

代码文档

程序功能文档

ls命令功能

```
ls [-ldRai] [path]
```

支持参数-l、-d、-R、-a、-i，顺序任意，可以连写
最后可以输入一个文件路径，如果没有则默认为当前路径

-l: 显示文件详细信息
-d: 仅显示选择的路径（如不指定就是当前路径）所指向的文件或目录
-R: 递归显示子目录 -a: 显示隐藏文件 -i: 显示inode号

wc命令功能

```
wc [path]
```

统计路径指向的文件的单词数、行数、字节数

设计文档

ls命令设计

1. 首先根据main函数的argc和argv处理参数信息，用若干个bool标记是否有对应的参数，最后一个参数可以是路径，默认为"."
2. 然后决定输出的文件范围，根据-a和-d参数，对要输出的文件进行过滤
3. 接下来输出每一个文件，根据-l和-i参数，决定输出格式与要输出的信息
4. 最后根据-R参数，决定是否要递归子路径，重复上述操作

wc命令设计

1. 首先定义两个函数分别用于统计单词数与行数，单词统计需要数空格、制表符、换行符等，行数统计换行符数量即可
2. 然后直接输出

源码比较

ls命令比较

源码具有根据文件类型决定输出颜色的功能

源码会将目录下的文件存入缓冲区并对他们根据字典序排序，而我的代码则是直接遍历输出

同时对于文件的管理更加细致，具有fileinfo结构体，形成文件树的树形结构，在使用的时候更加方便

在参数较多的情况下也能够灵活处理，而我的代码频繁地使用了系统api，并且只是在递归中体现了文件树

wc命令比较

源码一次性把行数单词数统计了，而我用了两个函数

源码用了大量的宏用来应对各种情况，而我的没有

我使用了set来方便的寻找特殊字符，而源码没有，因为它不是C++