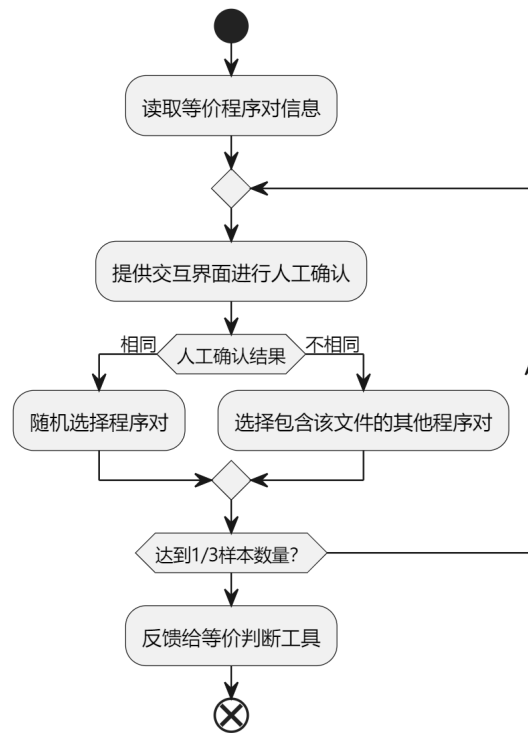


# 实验报告——实验5：交互式确认方法

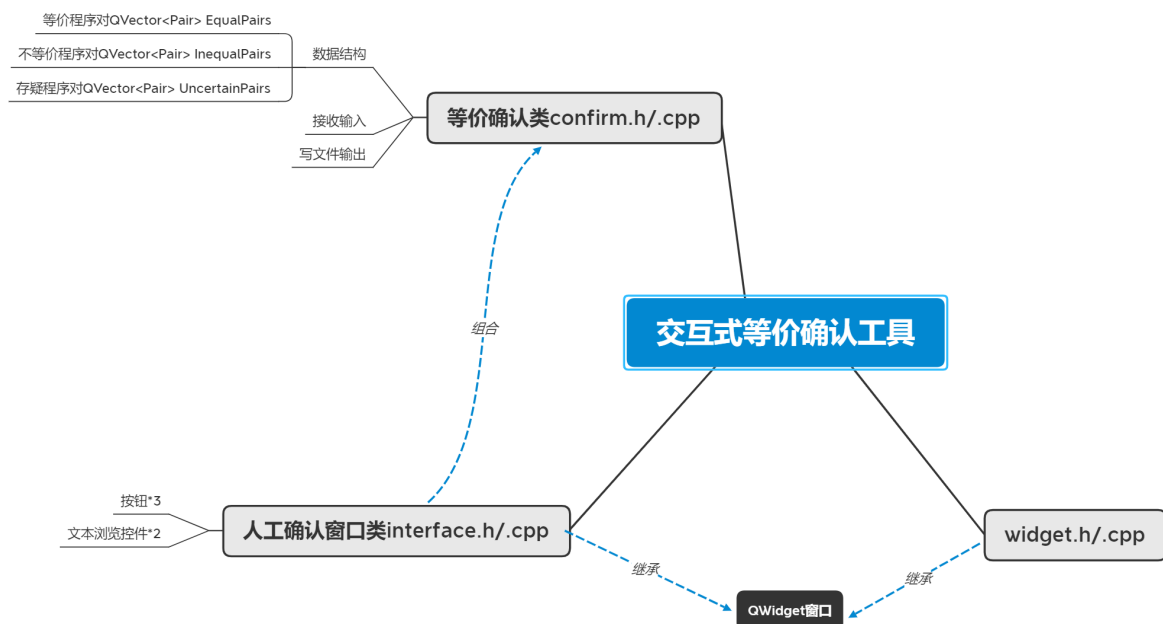
201220086 景雅

## 一、总体设计

参考实验3中的等价确认工具执行活动图，完成实验5中的辅助等价性确认GUI工具。GUI工具的设计思路与流程如下图所示。



针对上述流程进行模块划分，共分成三个类，总体结构与关系如下图所示。其中，等价确认类Confirm储存程序对的等价性信息，涉及处理程序对的输入、自动推荐待判断程序对、输出人工确认结果。人工确认窗口类Interface具有图形化界面，为用户提供了交互选项，通过按钮的槽函数反馈等价信息。最外层窗口类Widget判断输入文件是否存在，并决定是否进入Interface类中。



## 二、功能实现

### 1、接收输入

程序中使用结构体存储等价程序对信息，包含两个程序的路径名称和一个布尔变量，标志是否被人工确认过。

```
struct Pair
{
    QString file1;
    QString file2;
    bool flag; //是否被人工确认过
};
```

