实验二报告

邵钰杉 软件 51 2014011003

一．实验目标

1. 批量搜索功能

读取 query.txt 中的内容，其中每行为一个查询，关键词之间用空格分开。从 500 个网页中查询关键词所在的网页并按出现关键词的种类和数量排序。输出结果为 result.txt，每行为对应的查询结果，使用(docID,occurTimes)表示，空格隔开。

2.gui 交互界面

实现交互界面：输入关键词进行查询，返回在 500 个网页中的查询结果以及对应的文章信息。

二．实验环境

操作系统：Windows 10 教育版

IDE：Visual Studio 2015 Professional

编程语言：C++

三．抽象数据结构说明

1.平衡二叉树（MyAVL）

本次实验用模板实现了平衡二叉树，实现了以下接口：

insert（）：插入结点

search（）：查找结点

adjust（）：调整二叉树使其平衡

remove（）：移除某结点

2.文档链表（MyDoculist/MyHtmLInfo）

本次实验实现了两类文档链表，分别是 MyDoculist，MyHtmLInfo。前者是用 500 个网页生成的分词 AVL 树中结点的数据成员，用来记录每个不同的词语在文章中出现的次数以及文档编号，后者用来存储 html 的信息。

我为前者的接口实现了题目中要求的所有接口，即：

Add（）：添加文档

Search（）：搜索文档

Edit（）：修改文档

Remove（）：删除某文档

四．算法说明及流程概述

实验二的接口主要在 MyInterface2 文件中实现:

任务 1 的思路是这样的，先用哈希表去存储词典中的单词，然后对 500 个网页进行分词，每分出一个词就加到 AVL 树里，并进行记录文章编号以及排序等各种处理。最后得到一个 AVL 树。之后再 query.txt 里读取一行，对每一行的数据进行分词之后查询，输出综合排序后得到的结果。

任务 2 主要是做一个界面，我选用 QT 来做这个界面。和任务 1 的思路很相近，不过这次从 QLineEdit 中读取一行并进行查询，并且把查询的结果在 QTextEdit

中显示出来，并且把关键词高亮。

五．输入输出及操作相关说明

1.任务 1

输入：input 文件夹中的 500 个网页，dictionary 文件夹中的词典、query.txt 的查询 txt。

输出：result.txt，来输出查询结果。

2.任务 2

输入：input 文件夹中的 500 个网页，dictionary 文件夹中的词典

输出：一个 UI 界面，用来查询网页。界面中最上面的文本框用来输入要搜索的信息，点击”Search”按钮后，查询到的结果会显示在下方的文本框中并把搜索框内的信息高亮。下方有两个按钮，分别为 next 和 before，用来显示前一条网页和下一条网页。

（PS：尽管已经优化，但是载入这个 gui.exe 还是要 30s 时间，请助教耐心等待/(ㄒ o ㄒ)/~~）。

六．实验结果

完成所有要求的内容，并有额外完成的内容。额外完成的内容见功能亮点说明。

七．功能亮点说明

1.实验 1 中查询结果先按出现关键词从种类排序（出现关键词种类多的在前），若关键词出现的种类相同，则出现总次数越多越靠前。实现了按两个标准的排序。

2.平衡二叉树实现了题目要求必做的所有函数外，还实现了选做的 Remove 函数。

3.实现了两种词典索引机制，即平衡二叉树与哈希表。最终选用了哈希表作为词典索引。

4.使用了 QThread 多线程来加载数据。

八．实验体会

这次实验的任务 1 比较容易实现，因为实验一已经预留好了接口。对我而言花最多时间的就是任务 2，尽管小学期学了 Qt，但其实还是不是很熟悉。因为 VS2012 无法使用 Qt 插件，为了在 VS 里使用 QT 插件，专门下载了 VS2015。在虚拟机上跑 VS+QT 插件经常遇到各种奇怪的 bug（比如我在机子上尝试了各种方法也无法加载图片），不过还是磕磕绊绊地完成了这次实验。

这一学期在数据结构上花的时间真是不少，而这两次实验也加深了我对数据结构的理解，还是收获颇丰。