZBSCLYZCZZSSQDLL";

listJP << "MQYYHFSLQLLXFTYHABXGWNYWYYPLLSDLDLLBJCYXJZMLHLJDXYYQYTDLLLBUGBFDFBBQJZZMDPJHGCLGMJJPGAEHHBWCQXAXHHHZ";

listJP << "CHXYPHJAXHLPHJPGPZJQCQZGJJZZUZDMQYYBZZPHYHYBWHAZYJHYKFGDPFQSDLZMLJXKXGALXZDAGLMDGXMWZQYXXDXXPFDMMSSY";

listJP << "MPFMDMMKXKSYZYSHDZKXSYSMMZZZMSYDNZZCZXFPLSTMZDNMXCKJMZTYYMZMZZMSXHHDCZJEMXXKLJSTLWLSQLYJZLLZJSSDPPMH";

listJP << "NLZJCZYHMXXHGZCJMDHXTKGRMXFWMCGMWKDTKSXQMMMFZZYDKMSCLCMPCGMHSPXQPZDSSLCXKYXTWLWJYAHZJGZQMCSNXYYMMPML";

listJP << "KJXMHLMLQMXCTKZMJQYSZJSYSZHSYJZJCDAJZYBSDQJZGWZQQXFKDMSDJLFWEHKZQKJPEYPZYSZCDWYJFFMZZYLTTDZZEFMZLBNP";

listJP << "PLPLPEPSZALLTYLKCKQZKGENQLWAGYXYDPXLHSXQQWQCQXQCLHYXXMLYCCWLYMQYSKGCHLCJNSZKPYZKCQZQLJPDMDZHLASXLBYD";

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "WQLWDNBQCRYDDZTJYBKBWSZDXDTNPJDTCTQDFXQQMGNXECLTTBKPWSLCTYQLPWYZZKLPYGZCQQPLLKCCYLPQMZCZQCLJSLQZDJXL";

listJP << "DDHPZQDLJJXZQDXYZQKZLJCYQDYJPPYPQYKJYRMPCBYMCXKLLZLLFQPYLLLMBSGLCYSSLRSYSQTMXYXZQZFDZUYSYZTFFMZZSMZQ";

listJP << "HZSSCCMLYXWTPZGXZJGZGSJSGKDDHTQGGZLLBJDZLCBCHYXYZHZFYWXYZYMSDBZZYJGTSMTFXQYXQSTDGSLNXDLRYZZLRYYLXQHT";

listJP << "XSRTZNGZXBNQQZFMYKMZJBZYMKBPNLYZPBLMCNQYZZZSJZHJCTZKHYZZJRDYZHNPXGLFZTLKGJTCTSSYLLGZRZBBQZZKLPKLCZYS";

listJP << "SUYXBJFPNJZZXCDWXZYJXZZDJJKGGRSRJKMSMZJLSJYWQSKYHQJSXPJZZZLSNSHRNYPZTWCHKLPSRZLZXYJQXQKYSJYCZTLQZYBB";

listJP << "YBWZPQDWWYZCYTJCJXCKCWDKKZXSGKDZXWWYYJQYYTCYTDLLXWKCZKKLCCLZCQQDZLQLCSFQCHQHSFSMQZZLNBJJZBSJHTSZDYSJ";

listJP << "QJPDLZCDCWJKJZZLPYCGMZWDJJBSJQZSYZYHHXJPBJYDSSXDZNCGLQMBTSFSBPDZDLZNFGFJGFSMPXJQLMBLGQCYYXBQKDJJQYRF";

listJP << "KZTJDHCZKLBSDZCFJTPLLJGXHYXZCSSZZXSTJYGKGCKGYOQXJPLZPBPGTGYJZGHZQZZLBJLSQFZGKQQJZGYCZBZQTLDXRJXBSXXP";

listJP << "ZXHYZYCLWDXJJHXMFDZPFZHQHQMQGKSLYHTYCGFRZGNQXCLPDLBZCSCZQLLJBLHBZCYPZZPPDYMZZSGYHCKCPZJGSLJLNSCDSLDL";

listJP << "XBMSTLDDFJMKDJDHZLZXLSZQPQPGJLLYBDSZGQLBZLSLKYYHZTTNTJYQTZZPSZQZTLLJTYYLLQLLQYZQLBDZLSLYYZYMDFSZSNHL";

listJP << "XZNCZQZPBWSKRFBSYZMTHBLGJPMCZZLSTLXSHTCSYZLZBLFEQHLXFLCJLYLJQCBZLZJHHSSTBRMHXZHJZCLXFNBGXGTQJCZTMSFZ";

listJP << "KJMSSNXLJKBHSJXNTNLZDNTLMSJXGZJYJCZXYJYJWRWWQNZTNFJSZPZSHZJFYRDJSFSZJZBJFZQZZHZLXFYSBZQLZSGYFTZDCSZX";

listJP << "ZJBQMSZKJRHYJZCKMJKHCHGTXKXQGLXPXFXTRTYLXJXHDTSJXHJZJXZWZLCQSBTXWXGXTXXHXFTSDKFJHZYJFJXRZSDLLLTQSQQZ";

listJP << "QWZXSYQTWGWBZCGZLLYZBCLMQQTZHZXZXLJFRMYZFLXYSQXXJKXRMQDZDMMYYBSQBHGZMWFWXGMXLZPYYTGZYCCDXYZXYWGSYJYZ";

listJP << "NBHPZJSQSYXSXRTFYZGRHZTXSZZTHCBFCLSYXZLZQMZLMPLMXZJXSFLBYZMYQHXJSXRXSQZZZSSLYFRCZJRCRXHHZXQYDYHXSJJH";

listJP << "ZCXZBTYNSYSXJBQLPXZQPYMLXZKYXLXCJLCYSXXZZLXDLLLJJYHZXGYJWKJRWYHCPSGNRZLFZWFZZNSXGXFLZSXZZZBFCSYJDBRJ";

listJP << "KRDHHGXJLJJTGXJXXSTJTJXLYXQFCSGSWMSBCTLQZZWLZZKXJMLTMJYHSDDBXGZHDLBMYJFRZFSGCLYJBPMLYSMSXLSZJQQHJZFX";

listJP << "GFQFQBPXZGYYQXGZTCQWYLTLGWSGWHRLFSFGZJMGMGBGTJFSYZZGZYZAFLSSPMLPFLCWBJZCLJJMZLPJJLYMQDMYYYFBGYGYZMLY";

listJP << "ZDXQYXRQQQHSYYYQXYLJTYXFSFSLLGNQCYHYCWFHCCCFXPYLYPLLZYXXXXXKQHHXSHJZCFZSCZJXCPZWHHHHHAPYLQALPQAFYHXD";

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "YLUKMZQGGGDDESRNNZLTZGCHYPPYSQJJHCLLJTOLNJPZLJLHYMHEYDYDSQYCDDHGZUNDZCLZYZLLZNTNYZGSLHSLPJJBDGWXPCDU";

listJP << "TJCKLKCLWKLLCASSTKZZDNQNTTLYYZSSYSSZZRYLJQKCQDHHCRXRZYDGRGCWCGZQFFFPPJFZYNAKRGYWYQPQXXFKJTSZZXSWZDDF";

listJP << "BBXTBGTZKZNPZZPZXZPJSZBMQHKCYXYLDKLJNYPKYGHGDZJXXEAHPNZKZTZCMXCXMMJXNKSZQNMNLWBWWXJKYHCPSTMCSQTZJYXT";

listJP << "PCTPDTNNPGLLLZSJLSPBLPLQHDTNJNLYYRSZFFJFQWDPHZDWMRZCCLODAXNSSNYZRESTYJWJYJDBCFXNMWTTBYLWSTSZGYBLJPXG";

listJP << "LBOCLHPCBJLTMXZLJYLZXCLTPNCLCKXTPZJSWCYXSFYSZDKNTLBYJCYJLLSTGQCBXRYZXBXKLYLHZLQZLNZCXWJZLJZJNCJHXMNZ";

listJP << "ZGJZZXTZJXYCYYCXXJYYXJJXSSSJSTSSTTPPGQTCSXWZDCSYFPTFBFHFBBLZJCLZZDBXGCXLQPXKFZFLSYLTUWBMQJHSZBMDDBCY";

listJP << "SCCLDXYCDDQLYJJWMQLLCSGLJJSYFPYYCCYLTJANTJJPWYCMMGQYYSXDXQMZHSZXPFTWWZQSWQRFKJLZJQQYFBRXJHHFWJJZYQAZ";

listJP << "MYFRHCYYBYQWLPEXCCZSTYRLTTDMQLYKMBBGMYYJPRKZNPBSXYXBHYZDJDNGHPMFSGMWFZMFQMMBCMZZCJJLCNUXYQLMLRYGQZCY";

listJP << "XZLWJGCJCGGMCJNFYZZJHYCPRRCMTZQZXHFQGTJXCCJEAQCRJYHPLQLSZDJRBCQHQDYRHYLYXJSYMHZYDWLDFRYHBPYDTSSCNWBX";

listJP << "GLPZMLZZTQSSCPJMXXYCSJYTYCGHYCJWYRXXLFEMWJNMKLLSWTXHYYYNCMMCWJDQDJZGLLJWJRKHPZGGFLCCSCZMCBLTBHBQJXQD";

listJP << "SPDJZZGKGLFQYWBZYZJLTSTDHQHCTCBCHFLQMPWDSHYYTQWCNZZJTLBYMBPDYYYXSQKXWYYFLXXNCWCXYPMAELYKKJMZZZBRXYYQ";

listJP << "JFLJPFHHHYTZZXSGQQMHSPGDZQWBWPJHZJDYSCQWZKTXXSQLZYYMYSDZGRXCKKUJLWPYSYSCSYZLRMLQSYLJXBCXTLWDQZPCYCYK";

listJP << "PPPNSXFYZJJRCEMHSZMSXLXGLRWGCSTLRSXBZGBZGZTCPLUJLSLYLYMTXMTZPALZXPXJTJWTCYYZLBLXBZLQMYLXPGHDSLSSDMXM";

listJP << "BDZZSXWHAMLCZCPJMCNHJYSNSYGCHSKQMZZQDLLKABLWJXSFMOCDXJRRLYQZKJMYBYQLYHETFJZFRFKSRYXFJTWDSXXSYSQJYSLY";

listJP << "XWJHSNLXYYXHBHAWHHJZXWMYLJCSSLKYDZTXBZSYFDXGXZJKHSXXYBSSXDPYNZWRPTQZCZENYGCXQFJYKJBZMLJCMQQXUOXSLYXX";

listJP << "LYLLJDZBTYMHPFSTTQQWLHOKYBLZZALZXQLHZWRRQHLSTMYPYXJJXMQSJFNBXYXYJXXYQYLTHYLQYFMLKLJTMLLHSZWKZHLJMLHL";

listJP << "JKLJSTLQXYLMBHHLNLZXQJHXCFXXLHYHJJGBYZZKBXSCQDJQDSUJZYYHZHHMGSXCSYMXFEBCQWWRBPYYJQTYZCYQYQQZYHMWFFHG";

listJP << "ZFRJFCDPXNTQYZPDYKHJLFRZXPPXZDBBGZQSTLGDGYLCQMLCHHMFYWLZYXKJLYPQHSYWMQQGQZMLZJNSQXJQSYJYCBEHSXFSZPXZ";

listJP << "WFLLBCYYJDYTDTHWZSFJMQQYJLMQXXLLDTTKHHYBFPWTYYSQQWNQWLGWDEBZWCMYGCULKJXTMXMYJSXHYBRWFYMWFRXYQMXYSZTZ";

