**文件夹同步工具**

姓名：王少杰 学号：2013202589

**一、问题描述及分析**

有2个文件夹A和B，经过之前的同步操作，文件夹B中的内容与文件夹A中内容完全一致，现在用户对A文件夹中部分文件进行了编辑、移动、删除或者新增的操作，请你的程序根据文件夹A中的内容变化对文件夹B进行增量更新。

基本要求：

文件夹A中保持不变的文件在文件夹B中保持不变，不更新。

给出文件夹A中文件的更新情况列表，即列出与上次同步后文件夹A中所有发生变化的文件列表，包括

a) 与上次更新相比，所有新增文件的列表

b) 与上次更新相比，所有删除文件的列表

c) 与上次更新相比，所有发生变动的文件的列表

只对文件夹A中发生变化的文件在文件夹B中更新，即针对上述3中变化的文件，删除文件夹B中的相应原文件，并将A中最新文件复制到文件夹B中。

注意：删除时需要提示用户是否确定删除源文件。

提高要求：

1、将上述工具移植到windows平台下

2、在文件夹B中保留更新前的文件（比如，重新以特定的命名方式命名或者保存在特定的文件夹中的方式）

3、开发图形界面

问题分析：

实质上就是对文件夹进行遍历，并比较两者之间是否相同。对于文件夹下的子文件夹，则进行递归遍历。

**二、实现方案**

1、开始时输入路径，读取根目录下的日志文件以便同步时的比较，并从根目录本身开始比较。

2、先比较文件，并做上标记，剩下未标记的则为子文件夹。然后用递归的方法来比较子文件夹。

3、开始时有提示输入需要同步的文件夹的路径；在完成每一项工作时输出日志，做到界面友好。

4、提高要求中做了移植到Windows平台以及备份。由于Linux下有很多东西在Windows下不支持，所以移植的过程中我把所有不支持的东西都重新定义了一遍。文件的备份则是在操作之前先复制文件，并添加后缀 “\_bk”。

**三、算法描述**

以Linux下的代码为例：

同步的函数为int synchronize(string curA, string curB)，其中curA和curB指的是比较A文件夹下的curA文件和B文件夹下的curB文件。

然后是打开curA并存储相关信息，以便和curB比较：

DIR \*dir = opendir(pathA.c\_str());

while ((dirptr = readdir(dir)) != NULL){

string filename;

filename.assign(dirptr->d\_name);

if (filename == "." || filename == "..") continue;

if (dirptr->d\_type == 4){

dirs.insert(make\_pair(filename, 1));

}

else{

struct stat buf;

stat((pathA + "/" + filename).c\_str(), &buf);

files.insert(make\_pair(filename, buf.st\_mtime));

}

}

现在打开curB：

dir = opendir(pathB.c\_str());

while ((dirptr = readdir(dir)) != NULL){

string filename;

filename.assign(dirptr->d\_name);

if (filename == "." || filename == "..") continue;

if (dirptr->d\_type == 4){

if (dirs.find(filename) == dirs.end()){

//如果找到了结尾，说明B中这个文件不存在于A，故删除之

system(("cp -p -r " + pathB + "/" + filename + " " + pathB + "/" + filename + "\_bk").c\_str());//备份

system(("rm -r -i " + pathB + "/" + filename).c\_str());

cout << ("Folder deleted: " + pathB + "/" + filename) << endl;

}

else dirs.find(filename)->second = 0;//标记为已同步

}

else{

if (files.find(filename) == files.end()){

system(("cp -p " + pathB + "/" + filename + " " + pathB + "/" + filename + "\_bk").c\_str());//备份

system(("rm -r -i " + pathB + "/" + filename).c\_str());

cout << ("File deleted: " + pathB + "/" + filename) << endl;

}

else{

struct stat buf;

stat((pathB + "/" + filename).c\_str(), &buf);

if (files.find(filename)->second != buf.st\_mtime){

//如果修改时间不相同，则更新B文件夹中的相关内容

system(("cp -p " + pathB + "/" + filename + " " + pathB + "/" + filename + "\_bk").c\_str());//备份

system(("cp -p " + pathA + "/" + filename + " " + pathB + "/" + filename).c\_str());

cout << ("File synchronized: " + pathB + "/" + filename) << endl;

cout << ("File backup: " + pathB + "/" + filename + "\_bk") << endl;

}

files.find(filename)->second = 0; //标记为已同步

}

}

}

然后对子文件夹进行递归的处理：

for (it = dirs.begin(); it != dirs.end(); it++){

if (it->second){

system(("mkdir "+ pathB + "/" + it->first).c\_str());

cout << ("Folder created: " + pathB + "/" + it->first);

}

synchronize(curA + "/" + it->first, curB + "/" + it->first);

}

for (it = files.begin(); it != files.end(); it++){

if (it->second){ \_

system(("cp -p " + pathA + "/" + it->first + " " + pathB + "/" + it->first).c\_str());

cout << ("File created: " + pathB + "/" + it->first) << endl;

}

}

return 0;

}

**四、遇到的问题以及解决**

1、写路径文件的时候一开始用的是最简单的c语言，然后发现比较麻烦，后来用了串这个数据结构，即string，简化了文件路径的问题。

2、文件之间的对比问题。一开始想逐字对比，后来想到可以用对比修改时间的方法来检测是否被修改过，这样一来大大提高了运算效率。

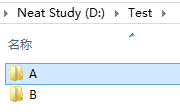
3、文件和文件夹之间的处理不太相同，所以想把它们分开处理。后来把已经同步过的文件find(filename)->second全赋值为0，这样一来剩下的就都是文件夹了，然后用递归函数进行操作即可。

