# **<实验四> Linux文件I/O操作**

班级学号： 姓名：

≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡≡

|  |  |
| --- | --- |
| 实验目的和内容 | 实验目的：  通过编写文件读写及上锁的程序，进一步熟悉Linux中文件I/O和进程相关的应用编程，并且熟练掌握open()、read()、write()、fcntl()等函数的使用。  实验内容：  本实验需要打开两个虚拟终端，分别运行生产者程序（producer.c）和消费者程序（customer.c）。此时两个进程可能同时对同一个共享缓冲池（用普通文件FIFO.txt模拟 ） 进行读写操作。所以可以使用文件锁机制保证两个进程对该临界缓冲区的访问都是原子操作。  先启动生产者进程，它负责创建模拟FIFO结构文件并投入生产，就是按照给定的时间间隔，向FIFO文件写入自动生成的字符（在程序中用宏定义选择使用数字还是使用英文字符），生产周期以及要生产的资源数通过参数传递给进程（默认生产周期1秒，要生产的资源数为10个字符）。然后后启动的消费者进程按照给定的数目进行消费，首先从文件中读取相应数目的字符并在屏幕显示，然后从文件中删除刚才消费过的数据。为了模拟FIFO结构，此时该需要访问两次。 |
|  | 微机、WINDOWS XP、 LINUX操作系统、VMWARE虚拟机软件 |
| 实验重点及难点 | 实验重点：  open()函数用于打开或创建文件，在打开或创建文件时可以指定文件的属性及用户的权限等各种参数。  read()函数用于将从指定的文件描述符中读出的数据放到缓存区中，并返回实际读入的字节数。若返回0，则表示没有数据可读。  write()函数用于向打开的文件读写数据，一般读写操作从文件当前指针位置开始。  难点: fcntl()函数用于向打开的文件属性实施访问或设置（如锁机制） |
| 实验过程（实验过程、实验结果 ） | |
| 实验过程可以看出运行生产者的程序时可以用命令参数控制一次生产多少个东西。而消费者在可以消费的情况下一次固定消费10个。在对应看看 FIFO文件中的容遵循fifo的规律，所以对于先进先出模拟成功。 | |
| 实验总结 | |
| 通过实验运行结果可以看出运行生产者的程序时可以用命令参数控制一次生产多少个东西。而消费者在可以消费的情况下一次固定消费10个。在对应看看 FIFO文件中的容遵循fifo的规律，所以对于先进先出模拟成功。 | |