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

listJP << "ZTFYKMLDHQDXWYYNLCRYJBLPSXCXYWLSPRRJWXHQYPHTYDNXHHMMYWYTZCSQMTSSCCDALWZTCPQPYJLLQZYJSWXMZZMMYLMXCLMX";

listJP << "CZMXMZSQTZPPQQBLPGXQZHFLJJHYTJSRXWZXSCCDLXTYJDCQJXSLQYCLZXLZZXMXQRJMHRHZJBHMFLJLMLCLQNLDXZLLLPYPSYJY";

listJP << "SXCQQDCMQJZZXHNPNXZMEKMXHYKYQLXSXTXJYYHWDCWDZHQYYBGYBCYSCFGPSJNZDYZZJZXRZRQJJYMCANYRJTLDPPYZBSTJKXXZ";

listJP << "YPFDWFGZZRPYMTNGXZQBYXNBUFNQKRJQZMJEGRZGYCLKXZDSKKNSXKCLJSPJYYZLQQJYBZSSQLLLKJXTBKTYLCCDDBLSPPFYLGYD";

listJP << "TZJYQGGKQTTFZXBDKTYYHYBBFYTYYBCLPDYTGDHRYRNJSPTCSNYJQHKLLLZSLYDXXWBCJQSPXBPJZJCJDZFFXXBRMLAZHCSNDLBJ";

listJP << "DSZBLPRZTSWSBXBCLLXXLZDJZSJPYLYXXYFTFFFBHJJXGBYXJPMMMPSSJZJMTLYZJXSWXTYLEDQPJMYGQZJGDJLQJWJQLLSJGJGY";

listJP << "GMSCLJJXDTYGJQJQJCJZCJGDZZSXQGSJGGCXHQXSNQLZZBXHSGZXCXYLJXYXYYDFQQJHJFXDHCTXJYRXYSQTJXYEFYYSSYYJXNCY";

listJP << "ZXFXMSYSZXYYSCHSHXZZZGZZZGFJDLTYLNPZGYJYZYYQZPBXQBDZTZCZYXXYHHSQXSHDHGQHJHGYWSZTMZMLHYXGEBTYLZKQWYTJ";

listJP << "ZRCLEKYSTDBCYKQQSAYXCJXWWGSBHJYZYDHCSJKQCXSWXFLTYNYZPZCCZJQTZWJQDZZZQZLJJXLSBHPYXXPSXSHHEZTXFPTLQYZZ";

listJP << "XHYTXNCFZYYHXGNXMYWXTZSJPTHHGYMXMXQZXTSBCZYJYXXTYYZYPCQLMMSZMJZZLLZXGXZAAJZYXJMZXWDXZSXZDZXLEYJJZQBH";

listJP << "ZWZZZQTZPSXZTDSXJJJZNYAZPHXYYSRNQDTHZHYYKYJHDZXZLSWCLYBZYECWCYCRYLCXNHZYDZYDYJDFRJJHTRSQTXYXJRJHOJYN";

listJP << "XELXSFSFJZGHPZSXZSZDZCQZBYYKLSGSJHCZSHDGQGXYZGXCHXZJWYQWGYHKSSEQZZNDZFKWYSSTCLZSTSYMCDHJXXYWEYXCZAYD";

listJP << "MPXMDSXYBSQMJMZJMTZQLPJYQZCGQHXJHHLXXHLHDLDJQCLDWBSXFZZYYSCHTYTYYBHECXHYKGJPXHHYZJFXHWHBDZFYZBCAPNPG";

listJP << "NYDMSXHMMMMAMYNBYJTMPXYYMCTHJBZYFCGTYHWPHFTWZZEZSBZEGPFMTSKFTYCMHFLLHGPZJXZJGZJYXZSBBQSCZZLZCCSTPGXM";

listJP << "JSFTCCZJZDJXCYBZLFCJSYZFGSZLYBCWZZBYZDZYPSWYJZXZBDSYUXLZZBZFYGCZXBZHZFTPBGZGEJBSTGKDMFHYZZJHZLLZZGJQ";

listJP << "ZLSFDJSSCBZGPDLFZFZSZYZYZSYGCXSNXXCHCZXTZZLJFZGQSQYXZJQDCCZTQCDXZJYQJQCHXZTDLGSCXZSYQJQTZWLQDQZTQCHQ";

listJP << "QJZYEZZZPBWKDJFCJPZTYPQYQTTYNLMBDKTJZPQZQZZFPZSBNJLGYJDXJDZZKZGQKXDLPZJTCJDQBXDJQJSTCKNXBXZMSLYJCQMT";

listJP << "JQWWCJQNJNLLLHJCWQTBZQYDZCZPZZDZYDDCYZZZCCJTTJFZDPRRTZTJDCQTQZDTJNPLZBCLLCTZSXKJZQZPZLBZRBTJDCXFCZDB";

listJP << "CCJJLTQQPLDCGZDBBZJCQDCJWYNLLZYZCCDWLLXWZLXRXNTQQCZXKQLSGDFQTDDGLRLAJJTKUYMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

YMKQLLTZYTDYYCZGJWYXDXFRSKS";

listJP << "TQTENQMRKQZHHQKDLDAZFKYPBGGPZREBZZYKZZSPEGJXGYKQZZZSLYSYYYZWFQZYLZZLZHWCHKYPQGNPGBLPLRRJYXCCSYYHSFZF";

listJP << "YBZYYTGZXYLXCZWXXZJZBLFFLGSKHYJZEYJHLPLLLLCZGXDRZELRHGKLZZYHZLYQSZZJZQLJZFLNBHGWLCZCFJYSPYXZLZLXGCCP";

listJP << "ZBLLCYBBBBUBBCBPCRNNZCZYRBFSRLDCGQYYQXYGMQZWTZYTYJXYFWTEHZZJYWLCCNTZYJJZDEDPZDZTSYQJHDYMBJNYJZLXTSST";

listJP << "PHNDJXXBYXQTZQDDTJTDYYTGWSCSZQFLSHLGLBCZPHDLYZJYCKWTYTYLBNYTSDSYCCTYSZYYEBHEXHQDTWNYGYCLXTSZYSTQMYGZ";

listJP << "AZCCSZZDSLZCLZRQXYYELJSBYMXSXZTEMBBLLYYLLYTDQYSHYMRQWKFKBFXNXSBYCHXBWJYHTQBPBSBWDZYLKGZSKYHXQZJXHXJX";

listJP << "GNLJKZLYYCDXLFYFGHLJGJYBXQLYBXQPQGZTZPLNCYPXDJYQYDYMRBESJYYHKXXSTMXRCZZYWXYQYBMCLLYZHQYZWQXDBXBZWZMS";

listJP << "LPDMYSKFMZKLZCYQYCZLQXFZZYDQZPZYGYJYZMZXDZFYFYTTQTZHGSPCZMLCCYTZXJCYTJMKSLPZHYSNZLLYTPZCTZZCKTXDHXXT";

listJP << "QCYFKSMQCCYYAZHTJPCYLZLYJBJXTPNYLJYYNRXSYLMMNXJSMYBCSYSYLZYLXJJQYLDZLPQBFZZBLFNDXQKCZFYWHGQMRDSXYCYT";

listJP << "XNQQJZYYPFZXDYZFPRXEJDGYQBXRCNFYYQPGHYJDYZXGRHTKYLNWDZNTSMPKLBTHBPYSZBZTJZSZZJTYYXZPHSSZZBZCZPTQFZMY";

listJP << "FLYPYBBJQXZMXXDJMTSYSKKBJZXHJCKLPSMKYJZCXTMLJYXRZZQSLXXQPYZXMKYXXXJCLJPRMYYGADYSKQLSNDHYZKQXZYZTCGHZ";

listJP << "TLMLWZYBWSYCTBHJHJFCWZTXWYTKZLXQSHLYJZJXTMPLPYCGLTBZZTLZJCYJGDTCLKLPLLQPJMZPAPXYZLKKTKDZCZZBNZDYDYQZ";

listJP << "JYJGMCTXLTGXSZLMLHBGLKFWNWZHDXUHLFMKYSLGXDTWWFRJEJZTZHYDXYKSHWFZCQSHKTMQQHTZHYMJDJSKHXZJZBZZXYMPAGQM";

listJP << "STPXLSKLZYNWRTSQLSZBPSPSGZWYHTLKSSSWHZZLYYTNXJGMJSZSUFWNLSOZTXGXLSAMMLBWLDSZYLAKQCQCTMYCFJBSLXCLZZCL";

listJP << "XXKSBZQCLHJPSQPLSXXCKSLNHPSFQQYTXYJZLQLDXZQJZDYYDJNZPTUZDSKJFSLJHYLZSQZLBTXYDGTQFDBYAZXDZHZJNHHQBYKN";

listJP << "XJJQCZMLLJZKSPLDYCLBBLXKLELXJLBQYCXJXGCNLCQPLZLZYJTZLJGYZDZPLTQCSXFDMNYCXGBTJDCZNBGBQYQJWGKFHTNPYQZQ";

listJP << "GBKPBBYZMTJDYTBLSQMPSXTBNPDXKLEMYYCJYNZCTLDYKZZXDDXHQSHDGMZSJYCCTAYRZLPYLTLKXSLZCGGEXCLFXLKJRTLQJAQZ";

listJP << "NCMBYDKKCXGLCZJZXJHPTDJJMZQYKQSECQZDSHHADMLZFMMZBGNTJNNLGBYJBRBTMLBYJDZXLCJLPLDLPCQDHLXZLYCBLCXZZJAD";

listJP << "JLNZMMSSSMYBHBSQKBHRSXXJMXSDZNZPXLGBRHWGGFCXGMSKLLTSJYYCQLTSKYWYYHYWXBXQYWPYWYKQLSQPTNTKHQCWDQKTWPXX";

listJP << "HCPTHTWUMSSYHBWCRWXHJMKMZNGWTMLKFGHKJYLSYYCXWHYECLQHKQHTTQKHFZLDXQWYZYYDESBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

