ls、wc 命令实现及源代码对比文档

161250030

弓宇德

1. 程序功能概述

ls -l (-d, -R, -a, -i)

显示当前目录下的文件及其相关信息（默认指令为-l），支持-d：仅显示当前目录，-i：显示 inode，-a：显示隐藏文件和隐藏文件夹，-R：递归显示当前目录下所有子目录文件夹

wc [filename]

统计指定目录下若干文件的字符数、行数、byte 数，如果文件多于一个，会统计总量。

2. 程序设计介绍

（1) ls

1. argv 用于接受参数（-d, -R, -a, -i），然后保存在 char 型数组 parmas（char[4])中。

2. 调用 file_operation()方法进行处理

3. 在 file_operation 方法中，根据传入参数的不同分别进行处理。使用了 stat 结构体和 dirent 结构体来获取文件的相关信息，这些方法主要写在 file_guid.c 和 file_mode.c 中。

4. 用 char dirs[100][100]保存子目录，用于-R 命令的递归调用。

（2) wc

1. 用 fopen、fgetc 方法获取文件内容，\n 作为行分隔符，EOF 代表文件结尾。

2. 用空白符（\t,\v,\n,\r,\f,空格）作为词分隔符

3. 使用全局变量来保存总用量

3. 与源码进行比较

（1)ls

源码中 ls 的实现更为复杂，比如考虑了对应不同文件，输出的颜色不同，对于链接，会显示链接的源文件等多功能。

此外，在源码中，ls 指令通过 peding_dirs 结构实现了输出的正序，而我的实现当中，所有文件的输出顺序是杂乱的。源码使用 active_dir_set 结构避免软链接死循环，而我则是采用递归时忽略”.”和”..”来实现的。

（2)wc

源码中 wc 的实现是使用一次遍历，多个变量用于保存；而我实现的 wc 中，词和行的统计是分开的。此外，源码中对于字符的定义，是采用宏或 typedef 来定义的，这样使得 wc 命令可以在不同的平台、不同的字符编码集下能有更好的兼容性表现，而我实现的 wc 命令中没有考虑这些因素，因此兼容性较差。