4、Linux下写的代码复制到Windows平台下之后，里面的汉字提示全都变成了乱码，无奈只好把说明改成英文的。

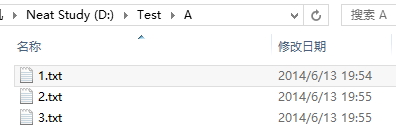
**五、使用说明**

方便起见，使用Windows下的截图。

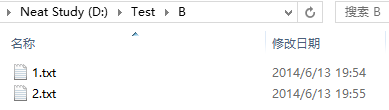
1、在D:\Test下同步A和B两个文件夹的内容：



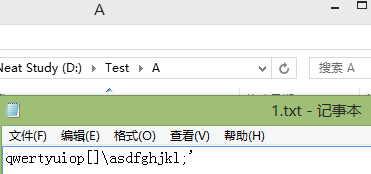
2、A文件夹下初始内容如下：

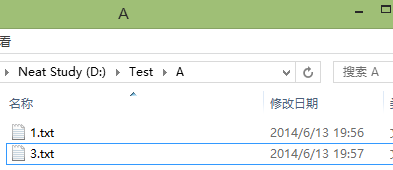


3、B文件夹下经同步后和A中相同：

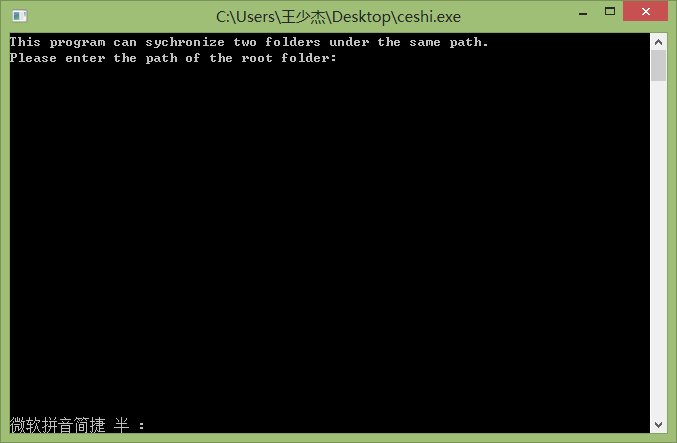


4、现在A中改动1.txt的内容，删除2.txt，新建3.txt：

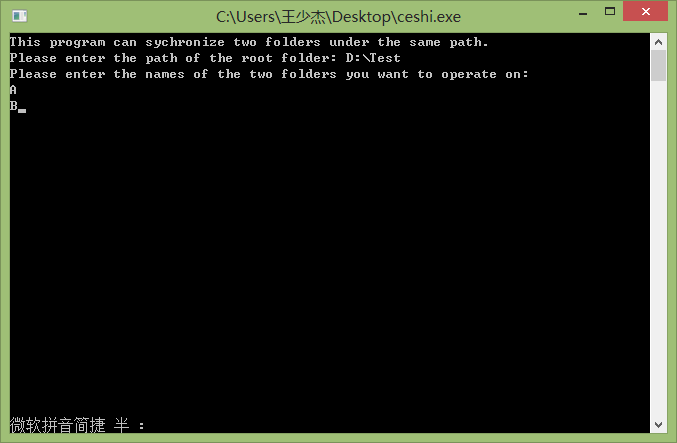




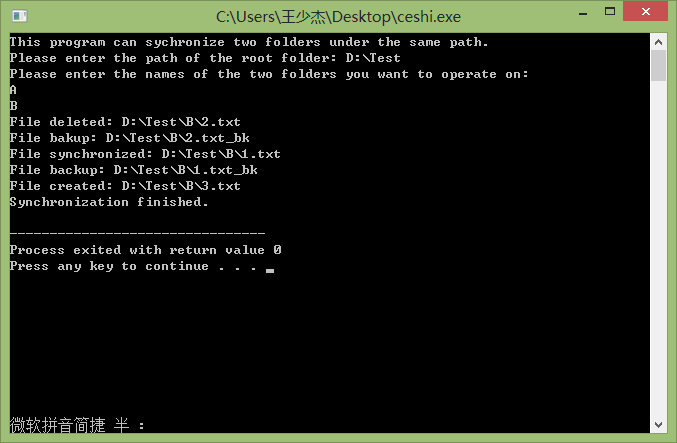
5、运行同步工具：



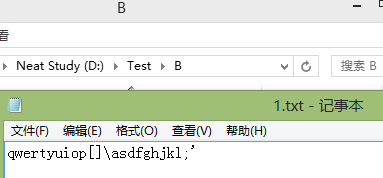
6、按照界面提示输入根目录路径以及需要同步的两个文件夹名称：

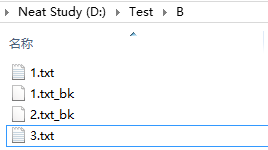


7、同步完成：



8、检查B文件夹中结果：





其中“.txt\_bk”文件为备份文件。

**六、附录**

附录一（代码）：

Linux-原始.cpp、Linux.cpp、Windows.cpp

原始版本只能在限定的目录下进行同步，如需更改目录则要更改代码，但运行更良好。

代码另附

附录二（对课程的一点看法）：

上课时候下面没人回应老师的话我都觉得很尴尬。我不知道老师了不了解情况，因为我们这边大多数都是打算学数学的，所以大部分人的观点就是把数学学好，其他的就放一放。再加上学分少了，所以这学期上课都不是很积极。不过这是教务方面的问题，您上课讲的还是蛮好的。这是我的一点看法，不知道能不能有用。