SBPKYRZPJFYYZJCEQDZZDLATZBB";

listJP << "FJLLCXDLMJSSXEGYGSJQXCWBXSSZPDYZCXDNYXPPZYDLYJCZPLTXLSXYZYRXCYYYDYLWWNZSAHJSYQYHGYWWAXTJZDAXYSRLTDPS";

listJP << "SYYFNEJDXYZHLXLLLZQZSJNYQYQQXYJGHZGZCYJCHZLYCDSHWSHJZYJXCLLNXZJJYYXNFXMWFPYLCYLLABWDDHWDXJMCXZTZPMLQ";

listJP << "ZHSFHZYNZTLLDYWLSLXHYMMYLMBWWKYXYADTXYLLDJPYBPWUXJMWMLLSAFDLLYFLBHHHBQQLTZJCQJLDJTFFKMMMBYTHYGDCQRDD";

listJP << "WRQJXNBYSNWZDBYYTBJHPYBYTTJXAAHGQDQTMYSTQXKBTZPKJLZRBEQQSSMJJBDJOTGTBXPGBKTLHQXJJJCTHXQDWJLWRFWQGWSH";

listJP << "CKRYSWGFTGYGBXSDWDWRFHWYTJJXXXJYZYSLPYYYPAYXHYDQKXSHXYXGSKQHYWFDDDPPLCJLQQEEWXKSYYKDYPLTJTHKJLTCYYHH";

listJP << "JTTPLTZZCDLTHQKZXQYSTEEYWYYZYXXYYSTTJKLLPZMCYHQGXYHSRMBXPLLNQYDQHXSXXWGDQBSHYLLPJJJTHYJKYPPTHYYKTYEZ";

listJP << "YENMDSHLCRPQFDGFXZPSFTLJXXJBSWYYSKSFLXLPPLBBBLBSFXFYZBSJSSYLPBBFFFFSSCJDSTZSXZRYYSYFFSYZYZBJTBCTSBSD";

listJP << "HRTJJBYTCXYJEYLXCBNEBJDSYXYKGSJZBXBYTFZWGENYHHTHZHHXFWGCSTBGXKLSXYWMTMBYXJSTZSCDYQRCYTWXZFHMYMCXLZNS";

listJP << "DJTTTXRYCFYJSBSDYERXJLJXBBDEYNJGHXGCKGSCYMBLXJMSZNSKGXFBNBPTHFJAAFXYXFPXMYPQDTZCXZZPXRSYWZDLYBBKTYQP";

listJP << "QJPZYPZJZNJPZJLZZFYSBTTSLMPTZRTDXQSJEHBZYLZDHLJSQMLHTXTJECXSLZZSPKTLZKQQYFSYGYWPCPQFHQHYTQXZKRSGTTSQ";

listJP << "CZLPTXCDYYZXSQZSLXLZMYCPCQBZYXHBSXLZDLTCDXTYLZJYYZPZYZLTXJSJXHLPMYTXCQRBLZSSFJZZTNJYTXMYJHLHPPLCYXQJ";

listJP << "QQKZZSCPZKSWALQSBLCCZJSXGWWWYGYKTJBBZTDKHXHKGTGPBKQYSLPXPJCKBMLLXDZSTBKLGGQKQLSBKKTFXRMDKBFTPZFRTBBR";

listJP << "FERQGXYJPZSSTLBZTPSZQZSJDHLJQLZBPMSMMSXLQQNHKNBLRDDNXXDHDDJCYYGYLXGZLXSYGMQQGKHBPMXYXLYTQWLWGCPBMQXC";

listJP << "YZYDRJBHTDJYHQSHTMJSBYPLWHLZFFNYPMHXXHPLTBQPFBJWQDBYGPNZTPFZJGSDDTQSHZEAWZZYLLTYYBWJKXXGHLFKXDJTMSZS";

listJP << "QYNZGGSWQSPHTLSSKMCLZXYSZQZXNCJDQGZDLFNYKLJCJLLZLMZZNHYDSSHTHZZLZZBBHQZWWYCRZHLYQQJBEYFXXXWHSRXWQHWP";

listJP << "SLMSSKZTTYGYQQWRSLALHMJTQJSMXQBJJZJXZYZKXBYQXBJXSHZTSFJLXMXZXFGHKZSZGGYLCLSARJYHSLLLMZXELGLXYDJYTLFB";

listJP << "HBPNLYZFBBHPTGJKWETZHKJJXZXXGLLJLSTGSHJJYQLQZFKCGNNDJSSZFDBCTWWSEQFHQJBSAQTGYPQLBXBMMYWXGSLZHGLZGQYF";

listJP << "LZBYFZJFRYSFMBYZHQGFWZSYFYJJPHZBYYZFFWODGRLMFTWLBZGYCQXCDJYGZYYYYTYTYDWEGAZYHXJLZYYHLRMGRXXZCLHNELJJ";

listJP << "TJTPWJYBJJBXJJTJTEEKHWSLJPLPSFYZPQQBDLQJJTYYQLYZKDKSQJYYQZLDQTGJQYZJSUCMRYQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

YQTHTEJMFCTYHYPKMHYZWJDQFHY";

listJP << "YXWSHCTXRLJHQXHCCYYYJLTKTTYTMXGTCJTZAYYOCZLYLBSZYWJYTSJYHBYSHFJLYGJXXTMZYYLTXXYPZLXYJZYZYYPNHMYMDYYL";

listJP << "BLHLSYYQQLLNJJYMSOYQBZGDLYXYLCQYXTSZEGXHZGLHWBLJHEYXTWQMAKBPQCGYSHHEGQCMWYYWLJYJHYYZLLJJYLHZYHMGSLJL";

listJP << "JXCJJYCLYCJPCPZJZJMMYLCQLNQLJQJSXYJMLSZLJQLYCMMHCFMMFPQQMFYLQMCFFQMMMMHMZNFHHJGTTHHKHSLNCHHYQDXTMMQD";

listJP << "CYZYXYQMYQYLTDCYYYZAZZCYMZYDLZFFFMMYCQZWZZMABTBYZTDMNZZGGDFTYPCGQYTTSSFFWFDTZQSSYSTWXJHXYTSXXYLBYQHW";

listJP << "WKXHZXWZNNZZJZJJQJCCCHYYXBZXZCYZTLLCQXYNJYCYYCYNZZQYYYEWYCZDCJYCCHYJLBTZYYCQWMPWPYMLGKDLDLGKQQBGYCHJ";

listJP << "XY";

}

//汉语转拼音

QString CNLHelper::CNToPY(const QString &cnstr)

{

QStringList list;

for (int i = 0; i < cnstr.length(); ++i) {

int unicode = QString::number(cnstr.at(i).unicode(), 10).toInt();

if (unicode >= 0x4E00 && unicode <= 0x9FA5) {

//这里的listPY就是按照UNICODE每个中文对应的拼音数组

list.append(listPY.at(unicode - 0x4E00));

} else {

list.append(cnstr.at(i));

}

}

return list.join("");

}

//汉语转所有字首字母

QString CNLHelper::CNToEL(const QString &cnstr)

{

QString strChineseFirstPY = listJP.join("");

if(cnstr.length() == 0) {

return cnstr;

}

QString str;

int index = 0;

for(int i = 0; i < cnstr.length(); i++) {

//若是字母或数字则直接输出

ushort vChar = cnstr.at(i).unicode() ;

if((vChar >= 'a' && vChar <= 'z' ) || (vChar >= 'A' && vChar <= 'Z')) {

str.append(cnstr.at(i).toLower());

}

if((vChar >= '0' && vChar <= '9')) {

str.append(cnstr.at(i));

} else {

index = (int)vChar - 19968;

if(index >= 0 && index < strChineseFirstPY.length()) {

str.append(strChineseFirstPY.at(index));

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

}

}

}

return str;

}

//汉语转第一个汉字首字母

QString CNLHelper::CNToFL(const QString &cnstr)

{

return CNToEL(cnstr.mid(0,1));//取第一个汉字 并计算首字母

}

源文件 delete.cpp

#include "delete.h"

#include "ui\_delete.h"

#include <qdebug.h>

using namespace std;

Delete::Delete(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Delete)

{

ui->setupUi(this);

ui->treeWidget->setHeaderLabels(QStringList()<<"用户名"<<"密码"<<"邮箱");

ifstream in;

in.open("User.bat",ios::in);

string name,password,email;

QString now;

while(getline(in,name))

{

getline(in,password);

getline(in,email);

QTreeWidgetItem \* item = new QTreeWidgetItem(QStringList()<<now.fromStdString(name)<<now.fromStdString(password)<<now.fromStdString(email));

ui->treeWidget->addTopLevelItem(item);

}

in.close();

connect(ui->treeWidget, &QTreeWidget::itemDoubleClicked, this, &Delete::on\_treeWidget\_itemDoubleClick);

}

Delete::~Delete()

{

delete ui;

}

void Delete::on\_treeWidget\_itemDoubleClick(QTreeWidgetItem \*item, int column)//获取子项信息的槽函数实现

{

QString password;

QString email;

QString name;

if(column==0)

{

name=item->text(column);

password=item->text(column+1);

email=item->text(column+2);

}

if(column==1)

{

name=item->text(column-1);

password=item->text(column);

email=item->text(column+1);

}

if(column==-1)

{

name=item->text(column+1);

password=item->text(column+2);

email=item->text(column+3);

}

if(column==2)

{

name=item->text(column-2);

password=item->text(column-1);

email=item->text(column);

}

if(QMessageBox::Yes==QMessageBox::question(this,"提示","是否删除",QMessageBox::Yes|QMessageBox::No))

{

del(name.toStdString(),password.toStdString(),email.toStdString());

this->removeItem(item);

}

}

void Delete::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

}

void del(string name,string password,string email)//删除实现

{

string name\_f;

string password\_f;

string email\_f;

ifstream in("User.bat");

ofstream out;

out.open("Usernew.bat",ios::trunc);//初始化替代文件

while(getline(in,name\_f))

{

getline(in,password\_f);

getline(in,email\_f);

if(!(name\_f==name&&password\_f==password&&email\_f==email))

{

out<<name\_f<<"\n"<<password\_f<<"\n"<<email\_f<<"\n";

}

}

in.close();

out.close();

remove("User.bat");

rename("Usernew.bat", "User.bat");

string file=name+"connecters.bat";

const char \*file\_c=file.c\_str();

remove(file\_c);

file=name+"bestconnexters.bat";

file\_c=file.c\_str();

remove(file\_c);

}

void Delete::removeItem(QTreeWidgetItem \*item)//删除节点，可视化实现

{

int count = item->childCount();

if(count==0)//没有子节点，直接删除

{

delete item;

return;

}

for(int i=0; i<count; i++)

{

QTreeWidgetItem \*childItem = item->child(0);//删除子节点

removeItem(childItem);

}

delete item;//最后将自己删除

return;

}

password=item->text(column);

email=item->text(column+1);

}

if(column==-1)

{

name=item->text(column+1);

password=item->text(column+2);

email=item->text(column+3);

}

if(column==2)

{

name=item->text(column-2);

password=item->text(column-1);

email=item->text(column);

}

if(QMessageBox::Yes==QMessageBox::question(this,"提示","是否删除",QMessageBox::Yes|QMessageBox::No))

{

del(name.toStdString(),password.toStdString(),email.toStdString());

this->removeItem(item);

}

}

void Delete::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

}

void del(string name,string password,string email)//删除实现

{

string name\_f;

string password\_f;

string email\_f;

ifstream in("User.bat");

ofstream out;

out.open("Usernew.bat",ios::trunc);//初始化替代文件

while(getline(in,name\_f))

{

getline(in,password\_f);

getline(in,email\_f);

if(!(name\_f==name&&password\_f==password&&email\_f==email))

