实验八

File system

目的：解决文件系统大文件存储以及软连接问题。

本实验要求实现用户态线程切换，并且实现“时钟”特性，使得可以对程序产生时钟中断。

**具体操作：**

大文件：本实验将增大文件的最大值。当前xv6的最大文件为268个块。这是因为现在xv6采用的是12个直接映射块加上1个单级间接映射块（容纳256个块）。本实验目的是将一个直接映射块改成两级间接映射块。

软链接：使用路径进行文件（目录不要求）链接。

步骤：

1. 准备工作：改变[mkfs]文件系统最大文件容量FSSIZE2000至200000（param.h），并重新分配空间。之后改变file.c相关参数。
2. 修改bmap 和 itrunc，扩展两级间接映射。小心，inode和dinode结构体的addrs都要改，他们在不同的代码文件中。运行之前记得make clean。
3. 软连接：增加symlink(char \*target, char \*path)系统调用，它将path软连接到target上。这和alarm实验一样。Symlink的功能是：创建在path下创建一个新的文件，内容是target。如果path已经对应了文件，那么软连接退出，错误。
4. 软连接相关定义和参数.
5. 实现symlink：
	1. 找到path所在位置，模仿link，找到父目录和最终位置。
	2. 创建新文件，类型为软连接类型。
	3. 将文件内容设置为target
6. 修改open，以解决软链接文件的查找问题。

注意target不一定要存在，如果不存在的话需要新建一个数据块来存储相应的文件（inode的数据块）。之后修改open的系统调用，以解决打开软连接时的情形，注意，当收到NOFOLLOW的时候，打开的不能为软连接文件，标志失败。

软连接还可以嵌套，这意味着open还要一直深入直到找到真正的文件。但软循环可能会造成文件环，不停的嵌套（设定嵌套最大值10解决），这要求返回错误。

我们从sysfile.c文件进行着手，先看看link和unlink是怎么运作的。他们实现的是硬链接操作。

创建新的inode的时候，需要设置对应的maor和minor属性，那么我们来了解一下这两个属性的意思。当然，我们直接使用

https://blog.csdn.net/cymm\_liu/article/details/11479537