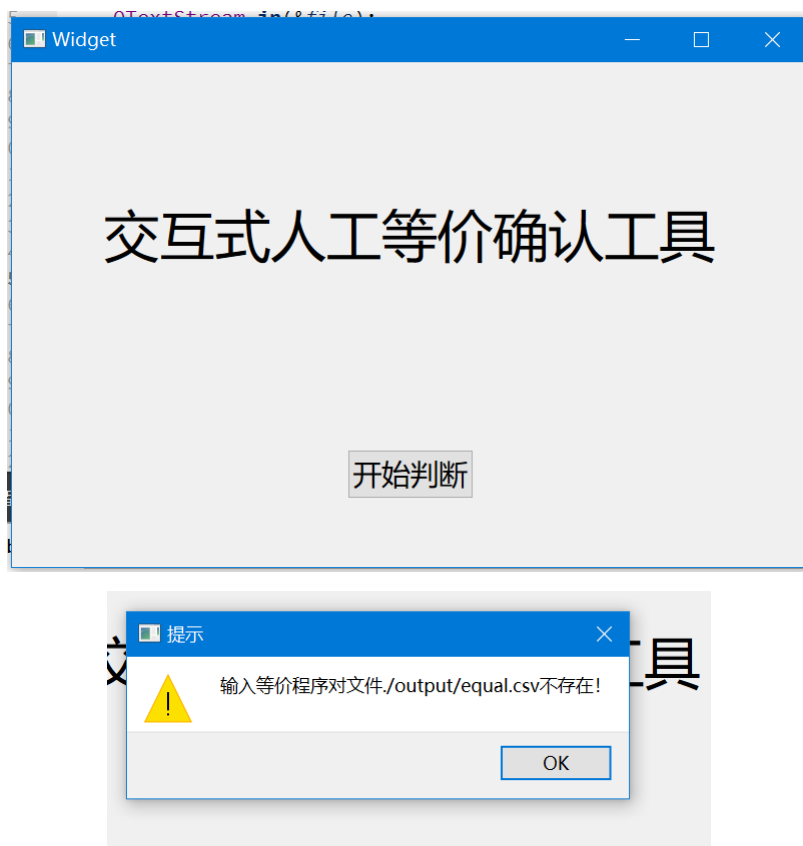
在等价确认类中使用Qt的通用容器类QVector，分别存储人工确认后等价程序对、不等价程序对、存疑程序对，它们是等价确认类中的私有变量。

```
QVector<Pair> EqualPairs;
QVector<Pair> InequalPairs;
QVector<Pair> UncertainPairs;
```

接收输入时，使用Qt中的文件流QTextStream读取实验4生成的output/equal.csv文件内容，将等价程序对信息添加到上面的等价程序对容器中，即 `EqualPairs.append({data_list[0], data_list[1], false});`，其中data\_list使用readLine()和split()函数先读取一行内容，再根据逗号分割获得。