{

out<<name\_f<<"\n"<<password\_f<<"\n"<<email\_f<<"\n";

}

}

in.close();

out.close();

remove("User.bat");

rename("Usernew.bat", "User.bat");

string file=name+"connecters.bat";

const char \*file\_c=file.c\_str();

remove(file\_c);

file=name+"bestconnexters.bat";

file\_c=file.c\_str();

remove(file\_c);

}

void Delete::removeItem(QTreeWidgetItem \*item)//删除节点，可视化实现

{

int count = item->childCount();

if(count==0)//没有子节点，直接删除

{

delete item;

return;

}

for(int i=0; i<count; i++)

{

QTreeWidgetItem \*childItem = item->child(0);//删除子节点

removeItem(childItem);

}

delete item;//最后将自己删除

return;

}

if(count==0)//没有子节点，直接删除

{

delete item;

return;

}

for(int i=0; i<count; i++)

{

QTreeWidgetItem \*childItem = item->child(0);//删除子节点

removeItem(childItem);

}

delete item;//最后将自己删除

return;

}

源文件 detailpage.cpp

#include "detailpage.h"

#include "ui\_detailpage.h"

#include "Login.h"

#include "mainwindow.h"

#include "cnlhelper.h"

#include<newperson.h>

Detailpage::Detailpage(string name,string number,string Username,QWidget \*parent)

: QMainWindow(parent),ui(new Ui::Detailpage),conname(name),connumber(number)

, username(Username)

{

ui->setupUi(this);

QString name\_ccc=name\_ccc.fromStdString(conname);

this->setWindowTitle(name\_ccc);

string file=username+"connecters.bat";

ifstream in;

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

ui->textEdit->setPlainText(Currtext.fromStdString(currtext));

ui->lineEdit\_address->setText(Currtext.fromStdString(curraddress));

ui->lineEdit\_birth->setText(Currtext.fromStdString(currbirth));

ui->lineEdit\_company->setText(Currtext.fromStdString(currcompany));

ui->lineEdit\_email->setText(Currtext.fromStdString(curremail));

ui->lineEdit\_name->setText(Currtext.fromStdString(currname));

ui->lineEdit\_number->setText(Currtext.fromStdString(currnumber));

ui->lineEdit\_social->setText(Currtext.fromStdString(currsocial));

}

}

}

Detailpage::~Detailpage()

{

delete ui;

}

void Detailpage::on\_pushButton\_clicked()//返回功能实现

{

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

this->close();

}

void Detailpage::on\_pushButton\_4\_clicked()//删除联系人功能实现

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","删除成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_6\_clicked()//保存修改按钮实现

{

if(QMessageBox::question(this,"提示","是否要修改",QMessageBox::Yes|QMessageBox::No))

{

QString uinumber;

uinumber=ui->lineEdit\_number->text();

if(Newperson::checknumber(uinumber.toStdString()))

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

currtext="";

currname="";

currnumber="";

curremail="";

currcompany="";

curraddress="";

currbirth="";

currsocial="";

int flag=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

else

{

flag=1;

}

}

if(flag==1)

{

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式错误");

}

}

}

void Detailpage::on\_pushButton\_2\_clicked()//分享联系人按钮实现

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

ui->lineEdit\_address->setText(Currtext.fromStdString(curraddress));

ui->lineEdit\_birth->setText(Currtext.fromStdString(currbirth));

ui->lineEdit\_company->setText(Currtext.fromStdString(currcompany));

ui->lineEdit\_email->setText(Currtext.fromStdString(curremail));

ui->lineEdit\_name->setText(Currtext.fromStdString(currname));

ui->lineEdit\_number->setText(Currtext.fromStdString(currnumber));

ui->lineEdit\_social->setText(Currtext.fromStdString(currsocial));

}

}

}

Detailpage::~Detailpage()

{

delete ui;

}

void Detailpage::on\_pushButton\_clicked()//返回功能实现

{

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

this->close();

}

void Detailpage::on\_pushButton\_4\_clicked()//删除联系人功能实现

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","删除成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_6\_clicked()//保存修改按钮实现

{

if(QMessageBox::question(this,"提示","是否要修改",QMessageBox::Yes|QMessageBox::No))

{

QString uinumber;

uinumber=ui->lineEdit\_number->text();

if(Newperson::checknumber(uinumber.toStdString()))

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

currtext="";

currname="";

currnumber="";

curremail="";

currcompany="";

curraddress="";

currbirth="";

currsocial="";

int flag=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

else

{

flag=1;

}

}

if(flag==1)

{

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式错误");

}

}

}

void Detailpage::on\_pushButton\_2\_clicked()//分享联系人按钮实现

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

}

}

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","删除成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_6\_clicked()//保存修改按钮实现

{

if(QMessageBox::question(this,"提示","是否要修改",QMessageBox::Yes|QMessageBox::No))

{

QString uinumber;

uinumber=ui->lineEdit\_number->text();

if(Newperson::checknumber(uinumber.toStdString()))

{

string file=username+"connecters.bat";

string file\_new=username+"connectersnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

currtext="";

currname="";

currnumber="";

curremail="";

currcompany="";

curraddress="";

currbirth="";

currsocial="";

int flag=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

else

{

flag=1;

}

}

if(flag==1)

{

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式错误");

}

}

}

void Detailpage::on\_pushButton\_2\_clicked()//分享联系人按钮实现

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

}

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

file=username+"bestconnexters.bat";

file\_new=username+"bestconnextersnew.bat";

in.open(file);

out.open(file\_new,ios::trunc);

currtext="";

currname="";

currnumber="";

curremail="";

currcompany="";

curraddress="";

currbirth="";

currsocial="";

int flag=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

else

{

flag=1;

}

}

if(flag==1)

{

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式错误");

}

}

}

void Detailpage::on\_pushButton\_2\_clicked()//分享联系人按钮实现

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(!(currname==conname&&currnumber==connumber))

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

}

else

{

flag=1;

}

}

if(flag==1)

{

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

}

in.close();

out.close();

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号格式错误");

}

}

}

void Detailpage::on\_pushButton\_2\_clicked()//分享联系人按钮实现

{

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

string file=username+"connecters.bat";

CNLHelper \*cnl=new CNLHelper;

QString name;

QString file\_q=cnl->CNToPY(name.fromStdString(conname));

string file\_con=file\_q.toStdString();

ifstream in;

in.open(file);

ofstream out;

out.open(file\_con+".txt",ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<"备注:\n"<<currtext<<"\n联系人名称:\n"<<currname<<"\n电话号:\n"<<currnumber<<"\n邮箱:\n"

<<curremail<<"\n联系人公司:\n"<<currcompany<<"\n联系人地址:\n"<<curraddress<<"\n联系人生日:\n"

<<currbirth<<"\n联系人社交资料\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","分享成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Detailpage::on\_pushButton\_3\_clicked()//收藏联系人功能实现

{

string file=username+"connecters.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(username+"bestconnexters.bat",ios::app);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

string currbirth="";

string currsocial="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

if(currname==conname&&currnumber==connumber)

{

out<<currtext<<"\n"<<currname<<"\n"<<currnumber<<"\n"<<

curremail<<"\n"<<currcompany<<"\n"<<curraddress<<"\n"<<currbirth<<"\n"<<currsocial<<"\n";

break;

}

}

out.close();

in.close();

QMessageBox::information(this,"提示","收藏成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

源文件Login.cpp

#include "Login.h"

#include "ui\_Login.h"

#include "mainwindow.h"

#include "signup.h"

#include "retrieverpassword.h"

#include "ui\_retrieverpassword.h"

#include "delete.h"

using namespace std;

typedef struct Users//定义用户的结构体，存储用户的数据

{

string name;

string password;

string email;

Users \*next;

}\*user,User;

User CurrentUser;//当前的使用者的账户和密码

User Adiminatoner;//管理员账号

user CorrectUser=(user)malloc(sizeof(User));//正确的使用者的账户和密码

bool flag=1;//有关是否显示密码的bool类型值

user head=(user)malloc(sizeof(User)\*100);

char characterx[4]={'+','-','\*'};

Login::Login(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::Login)

{

ui->setupUi(this);

ui->lineEdit\_2->setPlaceholderText("请输入用户名");//这两行为用户名和密码的输入的提示

ui->lineEdit->setPlaceholderText("请输入密码");

ui->lineEdit->setEchoMode(QLineEdit::Password);//密码显示为小圆点，不可见

ui->toolButton->setIcon(QIcon(":/eyesopen.png"));

}

Login::~Login()

{

delete ui;

}

int check()//比较输入的信息和用户信息是否相同tyx

{

ifstream in;

in.open("User.bat");

string name\_f,password\_f,email\_f;

while(getline(in,name\_f))

{

getline(in,password\_f);

getline(in,email\_f);

if(name\_f==CurrentUser.name&&password\_f==CurrentUser.password)

{

in.close();

return 1;

}

}

in.close();

return 0;

}

void Login::on\_pushButton\_2\_clicked()//退出键的实现，按下按钮退出

{

Login::close();

}

void Login::on\_pushButton\_clicked()//登录按钮的实现，当按下登录按钮的时候，判断是否与注册的账号密码相同tyx

{

FILE \*file;

file=fopen("User.bat","rb");

if(file==NULL)

{

QMessageBox::warning(this,"警告","文件错误，请联系管理员");

}

fclose(file);

QString name = ui->lineEdit\_2->text();//获取用户名的文本

QString password = ui->lineEdit->text();//获取密码的文本

string Name=name.toStdString();//强制转换qstring为string

string Password=password.toStdString();

CurrentUser.name=Name;

CurrentUser.password=Password;

Adiminatoner.name="pikachu";

Adiminatoner.password="2005415qwer";

string currentuser=CurrentUser.name;

if(CurrentUser.name==Adiminatoner.name&&CurrentUser.password==Adiminatoner.password)

{

QMessageBox::information(this,"提示","你好，管理员");

Delete \*del=new Delete();

del->show();

}

else if(check())//当前的用户名和密码与设置的用户名密码相同

{

//生成登录成功的对话框

QMessageBox::information(this,"提示","登录成功");

//跳转至主界面并关闭当前界面

MainWindow \*main=new MainWindow(currentuser);

main->show();

this->close();

}

else

{

QMessageBox::warning(this,"警告","输入错误");

}

}

void Login::on\_toolButton\_clicked()//实现密码显示时，眼睛图标的转换

{

if(!flag)//如果判定flag为假，显示眼睛开始的状态

{

ui->toolButton->setIcon(QIcon(":/eyesopen.png"));

ui->lineEdit->setEchoMode(QLineEdit::Password);

flag=!flag;//flag标签反转

}

else//反之，显示眼睛关闭的状态

{

ui->toolButton->setIcon(QIcon(":/eyesclose.png"));

ui->lineEdit->setEchoMode(QLineEdit::Normal);//转换密码形式为可见，用户查看密码

flag=!flag;

}

}

void Login::on\_pushButton\_3\_clicked()//创建用户

{

Signup \*signup=new Signup;

signup->setWindowModality(Qt::WindowModal);

signup->show();

