WebSearcher 实验报告

2012013311　张凯

一、　实验目的

本实验希望通过实现一个简单的网页检索系统，对常用数据结构进行训练。要求对给定的网页提取关键信息，并通过分词的技术对提取的关键信息进行整理。

二、　实验环境

本实验的程序 WebSearcher 使用 C++语言实现。程序代码编写（包括注释）的整个过程在 Ubuntu 12.04 LTS 操作系统完成，但可以将程序移植到任何安装有 C++编译器的操作系统。程序针对 Visual Studio 2012 专门进行了修改，可以直接导入到 Visual Studio 的项目中。

如果您要查看程序源码，请不要使用 Windows 的”记事本”（notepad），那将会使代码显示的格式很糟糕。

三、　实验用到的数据结构的抽象数据类型

栈：Stack（模板）
链表：LinkedList（模板）
字符串：CharString（类）
词库：WordDic（类）

四、　实验基本算法和流程

1、 分析一个 HTML 网页，用栈存储其逻辑结构，获得网页的字符编码等相关信息，提取网页的正文并存储在临时文件中。
2、 按照网页的字符编码加载特定的词典，对词典中的词按特定顺序排序。
3、 将网页的正文分割为一定长度的字符串，并在词典中查找其是否存在，存在则输出至结果文件。为了加快查找效率，使用了二分查找算法。

五、　程序输入输出及操作说明

在此只介绍如何使用 Visual Studio 进行编译及运行。更多详情请参照 readme.txt。

1、 启动 Visual Studio，新建一个空的控制台项目；
2、 将 src 目录里的程序源码、dictionary 目录里的词典文件拷贝至 Visual Studio 项目目录下；
3、 在 Visual Studio 中选择添加现有项，将程序源码添加至项目中；
4、 执行“生成解决方案”，编译成功后开始执行；
5、 在控制台窗口输入待解析的 HTML 文件名（包括完整路径），并按回车；
6、 程序会输出相关信息，如果解析成功，会将分词结果保存至项目目录下的“WebSearcher.txt”文件中。

六、　实验测试结果

本次实验对资料中前 10 个网页（GBK 字符编码）和”http://bt.ktxp.com/”网站

首页（UTF 字符编码）进行了分词，分词结果保存在 result 目录下。

七、 分数申请

1、 栈（10%）

程序实现了一个栈模板 Stack。具体请参照 Stack.h 文件。

2、 链表（10%）

程序实现了一个链表模板 LinkedList。具体请参照 LinkedList.h 文件。字符串实例化的链表模板可以成为字符串链表。

3、 字符串（15%）

程序实现了一个字符串类 CharString。具体请参照 CharString.h 和 CharString.cpp 文件。

4、 文本解析（20%）

WebSearcher 类具有解析 HTML 网页的功能，可以借助栈分析网页结构，并提取网页正文，。具体请参照 WebSearcher.cpp 文件中的 analyze()成员函数。

5、 分词算法（20%）

WebSearcher 类具有网页正文分词的功能，可以将网页的正文分割为一定长度的字符串，并在词典中查找其是否存在，存在则输出至结果文件。具体请参照 WebSearcher.cpp 中的 parse()成员函数。

6、 相关文档（10%）

撰写了相关文档详细说明了程序的编译、运行过程。具体请参照 readme.txt 和本实验报告。

7、 代码注释（5%）

程序源码加入了详细的注释。具体请参照程序源码。

8、 加分

（1） 处理转义字符（3%）

在 WebSearcher.cpp 文件中的 analyze()成员函数分析网页时考虑了转义字符。具体请参照相关代码及注释。

（2） 处理字符编码（3%）

在 WebSearcher.cpp 文件中的 analyze()成员函数分析网页时分析了网页的字符编码（UTF 或 GBK），在加载词库时会根据网页字符编码的不同加载不同的词库。具体请参照 WebSearcher.cpp 文件中的 analyze()成员函数和 loadDic()成员函数及相关注释。

（3） 词库的实现（3%）

实现了选做任务词库 WordDic，具体请参照 WordDic.h 和 WordDic.cpp 文件。

（4） 二分查找算法优化（6%）

为了提高查找效率，在从词典中查找一个单词是否存在是用了二分查找算法，将时间复杂度降低到 O(logn)，大大加快了查找时间。具体请参照 WordDic.cpp 中的 find()成员函数。

（5） 程序的系统兼容性和可移植性（5%）

程序开发的整个过程中充分考虑了程序对于不同操作系统、不同 IDE 的兼容性和可移植性，包括 readme.txt 详尽的说明了不同操作系统下程序的配置方法，程序注释也考虑到兼容性的问题而使用

了英文注释。具体请参照程序源码及 readme.txt。

分数总计：10%+10%+15%+20%+20%+10%+5%+3%+3%+3%+6%+5%=110%