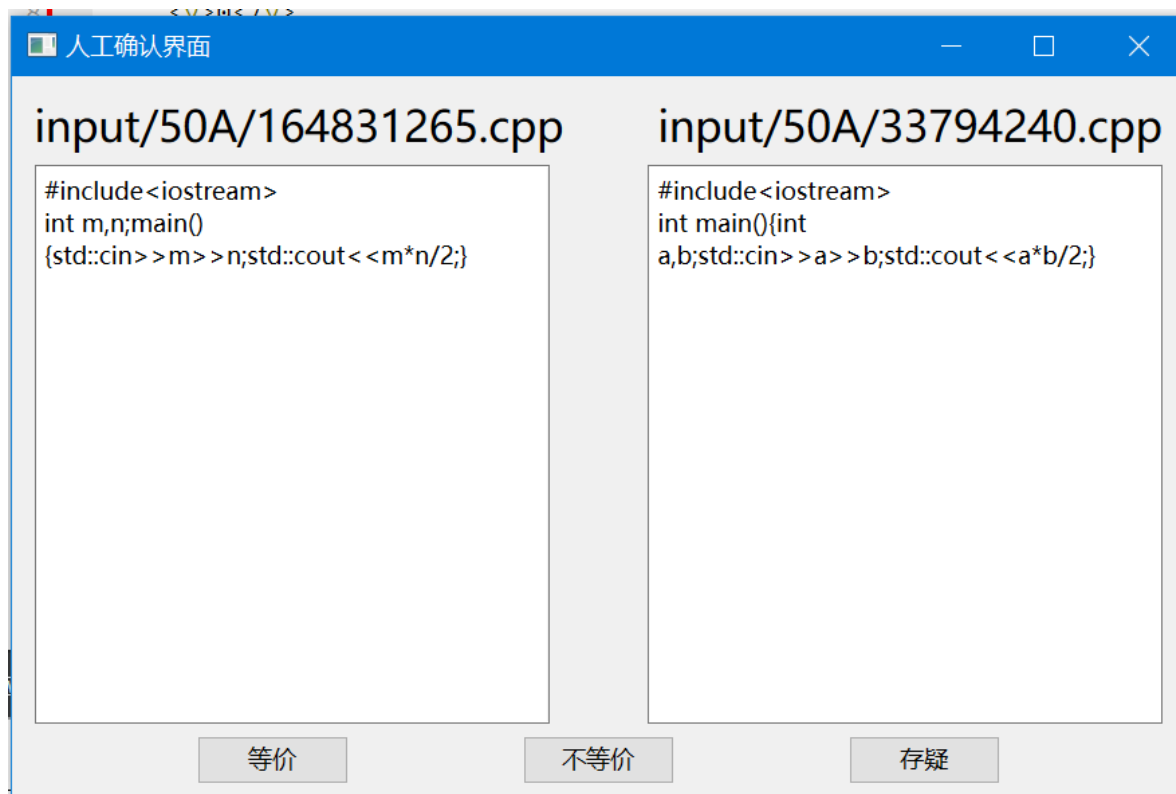
### 2、图形化界面

进入程序后，系统会显示如下开始界面，点击“开始判断”按钮，程序会检查是否存在output/equal.csv文件，存在直接进入判断界面，不存在则通过QMessageBox弹出对话框，显示报错信息。

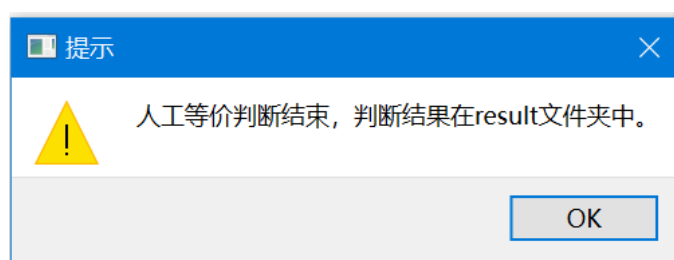


人工确认界面如下图所示，系统会随机选择需要确认的等价程序对，并将内容分栏显示。最下层有“等价”、“不等价”、“存疑”三个确认按钮，按下对应按钮会将相应结果返回给程序，系统自动推荐下一个需要判断的等价对。

此处的自动推荐实现时，考虑一个程序若人工判定与机器判定的结果不一致，即人工认为其是“不等价”或“存疑”的，我们有理由怀疑包含这两个程序的其他等价对也可能会出现判定失误，因此程序的自动推荐会遍历选择其他包含这两个程序之一、且没有被确认过的等价程序对。若判定结果为“等价”，则依然采用随机数生成。

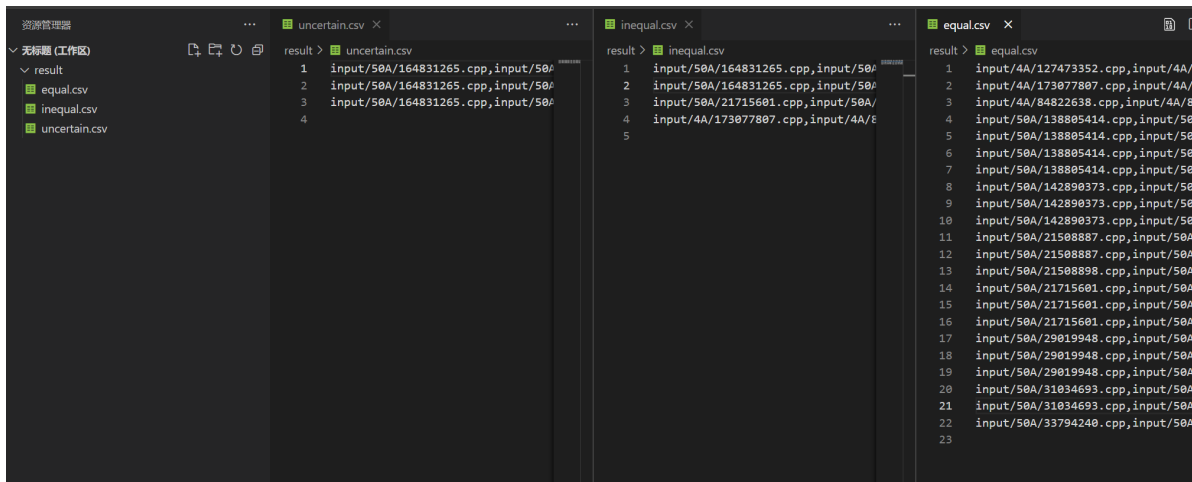


在实验2的软件需求分析中，可知此处等价确认工具是对等价判断工具功能的补充，且程序数量过多时，难以人工确认所有等价程序对。因此，这里设计时只需人工确认总数的三分之一，确认数量达到时，程序会弹出对话框，点击OK后，自动结束程序。最终，用户可以到result文件夹中查看等价确认后的结果。