}

void Login::on\_pushButton\_4\_clicked()//忘记密码按钮实现

{

srand((unsigned)time(NULL));

int randnum1 = rand() % 10+1;

int randnum2 = rand() % 10+1;

int randchar = rand() % (3-1);

char x;

x=characterx[randchar];

string str=std::to\_string(randnum1)+x+std::to\_string(randnum2);

Retrieverpassword \*retrie2=new Retrieverpassword();

retrie2->ui->lineEdit\_5->setText(QString::fromStdString(str));

int randnum3 = rand() % 10+1;

int randnum4 = rand() % 10+1;

int randchar2 = rand() % (3-1);

x=characterx[randchar2];

str=std::to\_string(randnum3)+x+std::to\_string(randnum4);

retrie2->ui->lineEdit\_8->setText(QString::fromStdString(str));

retrie2->show();

}

}

in.close();

return 0;

}

void Login::on\_pushButton\_2\_clicked()//退出键的实现，按下按钮退出

{

Login::close();

}

void Login::on\_pushButton\_clicked()//登录按钮的实现，当按下登录按钮的时候，判断是否与注册的账号密码相同tyx

{

ifstream in;

in.open("User.bat");

in.close();

QString name = ui->lineEdit\_2->text();//获取用户名的文本

QString password = ui->lineEdit->text();//获取密码的文本

string Name=name.toStdString();//强制转换qstring为string

string Password=password.toStdString();

CurrentUser.name=Name;

CurrentUser.password=Password;

Adiminatoner.name="pikachu";

Adiminatoner.password="2005415qwer";

string currentuser=CurrentUser.name;

if(CurrentUser.name==Adiminatoner.name&&CurrentUser.password==Adiminatoner.password)

{

QMessageBox::information(this,"提示","你好，管理员");

Delete \*del=new Delete();

del->show();

}

else if(check())//当前的用户名和密码与设置的用户名密码相同

{

//生成登录成功的对话框

QMessageBox::information(this,"提示","登录成功");

//跳转至主界面并关闭当前界面

MainWindow \*main=new MainWindow(currentuser);

main->show();

this->close();

}

else

{

QMessageBox::warning(this,"警告","输入错误");

}

}

void Login::on\_toolButton\_clicked()//实现密码显示时，眼睛图标的转换

{

if(!flag)//如果判定flag为假，显示眼睛开始的状态

{

ui->toolButton->setIcon(QIcon(":/eyesopen.png"));

ui->lineEdit->setEchoMode(QLineEdit::Password);

flag=!flag;//flag标签反转

}

else//反之，显示眼睛关闭的状态

{

ui->toolButton->setIcon(QIcon(":/eyesclose.png"));

ui->lineEdit->setEchoMode(QLineEdit::Normal);//转换密码形式为可见，用户查看密码

flag=!flag;

}

}

void Login::on\_pushButton\_3\_clicked()//创建用户

{

Signup \*signup=new Signup;

signup->setWindowModality(Qt::WindowModal);

signup->show();

}

void Login::on\_pushButton\_4\_clicked()//忘记密码按钮实现

{

srand((unsigned)time(NULL));

int randnum1 = rand() % 10+1;

int randnum2 = rand() % 10+1;

int randchar = rand() % (3-1);

char x;

x=characterx[randchar];

string str=std::to\_string(randnum1)+x+std::to\_string(randnum2);

Retrieverpassword \*retrie2=new Retrieverpassword();

retrie2->ui->lineEdit\_5->setText(QString::fromStdString(str));

int randnum3 = rand() % 10+1;

int randnum4 = rand() % 10+1;

int randchar2 = rand() % (3-1);

x=characterx[randchar2];

str=std::to\_string(randnum3)+x+std::to\_string(randnum4);

retrie2->ui->lineEdit\_8->setText(QString::fromStdString(str));

retrie2->show();

}

}

else//反之，显示眼睛关闭的状态

{

ui->toolButton->setIcon(QIcon(":/eyesclose.png"));

ui->lineEdit->setEchoMode(QLineEdit::Normal);//转换密码形式为可见，用户查看密码

flag=!flag;

}

}

void Login::on\_pushButton\_3\_clicked()//创建用户

{

Signup \*signup=new Signup;

signup->setWindowModality(Qt::WindowModal);

signup->show();

}

void Login::on\_pushButton\_4\_clicked()//忘记密码按钮实现

{

srand((unsigned)time(NULL));

int randnum1 = rand() % 10+1;

int randnum2 = rand() % 10+1;

int randchar = rand() % (3-1);

char x;

x=characterx[randchar];

string str=std::to\_string(randnum1)+x+std::to\_string(randnum2);

Retrieverpassword \*retrie2=new Retrieverpassword();

retrie2->ui->lineEdit\_5->setText(QString::fromStdString(str));

int randnum3 = rand() % 10+1;

int randnum4 = rand() % 10+1;

int randchar2 = rand() % (3-1);

x=characterx[randchar2];

str=std::to\_string(randnum3)+x+std::to\_string(randnum4);

retrie2->ui->lineEdit\_8->setText(QString::fromStdString(str));

retrie2->show();

}

源文件 main.cpp

#include "Login.h"

#include <QApplication>

#include <QLocale>

#include <QTranslator>

int main(int argc, char \*argv[])

{

QApplication a(argc, argv);

QTranslator translator;

const QStringList uiLanguages = QLocale::system().uiLanguages();

for (const QString &locale : uiLanguages) {

const QString baseName = "LoginInterface\_" + QLocale(locale).name();

if (translator.load(":/i18n/" + baseName)) {

a.installTranslator(&translator);

break;

}

}

Login w;

w.show();

return a.exec();

}

源文件 mainwindow.cpp

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include "Login.h"

#include "newperson.h"

#include "ui\_newperson.h"

#include "detailpage.h"

#include "ui\_detailpage.h"

#include "cnlhelper.h"

#include "owninformation.h"

#include "ui\_owninformation.h"

#include "queryinterface.h"

#include "ui\_queryinterface.h"

connecters connects[10000];

int i=0;

QLocale local(QLocale::Chinese);

QCollator collator(local);

MainWindow::MainWindow(string name,QWidget \*parent)

: QMainWindow(parent),username(name)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

ini();

connect(ui->treeWidget, &QTreeWidget::itemDoubleClicked, this, &MainWindow::on\_treeWidget\_itemDoubleClicked);

}

void MainWindow::on\_treeWidget\_itemDoubleClicked(QTreeWidgetItem \*item, int column)//获取子项信息的槽函数实现

{

QString number="";

QString name="";

// 检查所选项是否为子项

if (item->parent() != 0) {

// 获取子项信息

if(column==1)

{

number=item->text(column);

name=item->text(column-1);

}

if(column==0)

{

number=item->text(column+1);

name=item->text(column);

}

Detailpage \*detail=new Detailpage(name.toStdString(),number.toStdString(),username,this);

detail->setWindowModality(Qt::WindowModal);

this->close();

detail->show();

}

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::on\_pushButton\_5\_clicked()//新建联系人实现

{

Newperson \*newperson= new Newperson(username);

newperson->setWindowModality(Qt::WindowModal);

newperson->show();

this->close();

}

void MainWindow::ini()//初始化

{

ui->treeWidget->clear();

QString Username\_c=Username\_c.fromStdString(username);

this->setWindowTitle(Username\_c);

//设置水平头

QString Username=QString::fromStdString(username);

this->setToolTip(Username);

ui->treeWidget->setHeaderLabels(QStringList()<<"联系人"<<"电话号码");

//加载节点

QTreeWidgetItem \* bestitem = new QTreeWidgetItem(QStringList()<<"特别关心");

ui->treeWidget->addTopLevelItem(bestitem);

QTreeWidgetItem \* signitem = new QTreeWidgetItem(QStringList()<<"\*");

ui->treeWidget->addTopLevelItem(signitem);

QTreeWidgetItem \* Aitem = new QTreeWidgetItem(QStringList()<<"A");

ui->treeWidget->addTopLevelItem(Aitem);

QTreeWidgetItem \* Bitem = new QTreeWidgetItem(QStringList()<<"B");

ui->treeWidget->addTopLevelItem(Bitem);

QTreeWidgetItem \* Citem = new QTreeWidgetItem(QStringList()<<"C");

ui->treeWidget->addTopLevelItem(Citem);

QTreeWidgetItem \* Ditem = new QTreeWidgetItem(QStringList()<<"D");

ui->treeWidget->addTopLevelItem(Ditem);

QTreeWidgetItem \* Eitem = new QTreeWidgetItem(QStringList()<<"E");

ui->treeWidget->addTopLevelItem(Eitem);

QTreeWidgetItem \* Fitem = new QTreeWidgetItem(QStringList()<<"F");

ui->treeWidget->addTopLevelItem(Fitem);

QTreeWidgetItem \* Gitem = new QTreeWidgetItem(QStringList()<<"G");

ui->treeWidget->addTopLevelItem(Gitem);

QTreeWidgetItem \* Hitem = new QTreeWidgetItem(QStringList()<<"H");

ui->treeWidget->addTopLevelItem(Hitem);

QTreeWidgetItem \* Iitem = new QTreeWidgetItem(QStringList()<<"I");

ui->treeWidget->addTopLevelItem(Iitem);

QTreeWidgetItem \* Jitem = new QTreeWidgetItem(QStringList()<<"J");

ui->treeWidget->addTopLevelItem(Jitem);

QTreeWidgetItem \* Kitem = new QTreeWidgetItem(QStringList()<<"K");

ui->treeWidget->addTopLevelItem(Kitem);

QTreeWidgetItem \* Litem = new QTreeWidgetItem(QStringList()<<"L");

ui->treeWidget->addTopLevelItem(Litem);

QTreeWidgetItem \* Mitem = new QTreeWidgetItem(QStringList()<<"M");

ui->treeWidget->addTopLevelItem(Mitem);

QTreeWidgetItem \* Nitem = new QTreeWidgetItem(QStringList()<<"N");

ui->treeWidget->addTopLevelItem(Nitem);

QTreeWidgetItem \* Oitem = new QTreeWidgetItem(QStringList()<<"O");

ui->treeWidget->addTopLevelItem(Oitem);

QTreeWidgetItem \* Pitem = new QTreeWidgetItem(QStringList()<<"P");

ui->treeWidget->addTopLevelItem(Pitem);

QTreeWidgetItem \* Qitem = new QTreeWidgetItem(QStringList()<<"Q");

ui->treeWidget->addTopLevelItem(Qitem);

QTreeWidgetItem \* Ritem = new QTreeWidgetItem(QStringList()<<"R");

ui->treeWidget->addTopLevelItem(Ritem);

QTreeWidgetItem \* Sitem = new QTreeWidgetItem(QStringList()<<"S");

ui->treeWidget->addTopLevelItem(Sitem);

QTreeWidgetItem \* Titem = new QTreeWidgetItem(QStringList()<<"T");

ui->treeWidget->addTopLevelItem(Titem);

QTreeWidgetItem \* Uitem = new QTreeWidgetItem(QStringList()<<"U");

ui->treeWidget->addTopLevelItem(Uitem);

