

## CS302 lab11 补充说明

Q:输出 `path` 的时候最后要不要加/?

A: 要。所有路径是`/xx/xx/xx/`的格式，方便在处理 `path` 相关内容的时候，根据`'/'`来进行划分，获取目录信息。

Q:用户之间共享文件系统吗?

A:共享。创建的任何文件和目录对所有人都可见。所有文件的权限对任何用户来说都相同。

Q:关于文件相关的操作，能否更细致的说明?

A:任何文件相关操作都要进行文件存在性检查。如果创建时已经存在该文件，则返回错误。如果对文件进行操作时，文件并不存在，则返回错误。`write` 和 `cat` 操作必须在文件打开(`open`)之后才可以进行。`truncate` 和 `reposition` 命令则不需要。`write` 和 `truncate` 命令执行之后，文件指针会回到文件末尾。`write` 和 `cat` 分别对应读和写，当打开文件的权限不对应时，则返回权限错误。这里 `truncate` 并不需要写权限。`write` 函数写入的 `buff` 是一个字符串，该字符串可以被 `cin/scanf` 一次读入，也就是不含空格、换行等分隔符。任何人都可以对文件的权限进行修改。同一个文件不能被打开多次，如果发生此状况，返回文件已打开错误。

Q:man 命令里为什么没有 cd、cdp?

A:我疏忽了，最新的 man 截图如下：

```
Manual:
0) pwd (current work directory)
1) mkdir(p) (path) directory_name
2) dedir(p) (path) directory_name
3) search file_name
4) create(p) (path) file_name omode
5) cat(p) (path) file_name
6) delete(p) (path) file_name
7) write(p) (path) file_name buff wmode
8) reposition(p) (path) file_name pos
9) truncate(p) (path) file_name cnt
10) ls(p) (path)
11) open(p) (path) file_name
12) close(p) (path) file_name
13) regusr user_name password
14) delusr user_name [root only]
15) disusr
16) login user_name
17) lsfile
18) chmod(p) (path) file_name mode
19) cd(p) (path) dir_name
20) exit
root$/%:
```

Q: 关于用户相关的命令，能否做更细致的阐释？

A: 只有 root 用户可以显示用户列表、删除用户。当非 root 用户进行显示、删除操作时，返回权限错误。所有用户都可以注册新的用户。注意当删除 root 用户时，应返回权限错误。成功删除一个用户时，不需要输出任何提示信息。用户切换时，当前工作目录不改变；目录切换时，当前用户不改变。

Q: ls 的格式有要求吗？

A:有。按照创建时间顺序先输出子目录名称，输出完子目录之后在输出文件名称，每个一行。如果是空目录，则什么也不输出。

Q:这样用户相关的实现有什么用？

A:实现一个基本的框架，以便于能力强的同学做 **bonus**。

Q:**exit** 命令运行时，是退出程序还是其他的操作？

A: 退出程序。

Q:**search** 会从当前目录开始找还是从根目录开始找？

A:从根开始，找出所有文件。

Q: **search** 的时候采取什么方式？

A: 不管黑猫白猫，抓到耗子就是好猫。

如果有其他问题，请及时在 **bb** 里提出与大家讨论。

关于错误提示方面优先级的总结：

存在性 > 合法性

对测试数据的具体描述：

不会出现没有出现在文档里的情况。