**信息检索实验报告**

**——倒排索引**

实验编号 一

实验名称 倒排索引

班级 软件学院14级二班

学号-姓名 刘嘉洋-1412620

实验日期 2016年9月23日

至 2016年10月13日

评分教师 实验报告成绩

评分日期 年 月 日

**一、实验目的：**

* 了解信息检索原理的基本思路
* 掌握倒排索引的基础构建过程

**二、实验环境**

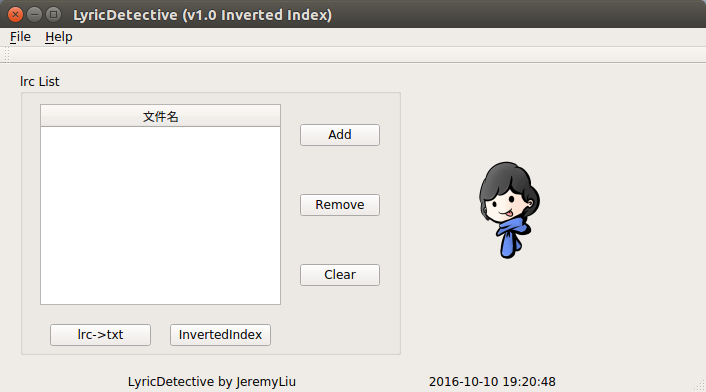
Ubuntu v16.04

Qt Creator 4.0.2

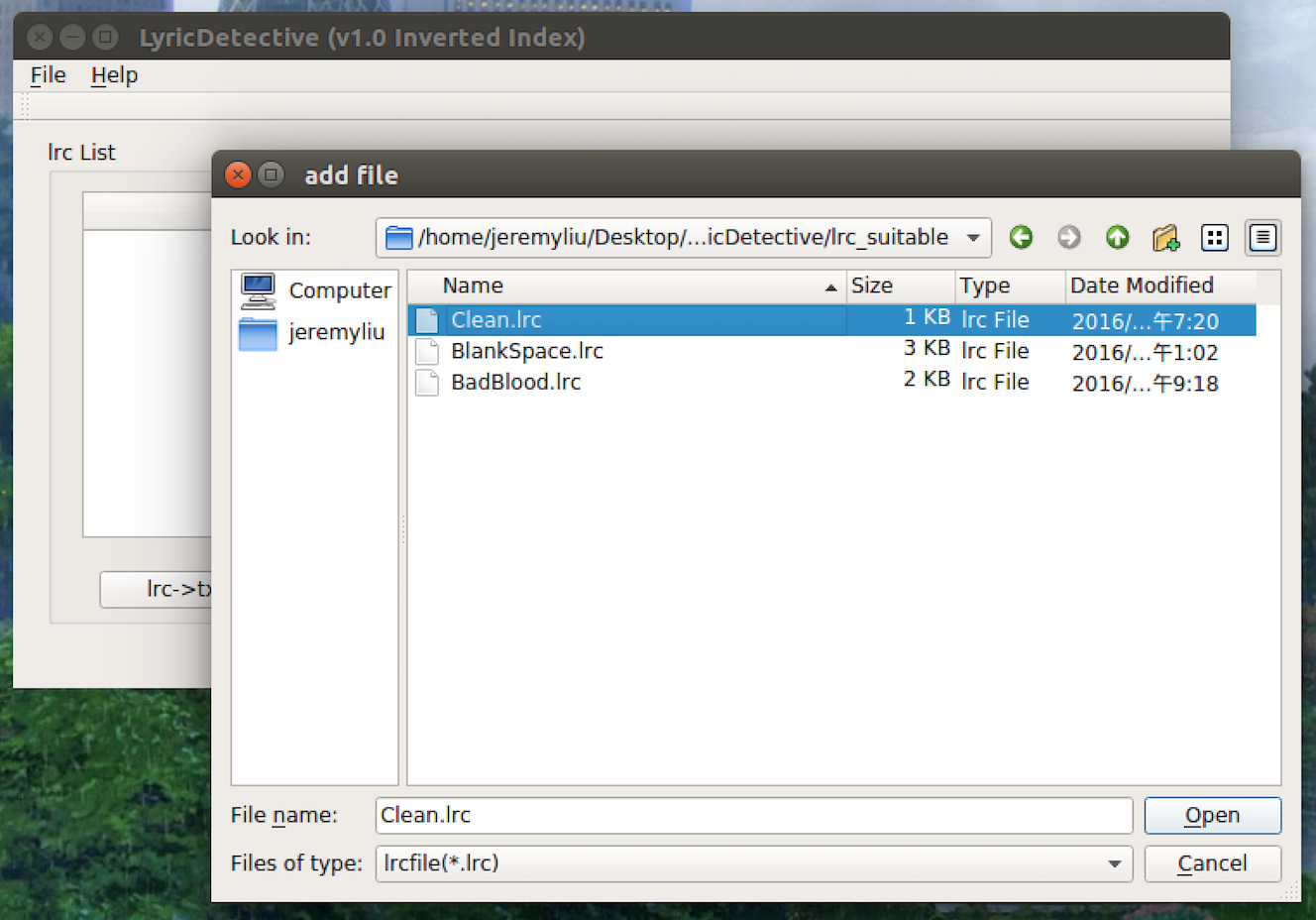
**三、实验要求**

* 编写程序实现为给定目录下lrc(歌词)文件建立倒排索引文件
* 要求过滤掉lrc文件中的标识和时间标签

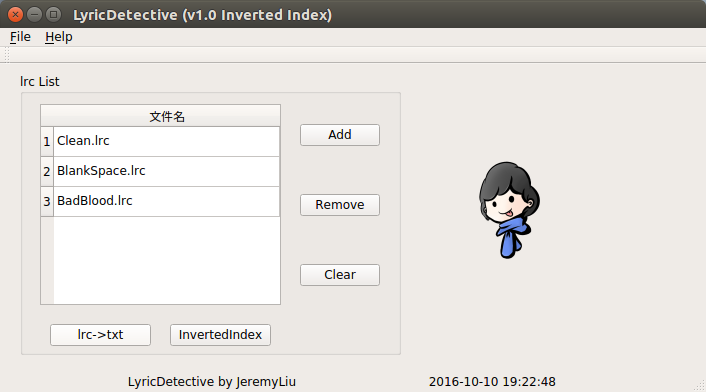