QTreeWidgetItem \* Vitem = new QTreeWidgetItem(QStringList()<<"V");

ui->treeWidget->addTopLevelItem(Vitem);

QTreeWidgetItem \* Witem = new QTreeWidgetItem(QStringList()<<"W");

ui->treeWidget->addTopLevelItem(Witem);

QTreeWidgetItem \* Xitem = new QTreeWidgetItem(QStringList()<<"X");

ui->treeWidget->addTopLevelItem(Xitem);

QTreeWidgetItem \* Yitem = new QTreeWidgetItem(QStringList()<<"Y");

ui->treeWidget->addTopLevelItem(Yitem);

QTreeWidgetItem \* Zitem = new QTreeWidgetItem(QStringList()<<"Z");

ui->treeWidget->addTopLevelItem(Zitem);

//追加子节点

string file=username+"connecters.bat";

ifstream in;

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

i=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

connects[i].currtext=currtext;

connects[i].currname=currname;

connects[i].currnumber=currnumber;

connects[i].curremail=curremail;

connects[i].currcompany=currcompany;

connects[i].curraddress=curraddress;

connects[i].currbirth=currbirth;

connects[i].currsocial=currsocial;

if(!IncludeChinese(connects[i].currname.c\_str()))

{

connects[i].check=connects[i].currname;

}

else

{

CNLHelper \*cnl=new CNLHelper;

Currtext=Currtext.fromStdString(connects[i].currname);

Currtext=cnl->CNToPY(Currtext);

connects[i].check=Currtext.toStdString();

}

i++;

}

int sum=i;

QuickSort(sum);

in.close();

for(int j=0;j<i;j++)

{

if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='b'||connects[j].check[0]=='B')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Bitem->addChild(a);

}

else if(connects[j].check[0]=='c'||connects[j].check[0]=='C')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Citem->addChild(a);

}

else if(connects[j].check[0]=='d'||connects[j].check[0]=='D')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ditem->addChild(a);

}

else if(connects[j].check[0]=='e'||connects[j].check[0]=='E')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Eitem->addChild(a);

}

else if(connects[j].check[0]=='f'||connects[j].check[0]=='F')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Fitem->addChild(a);

}

else if(connects[j].check[0]=='g'||connects[j].check[0]=='G')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Gitem->addChild(a);

}

else if(connects[j].check[0]=='h'||connects[j].check[0]=='H')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Hitem->addChild(a);

}

else if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='i'||connects[j].check[0]=='I')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Iitem->addChild(a);

}

else if(connects[j].check[0]=='j'||connects[j].check[0]=='J')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

{

delete ui;

}

void MainWindow::on\_pushButton\_5\_clicked()//新建联系人实现

{

Newperson \*newperson= new Newperson(username);

newperson->setWindowModality(Qt::WindowModal);

newperson->show();

this->close();

}

void MainWindow::ini()//初始化

{

ui->treeWidget->clear();

QString Username\_c=Username\_c.fromStdString(username);

this->setWindowTitle(Username\_c);

//设置水平头

QString Username=QString::fromStdString(username);

this->setToolTip(Username);

ui->treeWidget->setHeaderLabels(QStringList()<<"联系人"<<"电话号码");

//加载节点

QTreeWidgetItem \* bestitem = new QTreeWidgetItem(QStringList()<<"特别关心");

ui->treeWidget->addTopLevelItem(bestitem);

QTreeWidgetItem \* signitem = new QTreeWidgetItem(QStringList()<<"\*");

ui->treeWidget->addTopLevelItem(signitem);

QTreeWidgetItem \* Aitem = new QTreeWidgetItem(QStringList()<<"A");

ui->treeWidget->addTopLevelItem(Aitem);

QTreeWidgetItem \* Bitem = new QTreeWidgetItem(QStringList()<<"B");

ui->treeWidget->addTopLevelItem(Bitem);

QTreeWidgetItem \* Citem = new QTreeWidgetItem(QStringList()<<"C");

ui->treeWidget->addTopLevelItem(Citem);

QTreeWidgetItem \* Ditem = new QTreeWidgetItem(QStringList()<<"D");

ui->treeWidget->addTopLevelItem(Ditem);

QTreeWidgetItem \* Eitem = new QTreeWidgetItem(QStringList()<<"E");

ui->treeWidget->addTopLevelItem(Eitem);

QTreeWidgetItem \* Fitem = new QTreeWidgetItem(QStringList()<<"F");

ui->treeWidget->addTopLevelItem(Fitem);

QTreeWidgetItem \* Gitem = new QTreeWidgetItem(QStringList()<<"G");

ui->treeWidget->addTopLevelItem(Gitem);

QTreeWidgetItem \* Hitem = new QTreeWidgetItem(QStringList()<<"H");

ui->treeWidget->addTopLevelItem(Hitem);

QTreeWidgetItem \* Iitem = new QTreeWidgetItem(QStringList()<<"I");

ui->treeWidget->addTopLevelItem(Iitem);

QTreeWidgetItem \* Jitem = new QTreeWidgetItem(QStringList()<<"J");

ui->treeWidget->addTopLevelItem(Jitem);

QTreeWidgetItem \* Kitem = new QTreeWidgetItem(QStringList()<<"K");

ui->treeWidget->addTopLevelItem(Kitem);

QTreeWidgetItem \* Litem = new QTreeWidgetItem(QStringList()<<"L");

ui->treeWidget->addTopLevelItem(Litem);

QTreeWidgetItem \* Mitem = new QTreeWidgetItem(QStringList()<<"M");

ui->treeWidget->addTopLevelItem(Mitem);

QTreeWidgetItem \* Nitem = new QTreeWidgetItem(QStringList()<<"N");

ui->treeWidget->addTopLevelItem(Nitem);

QTreeWidgetItem \* Oitem = new QTreeWidgetItem(QStringList()<<"O");

ui->treeWidget->addTopLevelItem(Oitem);

QTreeWidgetItem \* Pitem = new QTreeWidgetItem(QStringList()<<"P");

ui->treeWidget->addTopLevelItem(Pitem);

QTreeWidgetItem \* Qitem = new QTreeWidgetItem(QStringList()<<"Q");

ui->treeWidget->addTopLevelItem(Qitem);

QTreeWidgetItem \* Ritem = new QTreeWidgetItem(QStringList()<<"R");

ui->treeWidget->addTopLevelItem(Ritem);

QTreeWidgetItem \* Sitem = new QTreeWidgetItem(QStringList()<<"S");

ui->treeWidget->addTopLevelItem(Sitem);

QTreeWidgetItem \* Titem = new QTreeWidgetItem(QStringList()<<"T");

ui->treeWidget->addTopLevelItem(Titem);

QTreeWidgetItem \* Uitem = new QTreeWidgetItem(QStringList()<<"U");

ui->treeWidget->addTopLevelItem(Uitem);

QTreeWidgetItem \* Vitem = new QTreeWidgetItem(QStringList()<<"V");

ui->treeWidget->addTopLevelItem(Vitem);

QTreeWidgetItem \* Witem = new QTreeWidgetItem(QStringList()<<"W");

ui->treeWidget->addTopLevelItem(Witem);

QTreeWidgetItem \* Xitem = new QTreeWidgetItem(QStringList()<<"X");

ui->treeWidget->addTopLevelItem(Xitem);

QTreeWidgetItem \* Yitem = new QTreeWidgetItem(QStringList()<<"Y");

ui->treeWidget->addTopLevelItem(Yitem);

QTreeWidgetItem \* Zitem = new QTreeWidgetItem(QStringList()<<"Z");

ui->treeWidget->addTopLevelItem(Zitem);

//追加子节点

string file=username+"connecters.bat";

ifstream in;

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

i=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

connects[i].currtext=currtext;

connects[i].currname=currname;

connects[i].currnumber=currnumber;

connects[i].curremail=curremail;

connects[i].currcompany=currcompany;

connects[i].curraddress=curraddress;

connects[i].currbirth=currbirth;

connects[i].currsocial=currsocial;

if(!IncludeChinese(connects[i].currname.c\_str()))

{

connects[i].check=connects[i].currname;

}

else

{

CNLHelper \*cnl=new CNLHelper;

Currtext=Currtext.fromStdString(connects[i].currname);

Currtext=cnl->CNToPY(Currtext);

connects[i].check=Currtext.toStdString();

}

i++;

}

int sum=i;

QuickSort(sum);

in.close();

for(int j=0;j<i;j++)

{

if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='b'||connects[j].check[0]=='B')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Bitem->addChild(a);

}

else if(connects[j].check[0]=='c'||connects[j].check[0]=='C')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Citem->addChild(a);

}

else if(connects[j].check[0]=='d'||connects[j].check[0]=='D')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ditem->addChild(a);

}

else if(connects[j].check[0]=='e'||connects[j].check[0]=='E')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Eitem->addChild(a);

}

else if(connects[j].check[0]=='f'||connects[j].check[0]=='F')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Fitem->addChild(a);

}

else if(connects[j].check[0]=='g'||connects[j].check[0]=='G')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Gitem->addChild(a);

}

else if(connects[j].check[0]=='h'||connects[j].check[0]=='H')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Hitem->addChild(a);

}

else if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='i'||connects[j].check[0]=='I')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Iitem->addChild(a);

}

else if(connects[j].check[0]=='j'||connects[j].check[0]=='J')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

ui->treeWidget->addTopLevelItem(Qitem);

QTreeWidgetItem \* Ritem = new QTreeWidgetItem(QStringList()<<"R");

ui->treeWidget->addTopLevelItem(Ritem);

QTreeWidgetItem \* Sitem = new QTreeWidgetItem(QStringList()<<"S");

ui->treeWidget->addTopLevelItem(Sitem);

QTreeWidgetItem \* Titem = new QTreeWidgetItem(QStringList()<<"T");

ui->treeWidget->addTopLevelItem(Titem);

QTreeWidgetItem \* Uitem = new QTreeWidgetItem(QStringList()<<"U");

ui->treeWidget->addTopLevelItem(Uitem);

QTreeWidgetItem \* Vitem = new QTreeWidgetItem(QStringList()<<"V");

ui->treeWidget->addTopLevelItem(Vitem);

QTreeWidgetItem \* Witem = new QTreeWidgetItem(QStringList()<<"W");

ui->treeWidget->addTopLevelItem(Witem);

QTreeWidgetItem \* Xitem = new QTreeWidgetItem(QStringList()<<"X");

ui->treeWidget->addTopLevelItem(Xitem);

QTreeWidgetItem \* Yitem = new QTreeWidgetItem(QStringList()<<"Y");

ui->treeWidget->addTopLevelItem(Yitem);

QTreeWidgetItem \* Zitem = new QTreeWidgetItem(QStringList()<<"Z");

ui->treeWidget->addTopLevelItem(Zitem);

//追加子节点

string file=username+"connecters.bat";

ifstream in;

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

i=0;

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

connects[i].currtext=currtext;

connects[i].currname=currname;

connects[i].currnumber=currnumber;

connects[i].curremail=curremail;

connects[i].currcompany=currcompany;

connects[i].curraddress=curraddress;

connects[i].currbirth=currbirth;

connects[i].currsocial=currsocial;

if(!IncludeChinese(connects[i].currname.c\_str()))

