**操作系统实习**

Lab5 文件系统 实习说明

本实习希望通过修改Nachos系统的底层源代码，达到“完善文件系统”的目标。

【背景描述】

Nachos文件系统建立在模拟磁盘上，提供了基本的文件操作，如创建、删除、读取、写入等等。文件的逻辑结构与物理位置之间的映射关系由文件系统统一维护，用户只需通过文件名即可对文件进行操作。

然而，相比于实际文件系统，Nachos文件系统还存在很多不足之处：

* 文件长度的限制

Nachos文件系统采用直接索引方式，故文件长度不能超过4KB（更准确的说，是((128 – 2 \* 4) / 4) \* 128 = 3840 B）。同时，文件的长度必须在创建时予以指定，且不得更改。

* 文件数目的限制

Nachos文件系统只有一级目录，系统中所有的文件都存于根目录下，且数目不能多于10个。

* 粗粒度的同步互斥机制

Nachos文件系统每次只允许一个线程进行访问，不支持多个线程同时访问文件系统。

* 性能优化与容错

Nachos文件系统没有Cache机制，也没有容错机制，即当文件系统正在使用时，如果系统突然中断，文件内容的正确性无法保证。

【实习内容】

**一、文件系统的基本操作**

Exercise 1 源代码阅读

阅读Nachos源代码中与文件系统相关的代码，理解Nachos文件系统的工作原理。

code/filesys/filesys.h和code/filesys/filesys.cc

code/filesys/filehdr.h和code/filesys/filehdr.cc

code/filesys/directory.h和code/filesys/directory.cc

code /filesys/openfile.h和code /filesys/openfile.cc

code/userprog/bitmap.h和code/userprog/bitmap.cc

Exercise 2 扩展文件属性

增加文件描述信息，如“类型”、“创建时间”、“上次访问时间”、“上次修改时间”、“路径”等等。尝试突破文件名长度的限制。

Exercise 3 扩展文件长度

改直接索引为间接索引，以突破文件长度不能超过4KB的限制。

Exercise 4 实现多级目录

Exercise 5 动态调整文件长度

对文件的创建操作和写入操作进行适当修改，以使其符合实习要求。

**二、文件访问的同步与互斥**

Exercise 6 源代码阅读

1. 阅读Nachos源代码中与异步磁盘相关的代码，理解Nachos系统中异步访问模拟磁盘的工作原理。

filesys/synchdisk.h和filesys/synchdisk.cc

1. 利用异步访问模拟磁盘的工作原理，在Class Console的基础上，实现Class SynchConsole。

Exercise 7 实现文件系统的同步互斥访问机制，达到如下效果：

1. 一个文件可以同时被多个线程访问。且每个线程独自打开文件，独自拥有一个当前文件访问位置，彼此间不会互相干扰。
2. 所有对文件系统的操作必须是原子操作和序列化的。例如，当一个线程正在修改一个文件，而另一个线程正在读取该文件的内容时，读线程要么读出修改过的文件，要么读出原来的文件，不存在不可预计的中间状态。
3. 当某一线程欲删除一个文件，而另外一些线程正在访问该文件时，需保证所有线程关闭了这个文件，该文件才被删除。也就是说，只要还有一个线程打开了这个文件，该文件就不能真正地被删除。

**三、Challenges题目（至少选做1个）**

Challenge 1 性能优化

1. 例如，为了优化寻道时间和旋转延迟时间，可以将同一文件的数据块放置在磁盘同一磁道上
2. 使用cache机制减少磁盘访问次数，例如延迟写和预读取。

Challenge 2 实现pipe机制

重定向openfile的输入输出方式，使得前一进程从控制台读入数据并输出至管道，后一进程从管道读入数据并输出至控制台。

【实习建议】

1. 数据结构的修改和维护

文件管理的升级基于对原有Nachos数据结构的修改。增加文件的描述信息需对文件头结构进行简单修改。多级目录中可创建目录也可创建文件，应根据实际的文件类型初始化文件头信息。

1. 实现多级目录应当注意
   * 目录文件的含义。每个目录对应一个文件，通过此文件可了解其子目录及父目录的信息。
   * Nachos的目录文件大小是预先定义的，但实际上，目录文件的大小应根据内容确定，且能改变。
   * 实现多级目录后，添加、删除目录项要根据具体的路径，对树的遍历要有深刻的理解。
2. 为了实现文件长度无限，可以采取混合索引的分配方式。