#### 4、输出结果

每次用户确认结束后，修改三个QVector容器，并进行文件的读写，以防数据丢失。输出结果格式同实验4中的等价判断工具，如下图所示。



### 三、git操作

#### 1、git add

将文件在工作区的修改添加到暂存区，git add操作执行结束后，修改后的内容变成可提交的状态。

```
PS D:\Qt_project\SElab5> git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   confirm.cpp

no changes added to commit (use "git add" and/or "git commit -a")
PS D:\Qt_project\SElab5> git add confirm.cpp
PS D:\Qt_project\SElab5> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   confirm.cpp
```

#### 2、git commit -m "XXX"

git commit提交修改，将暂存区的内容提交到当前分支，提交后当前工作树没有修改了的内容(clean)。

```
PS D:\Qt_project\SElab5> git commit -m "change confirm.cpp"
[master 6caa9e0] change confirm.cpp
 1 file changed, 1 insertion(+), 2 deletions(-)
PS D:\Qt_project\SElab5> git status
On branch master
nothing to commit, working tree clean
```

#### 3、git reset (回退) 和git revert (反做)

两者均可实现将当前版本变为之前版本的功能，但是实现方式大不相同，通过git log显示版本可以查看两者的差异。

git reset是回到某次提交，提交及之前的commit都会被保留，但是此commit id之后的修改都会被删除。

```
PS D:\Qt_project\SElab5> git log --pretty=oneline
8ed8b52ef012d8b0d33091d8f1a17f0cd2133692 (HEAD -> master) final version
4b8d77b45db8ca0e66351b82c776248a08dc207b change confirm.cpp
d7119783f3c35b49750fcec4b209940c7b323ed5 first version
PS D:\Qt_project\SElab5> git reset --soft HEAD^
PS D:\Qt_project\SElab5> git log --pretty=oneline
4b8d77b45db8ca0e66351b82c776248a08dc207b (HEAD -> master) change confirm.cpp
d7119783f3c35b49750fcec4b209940c7b323ed5 first version
```

git revert是用一次新的commit来回滚之前的commit，此次提交之前的commit都会被保留。

```
PS D:\Qt_project\SElab5> git log --pretty=oneline
8ed8b52ef012d8b0d33091d8f1a17f0cd2133692 (HEAD -> master) final version
4b8d77b45db8ca0e66351b82c776248a08dc207b change confirm.cpp
d7119783f3c35b49750fcec4b209940c7b323ed5 first version
PS D:\Qt_project\SElab5> git revert 4b8d77
[master 8dd24b7] Revert "change confirm.cpp"
1 file changed, 2 insertions(+), 1 deletion(-)
PS D:\Qt_project\SElab5> git log --pretty=oneline
8dd24b7e99eded79737ae0e919de85872168f32a (HEAD -> master) Revert "change confirm.cpp"
8ed8b52ef012d8b0d33091d8f1a17f0cd2133692 final version
4b8d77b45db8ca0e66351b82c776248a08dc207b change confirm.cpp
d7119783f3c35b49750fcec4b209940c7b323ed5 first version
```

## 4、远程仓库

`git push origin master` 将本地master分支push到远程仓库

远程仓库界面截图如下：

Surprise-Jing / se\_lab5 Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags

Go to file Add file <> Code

File	Commit Message	Time
input	first version	1 hour ago
output	first version	1 hour ago
.gitignore	first version	1 hour ago
SElab5.pro	first version	1 hour ago
confirm.cpp	Revert "change confirm.cpp"	1 hour ago
confirm.h	final version	1 hour ago
interface.cpp	first version	1 hour ago
interface.h	final version	1 hour ago
interface.ui	first version	1 hour ago
main.cpp	first version	1 hour ago
widget.cpp	first version	1 hour ago
widget.h	first version	1 hour ago
widget.ui	first version	1 hour ago

Help people interested in this repository understand your project by adding a README. Add a README

About: nju软件工程实验5: 交互式确认方法

Releases: No releases published. Create a new release

Packages: No packages published. Publish your first package

Languages: C++ 92.5%, QMake 7.5%