{

connects[i].check=connects[i].currname;

}

else

{

CNLHelper \*cnl=new CNLHelper;

Currtext=Currtext.fromStdString(connects[i].currname);

Currtext=cnl->CNToPY(Currtext);

connects[i].check=Currtext.toStdString();

}

i++;

}

int sum=i;

QuickSort(sum);

in.close();

for(int j=0;j<i;j++)

{

if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='b'||connects[j].check[0]=='B')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Bitem->addChild(a);

}

else if(connects[j].check[0]=='c'||connects[j].check[0]=='C')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Citem->addChild(a);

}

else if(connects[j].check[0]=='d'||connects[j].check[0]=='D')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ditem->addChild(a);

}

else if(connects[j].check[0]=='e'||connects[j].check[0]=='E')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Eitem->addChild(a);

}

else if(connects[j].check[0]=='f'||connects[j].check[0]=='F')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Fitem->addChild(a);

}

else if(connects[j].check[0]=='g'||connects[j].check[0]=='G')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Gitem->addChild(a);

}

else if(connects[j].check[0]=='h'||connects[j].check[0]=='H')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Hitem->addChild(a);

}

else if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='i'||connects[j].check[0]=='I')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Iitem->addChild(a);

}

else if(connects[j].check[0]=='j'||connects[j].check[0]=='J')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

connects[i].check=Currtext.toStdString();

}

i++;

}

int sum=i;

QuickSort(sum);

in.close();

for(int j=0;j<i;j++)

{

if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='b'||connects[j].check[0]=='B')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Bitem->addChild(a);

}

else if(connects[j].check[0]=='c'||connects[j].check[0]=='C')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Citem->addChild(a);

}

else if(connects[j].check[0]=='d'||connects[j].check[0]=='D')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ditem->addChild(a);

}

else if(connects[j].check[0]=='e'||connects[j].check[0]=='E')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Eitem->addChild(a);

}

else if(connects[j].check[0]=='f'||connects[j].check[0]=='F')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Fitem->addChild(a);

}

else if(connects[j].check[0]=='g'||connects[j].check[0]=='G')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Gitem->addChild(a);

}

else if(connects[j].check[0]=='h'||connects[j].check[0]=='H')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Hitem->addChild(a);

}

else if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='i'||connects[j].check[0]=='I')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Iitem->addChild(a);

}

else if(connects[j].check[0]=='j'||connects[j].check[0]=='J')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Fitem->addChild(a);

}

else if(connects[j].check[0]=='g'||connects[j].check[0]=='G')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Gitem->addChild(a);

}

else if(connects[j].check[0]=='h'||connects[j].check[0]=='H')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Hitem->addChild(a);

}

else if(connects[j].check[0]=='a'||connects[j].check[0]=='A')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Aitem->addChild(a);

}

else if(connects[j].check[0]=='i'||connects[j].check[0]=='I')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Iitem->addChild(a);

}

else if(connects[j].check[0]=='j'||connects[j].check[0]=='J')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Jitem->addChild(a);

}

else if(connects[j].check[0]=='l'||connects[j].check[0]=='L')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Litem->addChild(a);

}

else if(connects[j].check[0]=='M'||connects[j].check[0]=='m')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Mitem->addChild(a);

}

else if(connects[j].check[0]=='n'||connects[j].check[0]=='N')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Nitem->addChild(a);

}

else if(connects[j].check[0]=='o'||connects[j].check[0]=='O')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Oitem->addChild(a);

}

else if(connects[j].check[0]=='p'||connects[j].check[0]=='P')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Pitem->addChild(a);

}

else if(connects[j].check[0]=='q'||connects[j].check[0]=='Q')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Qitem->addChild(a);

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

else if(connects[j].check[0]=='r'||connects[j].check[0]=='R')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Ritem->addChild(a);

}

else if(connects[j].check[0]=='s'||connects[j].check[0]=='S')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Sitem->addChild(a);

}

else if(connects[j].check[0]=='t'||connects[j].check[0]=='T')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Titem->addChild(a);

}

else if(connects[j].check[0]=='u'||connects[j].check[0]=='U')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Uitem->addChild(a);

}

else if(connects[j].check[0]=='v'||connects[j].check[0]=='V')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Vitem->addChild(a);

}

else if(connects[j].check[0]=='w'||connects[j].check[0]=='W')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Witem->addChild(a);

}

else if(connects[j].check[0]=='x'||connects[j].check[0]=='X')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Xitem->addChild(a);

}

else if(connects[j].check[0]=='y'||connects[j].check[0]=='Y')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Yitem->addChild(a);

}

else if(connects[j].check[0]=='z'||connects[j].check[0]=='Z')

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

Zitem->addChild(a);

}

else

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

signitem->addChild(a);

}

}

string filebest=username+"bestconnexters.bat";

ifstream inbest;

inbest.open(filebest,ios::in);

while(getline(inbest,currtext))//逐行读取，直到结尾或者相等

{

getline(inbest,currname);

getline(inbest,currnumber);

getline(inbest,curremail);

getline(inbest,currcompany);

getline(inbest,curraddress);

getline(inbest,currbirth);

getline(inbest,currsocial);

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(currname)<<Currtext.fromStdString(currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

bestitem->addChild(a);

}

ui->treeWidget->setItemsExpandable(true);

ui->treeWidget->expandAll();

}

void QuickSort(int sum)//排序

{

for(int j=0;j<sum;j++)

{

for(int k=j+1;k<sum;k++)

{

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

if(connects[j].check>connects[k].check)

{

swap(connects[j],connects[k]);

}

}

}

}

int IncludeChinese(const char \*str)//查看是否含有中文

{

char c;

int j=0;

while(1)

{

c=str[j++];

if (c==0) break; //如果到字符串尾则说明该字符串没有中文字符

if (c&0x80) //如果字符高位为1且下一字符高位也是1则有中文字符

if (str[j] & 0x80) return 1;

}

return 0;

}

void MainWindow::on\_pushButton\_4\_clicked()//个人信息

{

Owninformation \*own=new Owninformation(username);

own->show();

this->close();

}

void MainWindow::on\_pushButton\_clicked()//查询按钮实现

{

QString Currtext;

Queryinterface \*query=new Queryinterface(username);

query->ui->treeWidget->clear();

query->ui->treeWidget->setHeaderLabels(QStringList()<<"查询结果"<<" ");

QTreeWidgetItem \* searchitem = new QTreeWidgetItem(QStringList()<<"联系人"<<"电话号");

query->ui->treeWidget->addTopLevelItem(searchitem);

QString Searchname=ui->lineEdit->text();

string searchname=Searchname.toStdString();

for(int j=0;j<i;j++)

{

if(connects[j].currname.find(searchname)!=connects[j].currname.npos||connects[j].currnumber.find(searchname)!=connects[j].currnumber.npos)

{

QStringList lianxirena1;

lianxirena1<<Currtext.fromStdString(connects[j].currname)<<Currtext.fromStdString(connects[j].currnumber);

QTreeWidgetItem \* a = new QTreeWidgetItem(lianxirena1);

searchitem->addChild(a);

}

}

query->show();

query->ui->treeWidget->setItemsExpandable(true);

query->ui->treeWidget->expandAll();

this->close();

}

源文件 newperson.cpp

#include "newperson.h"

#include "ui\_newperson.h"

#include "cstring"

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include "Login.h"

Newperson::Newperson(string username,QWidget \*parent)

: QMainWindow(parent),currentname(username)

, ui(new Ui::Newperson)

{

ui->setupUi(this);

}

Newperson::~Newperson()

{

delete ui;

}

void Newperson::on\_pushButton\_clicked()//新建联系人实现

{

QString Usertext=ui->textEdit->toPlainText();

QString Name=ui->lineEdit\_3->text();

QString Number=ui->lineEdit\_4->text();

QString Email=ui->lineEdit\_5->text();

QString Company=ui->lineEdit\_6->text();

QString Address=ui->lineEdit\_7->text();

QString Birth=ui->lineEdit\_8->text();

QString Social=ui->lineEdit\_9->text();

string usertext=Usertext.toStdString();

string name=Name.toStdString();

string number=Number.toStdString();

string email=Email.toStdString();

string company=Company.toStdString();

string address=Address.toStdString();

string birth=Birth.toStdString();

string social=Social.toStdString();

if(Newperson::checknumber(number))

{

ofstream out;

string file=currentname+"connecters.bat";

out.open(file,ios::app);//定位到文件的末尾

out<<usertext<<"\n"<<name<<"\n"<<number<<"\n"<<email<<"\n"<<company<<"\n"<<address<<"\n"<<birth<<"\n"<<social<<"\n";

out.close();

QMessageBox::information(this,"提示","新建联系人成功");

this->close();

MainWindow \*main=new MainWindow(currentname);

main->ini();

main->show();

}

else

{

QMessageBox::information(this,"提示","电话号码格式不正确");

}

}

void Newperson::on\_pushButton\_2\_clicked()//退出实现

{

this->close();

MainWindow \*main=new MainWindow(currentname);

main->ini();

main->show();

this->close();

}

int Newperson::checknumber(string number)

{

if(number.length()==11||number.length()==8)

{

for(int i=0;i<number.length();i++)

{

if(!isdigit(number[i]))

{

return 0;

}

}

}

else

{

return 0;

}

return 1;

}

QMessageBox::information(this,"提示","电话号码格式不正确");

}

}

void Newperson::on\_pushButton\_2\_clicked()//退出实现

{

this->close();

MainWindow \*main=new MainWindow(currentname);

main->ini();

main->show();

this->close();

}

int Newperson::checknumber(string number)

{

if(number.length()==11||number.length()==8)

{

for(int i=0;i<number.length();i++)

{

if(!isdigit(number[i]))

{

return 0;

}

}

}

else

{

return 0;

}

return 1;

}

源文件 owninformation.cpp

#include "owninformation.h"

#include "ui\_owninformation.h"

#include "Login.h"

#include "mainwindow.h"

#include "ui\_mainwindow.h"

Owninformation::Owninformation(string name,QWidget \*parent)

: QMainWindow(parent),username(name)

, ui(new Ui::Owninformation)

{

ui->setupUi(this);

QString Username\_c=Username\_c.fromStdString(username);

this->setWindowTitle(Username\_c);

QString Username=Username.fromStdString(username);

Owninformation::setToolTip(Username);

ifstream in;

string file=username+"owninformation.bat";

in.open(file,ios::in);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

QString Currtext="";

while(getline(in,currtext))//逐行读取，直到结尾或者相等

{

getline(in,currname);

getline(in,currnumber);

getline(in,curremail);

getline(in,currcompany);

getline(in,curraddress);

getline(in,currbirth);

getline(in,currsocial);

ui->textEdit->setPlainText(Currtext.fromStdString(currtext));

ui->lineEdit\_address->setText(Currtext.fromStdString(curraddress));

ui->lineEdit\_birth->setText(Currtext.fromStdString(currbirth));

ui->lineEdit\_company->setText(Currtext.fromStdString(currcompany));

ui->lineEdit\_email->setText(Currtext.fromStdString(curremail));

ui->lineEdit\_name->setText(Currtext.fromStdString(currname));

ui->lineEdit\_number->setText(Currtext.fromStdString(currnumber));

ui->lineEdit\_social->setText(Currtext.fromStdString(currsocial));

