



哈尔滨工业大学 (深圳)  
HARBIN INSTITUTE OF TECHNOLOGY

# 实验设计报告

开课学期: 2022 年秋季  
课程名称: 操作系统  
实验名称: File System  
实验性质: 额外实验  
实验时间: 10.30 地点: T2  
学生班级: 11  
学生学号: 200111132  
学生姓名: 吴桐  
评阅教师: \_\_\_\_\_  
报告成绩: \_\_\_\_\_

实验与创新实践教育中心印制

2022 年 9 月

## 一、 实验详细设计

*注意不要照搬实验指导书上的内容，请根据你的设计方案来填写*

### 实验一：大文件（Large files）

这部分其实和我们上课学的文件系统比较类似。

原来的 xv6 里面每一个文件的索引块地址数组中有 12 个为直接索引 12+1\*256 blocks，现在相当于我们就把其中的一个直接索引换成一个二级索引块，就可以增大文件的块数了。

先把 fs.h 里面的 NDIRECT 宏-1，然后修改一下对应的地址数组。

后面如何利用这扩大的块，我们就需要修改对应操作的函数。这里主要修改的是 bmap 和 itrunc。

主要是仿造他对一级数据块的一个使用，我就相当于对数组中最后一个地址，模仿着加了一层（找空位，内存分配新空间），就可以实现一个二级的访问了。

### 实验二：软链接

通俗易懂的来讲，就是快捷方式。这里其实是实现的一个系统调用（添加相关部分我就不在重述了，主要谈谈其中我实现的操作逻辑）

这里如何保存地址，我选择在 file 结构体中加入一个 target 字段，用于存储快捷方式连接的文件真实路径

系统调用创建链接 sys\_symlink 的时候，函数相当于就把目标路径付给作为快捷方式文件的 target 字段，便于后续访问。

然后就是修改访问的调用：根据文件类型是否是软链接（快捷），然后再去 inode 找他的 target 字段。这个时候需要进行连续判断，因为软链接到的文件还有可能仍然是快捷方式，所以我们需要继续判断并且查找。但是这里根据提示，我们查找次数要设置上限，因为可能会形成环路卡死。

## 二、 实验结果截图

```
make[1]: Leaving directory '/root/.myxv6/github/MIT6.S081-2020-labs'
== Test running bigfile ==
$ make qemu-gdb
running bigfile: OK (154.1s)
== Test running symlinktest ==
$ make qemu-gdb
(0.9s)
== Test  symlinktest: symlinks ==
symlinktest: symlinks: OK
== Test  symlinktest: concurrent symlinks ==
symlinktest: concurrent symlinks: OK
== Test usertests ==
$ make qemu-gdb
usertests: OK (246.4s)
== Test time ==
time: OK
Score: 100/100
root@M-8-8-ubuntu:~/myxv6/github/MIT6.S081-2020-labs#
```