**四、软件操作展示**



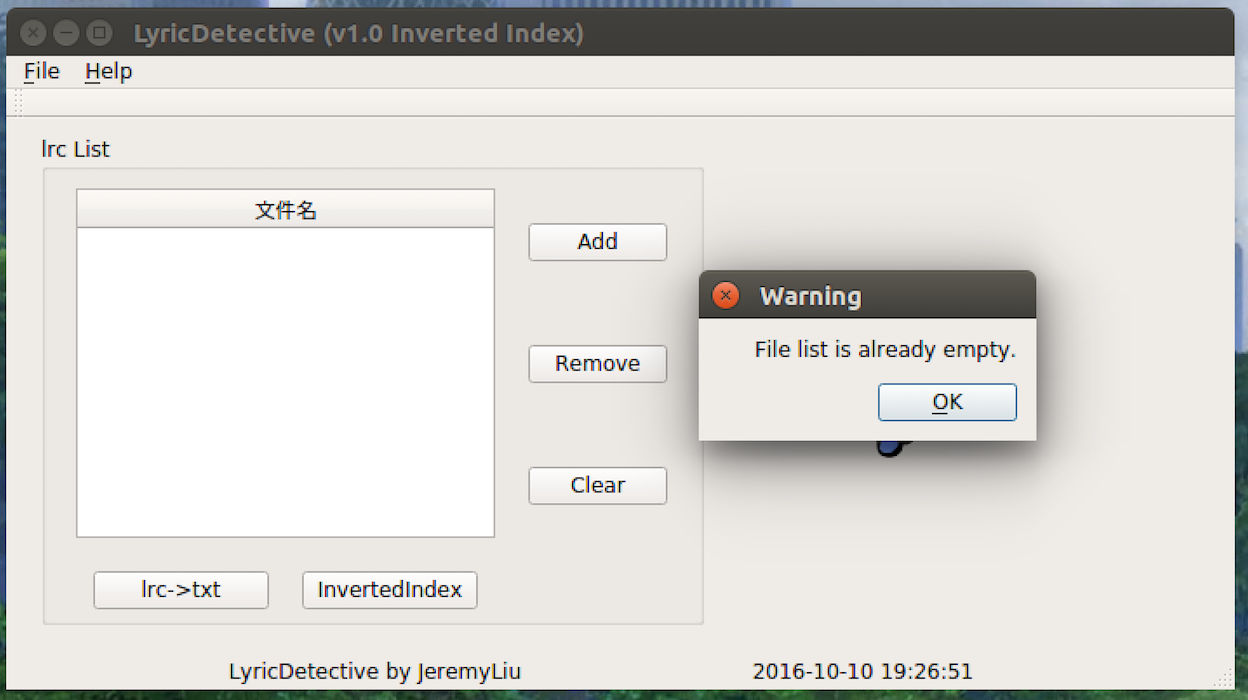
[图1] LyricDetective“歌词探长”界面



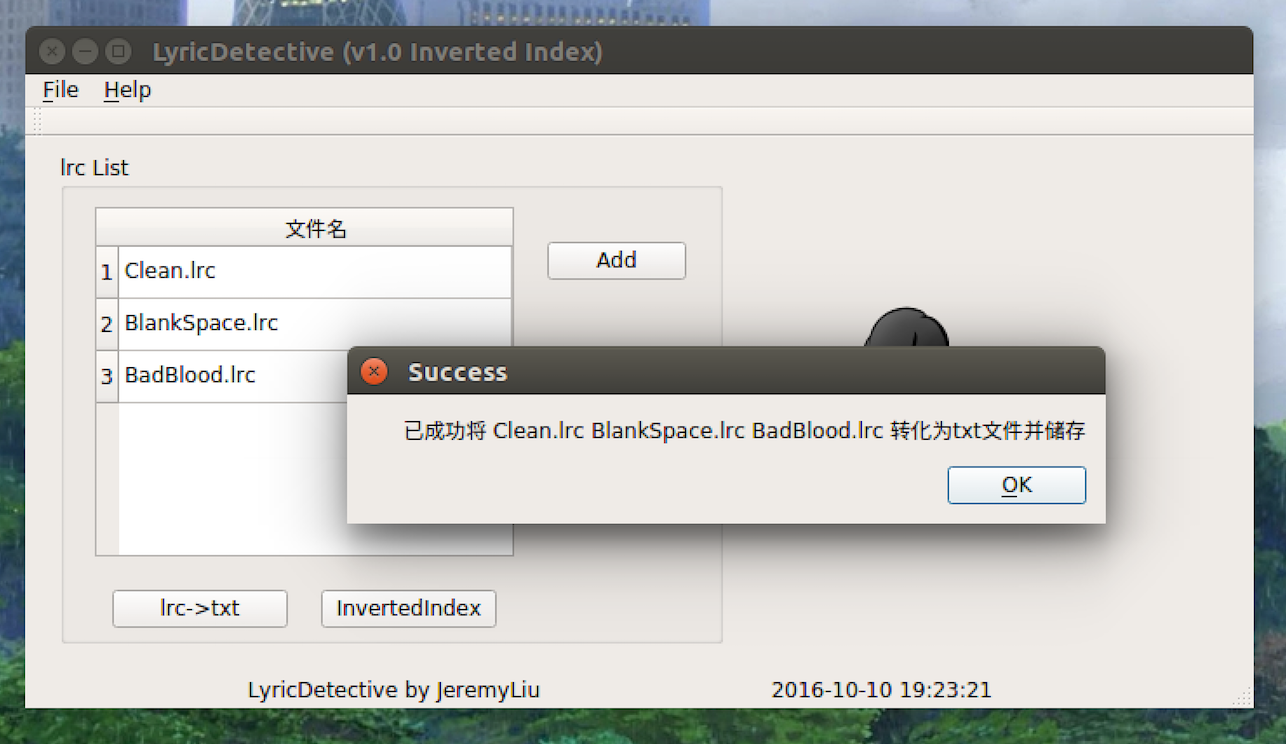
[图2] 点击”ADD”按钮或通过File菜单中的”Add .lrc”显示添加歌词文件界面