}

}

Owninformation::~Owninformation()

{

delete ui;

}

void Owninformation::on\_pushButton\_clicked()//保存按钮实现

{

string file=username+"owninformation.bat";

string file\_new=username+"owninformationnew.bat";

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Owninformation::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

ifstream in;

in.open(file);

ofstream out;

out.open(file\_new,ios::trunc);

string currtext="";

string currname="";

string currnumber="";

string curremail="";

string currcompany="";

string curraddress="";

string currbirth="";

string currsocial="";

out<<ui->textEdit->toPlainText().toStdString()<<"\n"<<ui->lineEdit\_name->text().toStdString()

<<"\n"<<ui->lineEdit\_number->text().toStdString()<<"\n"<<ui->lineEdit\_email->text().toStdString()

<<"\n"<<ui->lineEdit\_company->text().toStdString()<<"\n"<<ui->lineEdit\_address->text().toStdString()

<<"\n"<<ui->lineEdit\_birth->text().toStdString()<<"\n"<<ui->lineEdit\_social->text().toStdString()<<"\n";

in.close();

out.close();

const char \* file\_f;

const char \*filenew\_f;

file\_f=file.c\_str();

filenew\_f=file\_new.c\_str();

remove(file\_f);

rename(filenew\_f,file\_f);

QMessageBox::information(this,"提示","修改成功");

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

void Owninformation::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

}

源文件queryinterface.cpp

#include "queryinterface.h"

#include "ui\_queryinterface.h"

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include "detailpage.h"

#include "ui\_detailpage.h"

Queryinterface::Queryinterface(string name,QWidget \*parent)

: QMainWindow(parent),username(name)

, ui(new Ui::Queryinterface)

{

ui->setupUi(this);

connect(ui->treeWidget, &QTreeWidget::itemDoubleClicked, this, &Queryinterface::on\_treeWidget\_itemDoubleClicked);

}

Queryinterface::~Queryinterface()

{

delete ui;

}

void Queryinterface::on\_pushButton\_3\_clicked()//退出按钮实现

{

this->close();

MainWindow \*main=new MainWindow(username);

main->ini();

main->show();

this->close();

}

void Queryinterface::on\_treeWidget\_itemDoubleClicked(QTreeWidgetItem \*item, int column)//获取子项信息的槽函数实现

{

QString number="";

QString name="";

// 检查所选项是否为子项

if (item->parent() != 0) {

// 获取子项信息

if(column==1)

{

number=item->text(column);

name=item->text(column-1);

}

if(column==0)

{

number=item->text(column+1);

name=item->text(column);

}

Detailpage \*detail=new Detailpage(name.toStdString(),number.toStdString(),username,this);

detail->setWindowModality(Qt::WindowModal);

this->close();

detail->show();

}

}

源文件retrieverpassword.cpp

#include "retrieverpassword.h"

#include "ui\_retrieverpassword.h"

#include "Login.h"

using namespace std;

Retrieverpassword::Retrieverpassword(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Retrieverpassword)

{

ui->setupUi(this);

}

Retrieverpassword::~Retrieverpassword()

{

delete ui;

}

void Retrieverpassword::on\_pushButton\_clicked()//修改密码实现

{

QString ANS2=ui->lineEdit\_6->text();

QString Password=ui->lineEdit\_3->text();

QString Password\_confirm=ui->lineEdit\_4->text();

QString str1=ui->lineEdit\_5->text();

QString ANS1=ui->lineEdit\_6->text();

QString Name=ui->lineEdit\_2->text();

QString Email=ui->lineEdit->text();

string str\_2=str1.toStdString();

string str\_1=str1.toStdString();

string line6=ANS1.toStdString();

string line7=ANS2.toStdString();

string password=Password.toStdString();

string password\_confirm=Password\_confirm.toStdString();

string name=Name.toStdString();

string email=Email.toStdString();

int ans6;

int ans7;

sscanf(line6.c\_str(), "%d", &ans6);//读取文本框中用户写的答案

sscanf(line7.c\_str(), "%d", &ans7);//读取文本框中用户写的答案

if(!check(name))

{

QMessageBox::information(this,"提示","没有此用户");

}

else if(ans6==compare(str\_1)&&ans7==compare(str\_2)&&password==password\_confirm)

{

writeUser(name,email,password);//写入新的密码

QMessageBox::information(this,"提示","更改成功");

this->close();

}

else if(ans6!=compare(str\_1)||ans7!=compare(str\_2))

{

QMessageBox::warning(this,"警告","你的计算不如小学生");

}

else if(password!=password\_confirm)

{

QMessageBox::warning(this,"警告","两次密码不一样");

}

}

int compare(string str)//验证码随机数加减乘

{

int ans;

if(str[1]=='\*')

{

ans=(str[0]-'0')\*(str[2]-'0');

}

if(str[1]=='-')

{

ans=(str[0]-'0')-(str[2]-'0');

}

if(str[1]=='+')

{

ans=(str[0]-'0')+(str[2]-'0');

}

return ans;

}

void writeUser(string name,string email,string password)//写入改User文件

{

ifstream in("User.bat");

ofstream out;

out.open("Usernew.bat",ios::trunc); //ios::trunc表示在打开文件前将文件清空,由于是写入,文件不存在则创建

string name\_f,email\_f,password\_f;

while(getline(in,name\_f))//着行读取数据并存于namesc中，直至数据全部读取

{

getline(in,password\_f);

getline(in,email\_f);

if(name\_f!=name)

{

out<<name\_f<<"\n"<<password\_f<<"\n"<<email\_f<<"\n";

}

}

out<<name<<"\n"<<password<<"\n"<<email<<"\n";

in.close();

out.close();

remove("User.bat");

rename("Usernew.bat", "User.bat");

}

void Retrieverpassword::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

}

int check(string namebase)//比较输入的信息和用户信息是否相同

{

ifstream in;

in.open("User.bat");

string name\_f,password\_f,email\_f;

while(getline(in,name\_f))

{

getline(in,password\_f);

getline(in,email\_f);

if(name\_f==namebase)

{

in.close();

return 1;

}

}

in.close();

return 0;

}

QMessageBox::warning(this,"警告","两次密码不一样");

}

}

int compare(string str)//验证码随机数加减乘

{

int ans;

if(str[1]=='\*')

{

ans=(str[0]-'0')\*(str[2]-'0');

}

if(str[1]=='-')

{

ans=(str[0]-'0')-(str[2]-'0');

}

if(str[1]=='+')

{

ans=(str[0]-'0')+(str[2]-'0');

}

return ans;

}

void writeUser(string name,string email,string password)//写入改User文件

{

ifstream in("User.bat");

ofstream out;

out.open("Usernew.bat",ios::trunc); //ios::trunc表示在打开文件前将文件清空,由于是写入,文件不存在则创建

string name\_f,email\_f,password\_f;

while(getline(in,name\_f))//着行读取数据并存于namesc中，直至数据全部读取

{

getline(in,password\_f);

getline(in,email\_f);

if(name\_f!=name)

{

out<<name\_f<<"\n"<<password\_f<<"\n"<<email\_f<<"\n";

}

}

out<<name<<"\n"<<password<<"\n"<<email<<"\n";

in.close();

out.close();

remove("User.bat");

rename("Usernew.bat", "User.bat");

}

void Retrieverpassword::on\_pushButton\_2\_clicked()//退出按钮实现

{

this->close();

}

int check(string namebase)//比较输入的信息和用户信息是否相同

{

ifstream in;

in.open("User.bat");

string name\_f,password\_f,email\_f;

while(getline(in,name\_f))

{

getline(in,password\_f);

getline(in,email\_f);

if(name\_f==namebase)

{

in.close();

return 1;

}

}

in.close();

return 0;

}

if(name\_f==namebase)

{

in.close();

return 1;

}

}

in.close();

return 0;

}

源文件 signup.cpp

#include "signup.h"

#include "ui\_signup.h"

#include "mainwindow.h"

#include "Login.h"

using namespace std;

Signup::Signup(QWidget \*parent)

: QWidget(parent)

, ui(new Ui::Signup)

{

ui->setupUi(this);

}

Signup::~Signup()

{

delete ui;

}

void Signup::on\_pushButton\_clicked()//获取创建用户的用户信息

{

//获取信息

QString Name=ui->lineEdit->text();

QString Email=ui->lineEdit\_2->text();

QString Password=ui->lineEdit\_3->text();

QString Confirmpassword=ui->lineEdit\_4->text();

string name=Name.toStdString();

string email=Email.toStdString();

string password=Password.toStdString();

string confirmpassword=Confirmpassword.toStdString();

char name\_c[100];

strcpy(name\_c,name.c\_str());

if(checkusername(name))

{

QMessageBox::information(this,"提示","用户名已存在");

}

else

{

if(password!=""&&name!=""&&email!="")

{

if(password==confirmpassword)//若相同，记录users的信息

{

const char \*name\_f=name.c\_str();

const char \*password\_f=password.c\_str();

const char \*email\_f=email.c\_str();

FILE \*file;

file=fopen("User.bat","ab+");

fputs(name\_f,file);

fprintf(file,"\n");

fputs(password\_f,file);

fprintf(file,"\n");

fputs(email\_f,file);

fprintf(file,"\n");

fclose(file);

QMessageBox::information(this,"提示","创建成功");

this->close();

}

else

{

QMessageBox::warning(this,"警告","两次输入密码不正确");

}

}

else

{

QMessageBox::warning(this,"警告","输入信息不能为空");

}

}

}

void Signup::on\_pushButton\_2\_clicked()//退出界面

{

Signup::close();

}

int checkusername(string name)//查看文件里是否有相同的用户名

{

ifstream in("User.bat");

string namesc,emailsc,passwordsc;

while(getline(in,namesc))//着行读取数据并存于namesc中，直至数据全部读取

{

getline(in,emailsc);

getline(in,passwordsc);

if(namesc==name)

{

return 1;

}

}

return 0;

}

file=fopen("User.bat","ab+");

fputs(name\_f,file);

fprintf(file,"\n");

fputs(password\_f,file);

fprintf(file,"\n");

fputs(email\_f,file);

fprintf(file,"\n");

fclose(file);

QMessageBox::information(this,"提示","创建成功");

this->close();

}

else

{

QMessageBox::warning(this,"警告","两次输入密码不正确");

}

}

else

{

QMessageBox::warning(this,"警告","输入信息不能为空");

}

}

}

void Signup::on\_pushButton\_2\_clicked()//退出界面

{

Signup::close();

}

int checkusername(string name)//查看文件里是否有相同的用户名

{

ifstream in("User.bat");

string namesc,emailsc,passwordsc;

while(getline(in,namesc))//着行读取数据并存于namesc中，直至数据全部读取

{

getline(in,emailsc);

getline(in,passwordsc);

if(namesc==name)

{

return 1;

}

}

return 0;

}