程设大作业说明文档

模块一：刘登科（组员） 2021010851 计 17

模块二：马睿杰（组员） 2021010766 计 14

模块三：黄家骏（组长） 2020010436 能动 03

模块四：韩佑硕（组员） 2021080070 软 12

1 模块一（Diff 指令）

1.1 功能介绍

该 diff 函数主要使用最长公共子序列的方法。首先将两个文件读取后分别按行存入<vector>string str1 和<vector>string str2 中，再定义出 nstr1、2(用 str1、2 进行初始化)，然后动态规划找出他们的最长公共子序列存入<vector>string share 中，之后将 nstr1、2 与 share 进行比较并将结果存入标准输出。

1.2 指令功能

(除 help、B 和 q 外只需修改求最长公共子序列的 nstr1、2 即可)

--help：显示帮助，即 diff 指令的选项功能

-b：将空格字符替换为一个

-B：读取文件时忽略空白行

-i：将所有字符替换为小写

-q：比较 nstr1、2 与 share 是否完全一致

-w：删除所有的空格字符

-I[字符串]：将 nstr1、2 中含有指定字符串的字符串用指定字符串替换

1.3 合作过程中的问题及修改

（1）帮助提示输出乱码：将 txt 的编码调成 ANSI 格式

（2）全局变量重定义：统一在头文件中声明全局变量再引用

（3）换行问题：在合适位置加\n

2 模块二（Grep 指令）

2.1 功能介绍

本函数功能主要是将文件内的每行字符串与模式串进行比对，并将匹配字符串写入标准输出中。在无通配符的情况下，该函数主要通过遍历整个字符串的循环来寻找匹配串。在具有通配符的情况下，该函数主要通过使用了动态规划的递归算法找到匹配字符串的首尾位置。得到位置信息后，通过字符串拼接函数得到标准输出。

2.2 指令功能

--help：显示帮助

-c：计算符合样式的行数。

-h：在显示符合样式的那一行之前，不标示该行所属的文件名称。

-H：在显示符合样式的那一行之前，标示该行所属的文件名称。

-i：忽略字符大小写的差别。

-n：在显示符合样式的那一行之前，标示出该行的编号。

-A[行数]：除了显示符合范本样式的那一列之外，同时显示该行之后的[行数]行内容。多行连续匹配需要合并输出，不要冗余输出。

-B[行数]：除了显示符合样式的那一行之外，同时显示该行之前的[行数]内容。多行连续匹配需要合并输出，不要冗余输出。

备注：

1.输入时用“—”代替标准输入时须在指令输入后，否则会提示不存在该条指令。

2.Strin 既可代替模式串输入，也可代替文件，但作为模式串时应仅有一行。

2.3 合作过程中的问题及修改

（1）在改变字体颜色时会同时改变其后无关的文字：将无关文字设为默认色

（2）路径名不对应导致打开文件时程序崩溃：改变文件路径名的输入方法

（3）未设置全局变量导致指令未重置。

3 模块三（Tee、Cat、Cp、Cd、Pwd 指令）

3.1 Tee 指令

3.1.1 功能介绍

tee 指令会从标准输入读取数据，将其内容输出到标准输出，同时可以保存至多个文件。

3.1.2 指令功能

--help： 显示帮助。

-a： 附加到既有文件的后面，而非覆盖它。

3.1.3 合作过程中的问题及修改

单人作业时使用 VS2019 提示 strcat 等部分函数需要在后面加上“_s”才能使用，在与小组成员合作的过程中使用 devC++，无法正常使用，因此去掉“_s”。为达到不覆盖的效果，在用 ofstream 时增加 ios::app，以直接实现覆盖功能。

3.2 Cat 指令

3.2.1 功能介绍

cat 指令用于连接文件并输出到标准输出上。如果文件不存在则会提示读取失败。

3.2.2 指令功能

--help： 显示帮助。

-n：由 1 开始对所有输出的行数编号。

-b：和-n 相似，只不过对于空白行不编号，但需要输出空白行。

-s：当遇到有连续两行以上的空白行，就代换为一行的空白行。

-E： 在每行结束处显示$。

3.2.3 合作过程中的问题及修改

输出的最后一行缺少换行，补上即可。

3.3 Cp 指令

3.3.1 功能介绍

cp 指令用于复制文件。如果文件数超过 2 则不执行操作；如果第一个文件不存在则不进行操作；如果第二个文件不存在则生成新文件并复制。

3.3.2 指令功能

--help：显示帮助。

-n：不覆盖已存在的文件。

3.4 Cd 指令

3.4.1 功能介绍

cd 指令用于切换工作目录。

3.4.2 指令功能

[路径]如以“/”开头为绝对路径，否则为相对路径。 另外，“.”表示同一层目录，“..” 表示上一层目录。 注意不能切换到根目录的上层目录，这样会发出警报。

3.4.3 合作过程中的问题及修改

“/..”会出现工作目录为空的情况，完善代码逻辑判断即可。

3.5 Pwd 指令

3.5.1 功能介绍

pwd 指令用于显示工作目录。

3.5.2 指令功能

--help：显示帮助。

直接输入 pwd 即可

4 模块四

4.1 解释器框架

解释器框架先使用基本的标准输入函数（cin,cin.getline)完成设置计算机名，根目录，用户名的任务，再使用 while 语句输出提示符并接受输入。 主要使用 strtok 函数把每条指令拆成若干个 token 保存到 argc,再把 argc 传到自定义的，得出最终标准输出的函数（get_strout)，并输出到屏幕上。

备注：主要的难点在于释放所申请的动态分配。为了尽量模仿真实情况，cd 指令不是最后一个指令的话不执行

4.2 echo 指令

4.2.1 功能介绍

该函数通过把 gTerm.argv 里的已被拆好的 token 储存于标准输出（gTerm.strout）来实现。

4.2.2 指令功能

选项

--help ： 显示帮助

-n：表示输出之后不换行，通过最后 gTerm.strout 不加"\n"来实现。

备注：输入“echo -n --help” 时，不会显示帮助不换行，而会输出--help 不换行（按照真实情况实现）

4.3 复合指令

复合指令在自定义的 get_strout 函数里实现了扩展语法，当户输入“[指令 1] |” 时，（也可以出现多条指令，重要的是最后以“|”结束），解释器输出“pipe>”等待用户再输入下一个指令，用户输入后，处理那条指令

备注：多条复合指令多次出现的话， 输出 “parse error near ‘|’ ”（这里没考虑 double vertical bar ||，也当复合指令重复出现处理）

4.4 扩展功能（exit 功能）

当用户输入"exit"时，释放所有已申请的动态分配，并结束程序。

4.5 问题与修改

4.5.1 标准输入问题

执行每行命令的第一条指令之前要清空 gTerm.strin（为了合理执行读取标准输入的指令）。

4.5.2 复合指令扩展语法问题

实现复合指令时改变 argv 的地址，在本人设备不会出错，而在其他组员设备上出现越界问题。

改进方案 ：原来会访问没被 token 占用的地址，改善了这部分，只会访问被 token 占用的地址。随之，也稍微改变了释放内存的方法。