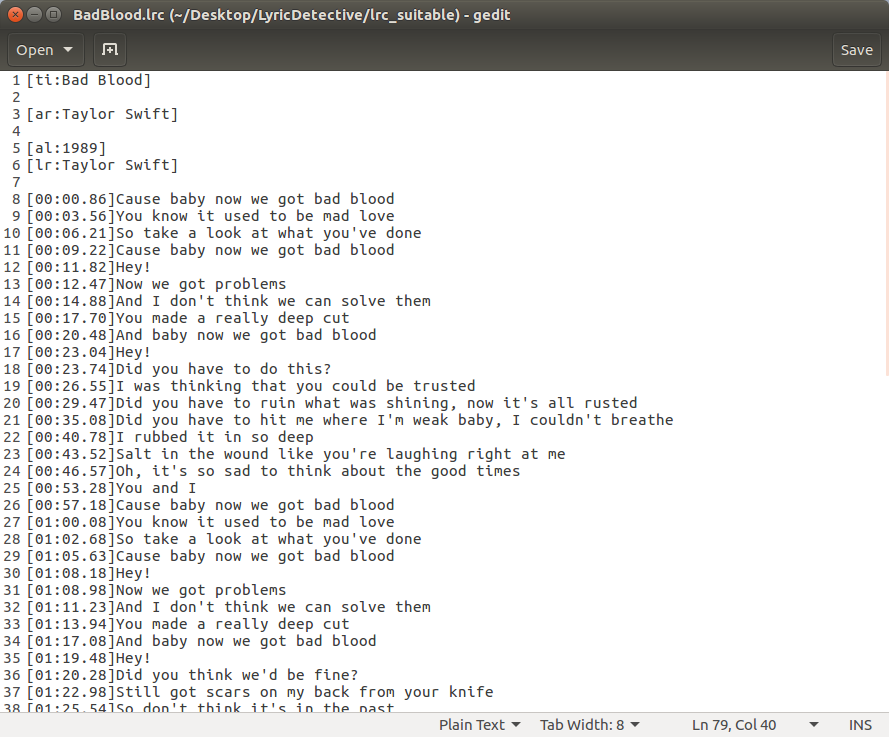
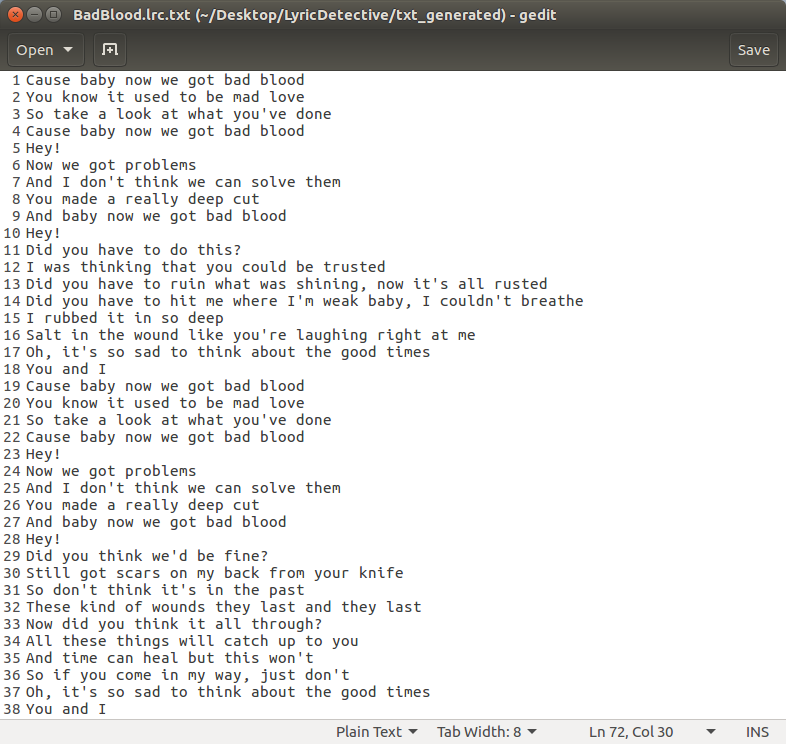
[图3] 完成.lrc文件列表创建，其间可进行添加、移除、清空操作



