

## 总结

1. link/unlink: 本质上是操作根目录下的DirEntry。如果进行link, 则查找原文件的id, 然后用新名字和id新建一个DirEntry加到根目录的disk\_inode下; unlink的时候就是找到对应名字的DirEntry给删掉即可, 当然如果发现这是最后一个(只有一个对应id的项了), 则还需要把文件的内容删掉。
2. get\_state: 需要获取的信息通过遍历即可拿到, 操作上比较容易; 而对于指针地址转换, 方法参照task\_info的写法即可。

附: 关于link/unlink的实现, 似乎也可以在fs下加一个map来实现。不过考虑到ls的时候理应将所有的链接都打出来, 这样的实现似乎不是很自然, 因此放弃了这种实现。

## 问答题

1. 在我们的easy-fs中, root inode起着什么作用? 如果root inode中的内容损坏了, 会发生什么?

在本次实验中, root inode是根目录的inode, 用来查找根目录的内容。由于我们只有一层目录, 所以相当于直接通过root inode查找所有文件。

如果root inode损坏, 则可能出现找不到其他文件inode的问题。