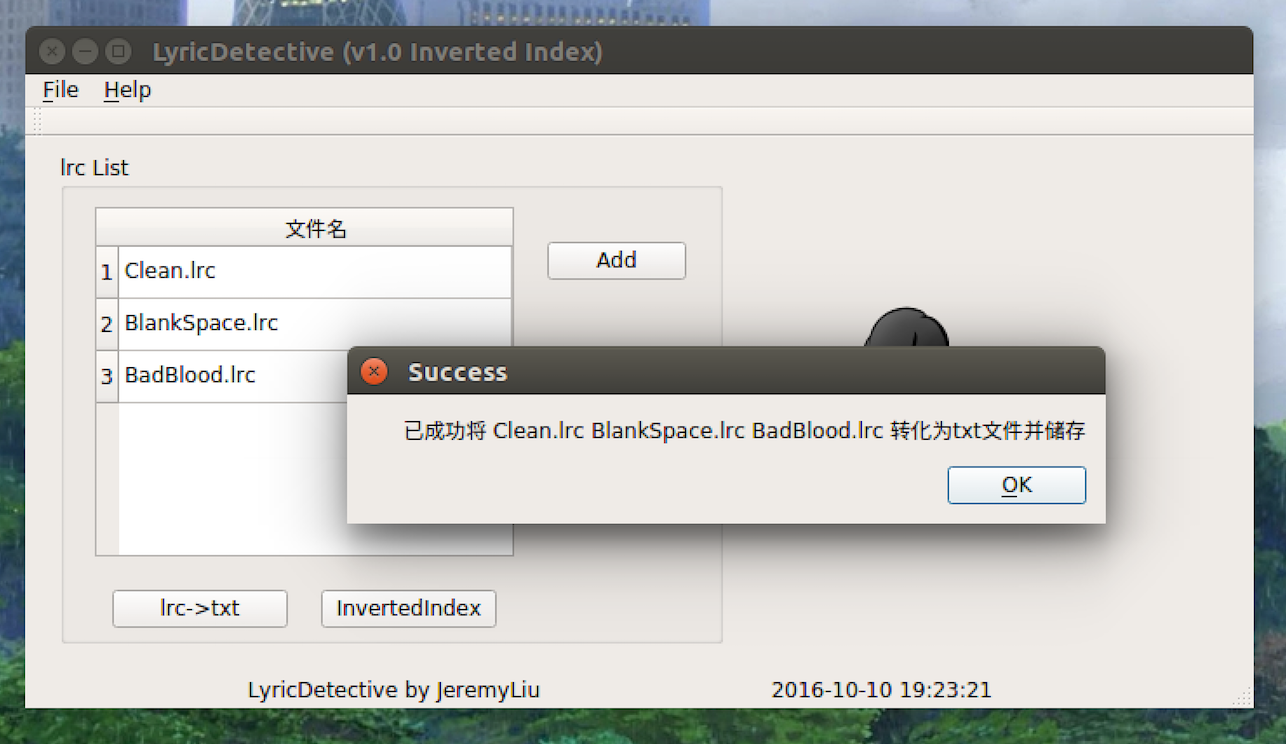
[图4] 移除或清空后若再进行删除操作，会弹出提示信息



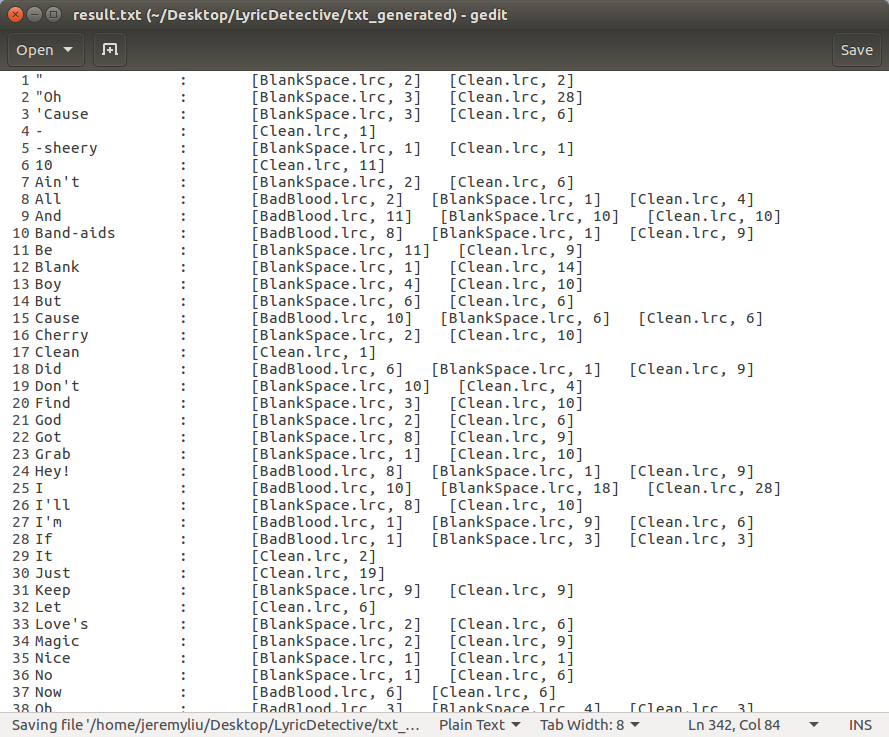
[图5] 点击”lrc->txt”按钮，将当前歌词列表进行标识与时间标签分离

[图6.a] 原.lrc文件格式样例 [图6.b] 分析过滤后生成对应.txt样例



[图7] 点击”InvertedIndex”按钮，得到生成的倒排索引结果输出到result.txt中



[图8] 倒排索引生成结果result.txt

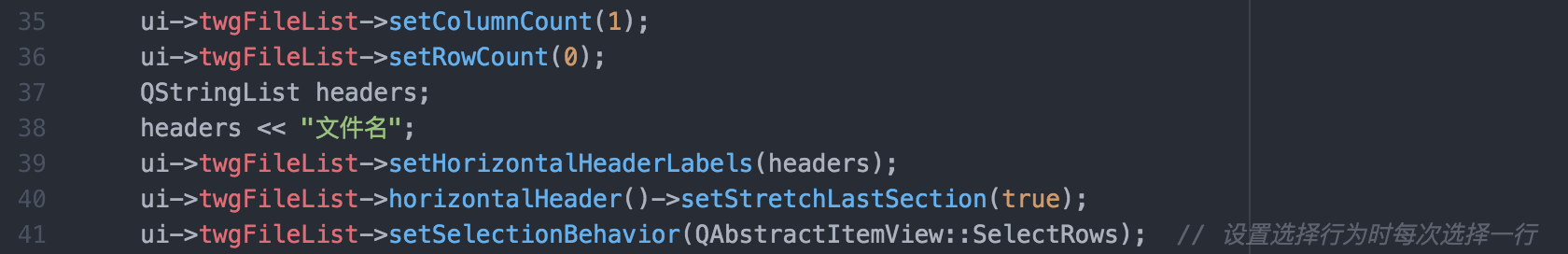
**五、核心代码说明**

该软件需要解决的核心问题有：

* QTableWidget中.lrc列表的增、删和清空
* .lrc文件中标识和时间标签的过滤
* 倒排索引的生成
* 生成结果的存储数据结构

5.1 QTableWidget中.lrc列表的增、删和清空

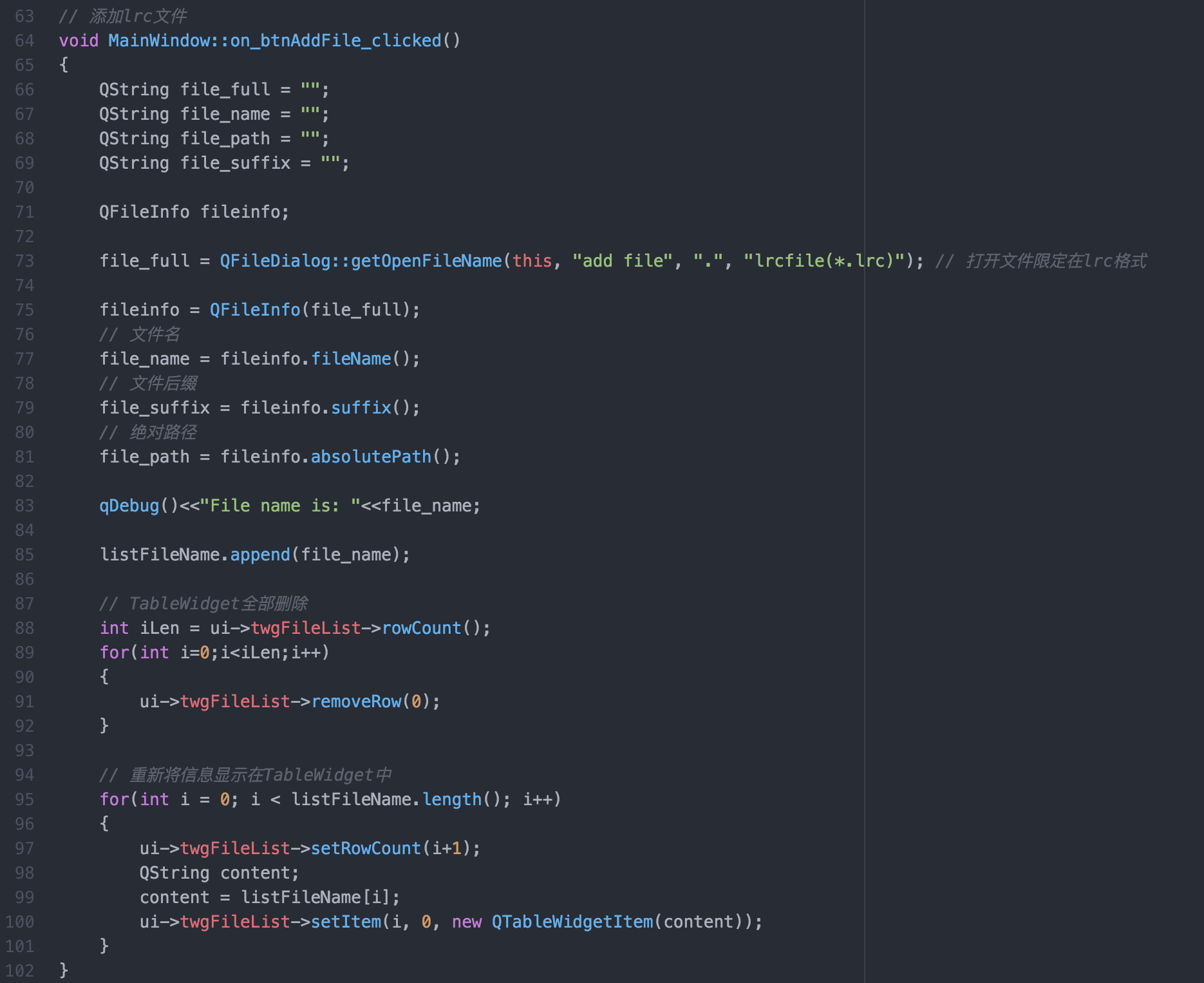
首先在initial()中定义QTableWidget的属性：



在主窗体头文件中定义歌词文件名列表存储形式：

*QVector<QString> listFileName*

在ADD按钮对应的信号槽出发函数中进行文件打开与选择操作，并将文件名append进容器中：



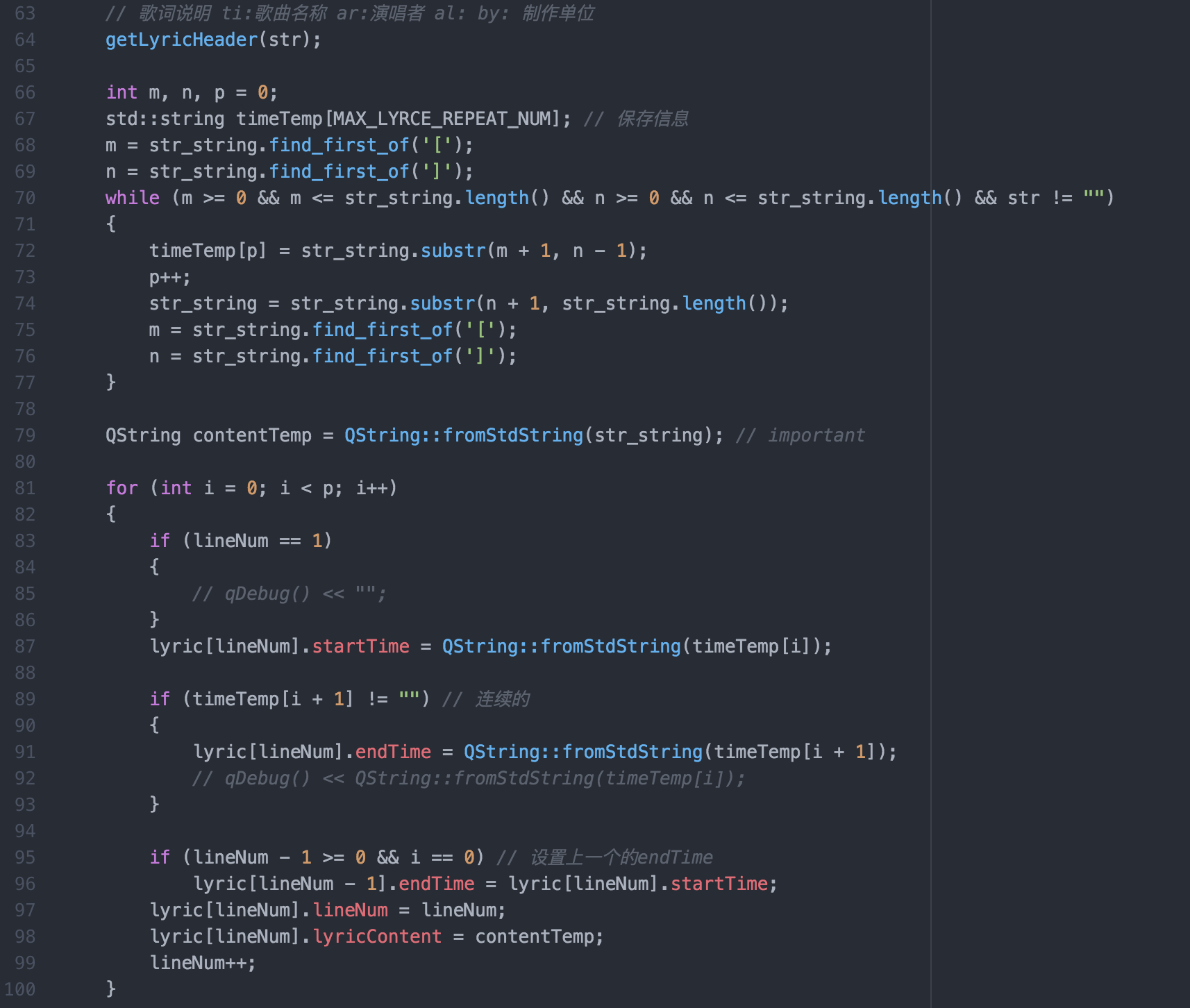
由此，Remove和Clear按钮操作实现即转换为对该QVector容器的操作。

需要注意的是，每次操作的最后都需要进行在TableWidget中的重新显示。

5.2 .lrc文件中标识和时间标签的过滤

这项功能的关键点在于对字符串的识别，而恰恰在此我被僵持住了很久，原因在于，QT中的字符串类型QString并没有标准C++ 11中string的诸如find\_first\_of()，最终我选择在QT环境中引入std库进行形式为”std::String”，虽然产生了QString与std::string之间多次的相互转换，但使得问题得以解决。

过滤逻辑如下：



最后将过滤结果分别保存至与.lrc文件名一致的.txt文件中：



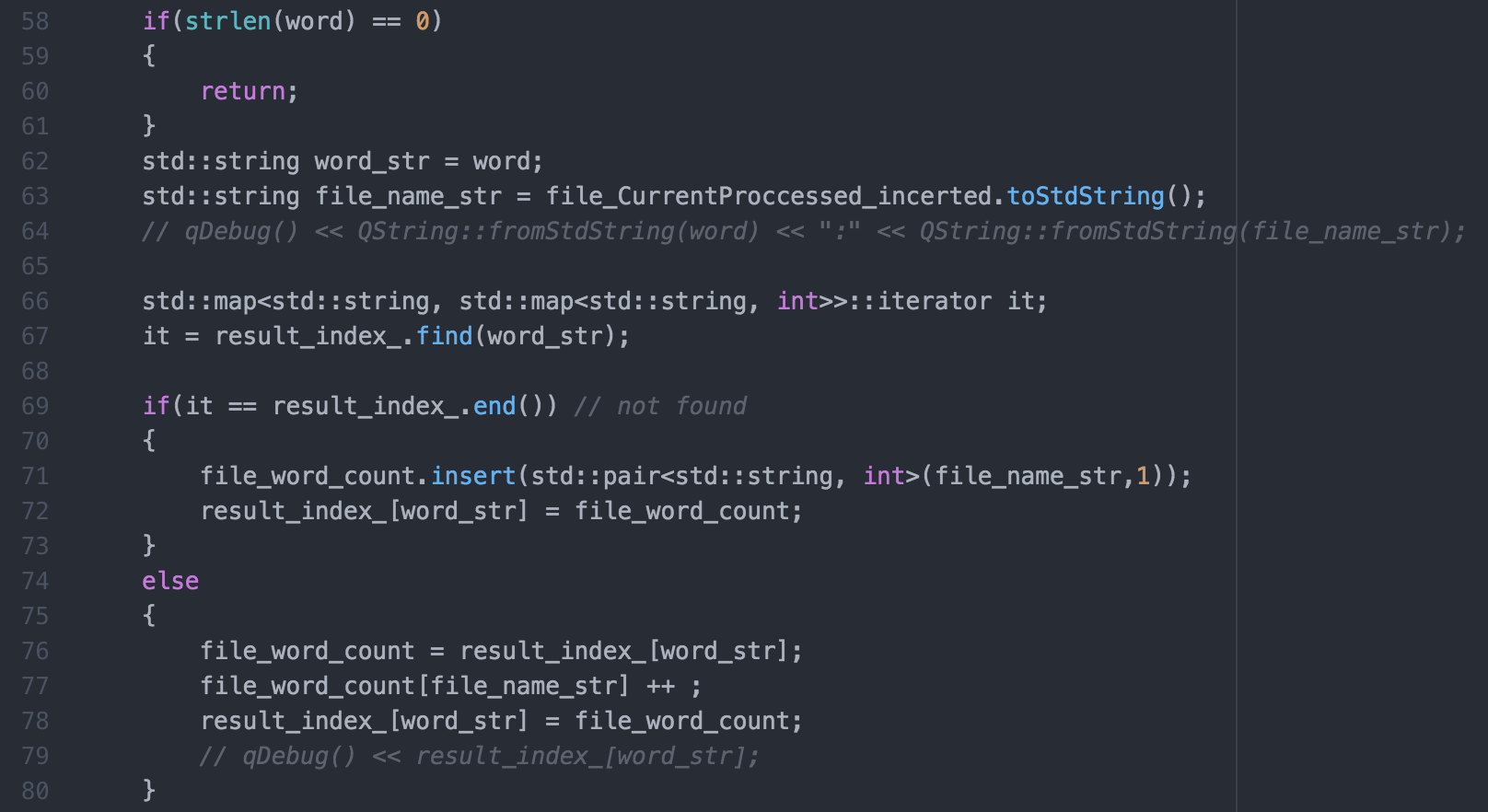
5.3 倒排索引的生成以及生成结果的存储结构

在此处的方法进行了借鉴，出处为：

<http://blog.csdn.net/okiwilldoit/article/details/51362839>

但我在其基础上进行了优化，主要着力点为：

* 调整为归一化的Map存储结构，方便了最后将结果通过Iterator遍历输出的时间复杂度。
* 重新调整了各个分步功能的实现位置，减少了全局变量的数量，使代码逻辑进一步清晰化。



最终，将得到的嵌套Map结构的倒排索引结构通过遍历器输出至result.txt文件中：



至此，该软件实现倒排索引的1.0版本全部完成，后期将会在界面lrc\_list的Group右侧添加布尔查询的用户操作端，从而实现从.lrc的解析、倒排索引的生成以及最后布尔查询的具体应用。

**六、写在最后**

感谢温延龙老师课上清晰的讲解，是对我倒排索引细节的掌握以及顺利制作完成该软件的基础。