

# Lab04 - Assignment

姓名:

学号:

## 1. 文件 I/O 函数

(1) 创建一个文件，内容为你的姓名的全拼（如张三同学，文件中的内容即为 zhangsan）。编写 c 语言程序实现以下功能：首先打开该文件并输出文件的内容，之后将文件的内容修改为 May the force be with you, \${姓名全拼}!, 比如 May the force be with you, zhangsan!, 输出修改后文件的内容，最后关闭文件。要求使用到 open() read() write() close() 函数。请详细叙述你的操作过程以及操作过程的截图，并给出你所编写的 C 程序的代码。

```
#include<unistd.h>
#include<stdio.h>
#include<fcntl.h>
#include<string.h>
int main(){
    char* s1="May the force be with you,";
    char* s2="!";
    char buf[20]={0};
    int fd;
    if((fd=open("1.txt",O_RDWR))== -1){
        perror("open error");
        exit(1);
    }

    ssize_t size=read(fd,buf,20);
    printf("%s",buf);
    if(size== -1){
        perror("read error");
        exit(1);
    }
    lseek(fd, 0, SEEK_SET);
    write(fd,s1,strlen(s1));
    write(fd,buf,size);
    write(fd,s2,strlen(s2));
    close(fd);
    return 0;
}
```

(2) 使用 fopen() fread() fwrite() fclose() 函数再次实现上述要求。请详细叙述你的操作过程以及操作过程的截图，并给出你所编写的 C 程序的代码。

```
#include<stdio.h>
#include<string.h>
int main(){
    char* s1="May the force be with you,";
    char* s2="!";
    char buf[20]={0};
    FILE *file;
```

```

file=fopen("a.txt","r+");
fread(buf,20,1,file);
printf("%s",buf);
fseek(file, 0, SEEK_SET);
fwrite(s1,strlen(s1),1,file);
for(int i=0;i<20;i++)
{
    if(buf[i]=='\n')
    {
        buf[i]='\0';
    }
}
fwrite(buf,strlen(buf),1,file);
fwrite(s2,strlen(s2),1,file);
fclose(file);
return 0;
}

```

## 2. 目录

编写一个 myls 程序，要求输入一个参数代表指定目录，例如 ./listdir/，打印目录下所有文件的名称。

(Hint: 参考课本例 4-8)

```

#include <sys/types.h>
#include <dirent.h>
#include <stdio.h>
int main( int ac, char **av)
{
    DIR * dir_ptr;
    struct dirent *dir;
    if (ac<2)
    {
        printf("usage: listdir directoryname");
        exit(1);
    }
    dir_ptr = opendir(av[1]);
    if(dir_ptr!=NULL)
    {
        while((dir=readdir(dir_ptr))!=NULL )
        printf("%s\n",dir->d_name);
        closedir(dir_ptr);
    }
}

```

## 3. 链接

(1) 创建文件 ~/srcfile，使用 ln 命令创建 srcfile 的软链接文件 ~/softlink，给出使用的命令；使用 ls -l 查看 ~，观察 softlink 的文件大小，并解释为什么；使用 ln 命令创建 srcfile 的硬链接文件 ~/hardlink，给出使用的命令；使用 ls -l 观察 srcfile 硬链接数的变化。

```
vim srcfile
```

A terminal window with a black background and yellow text displaying "Hello, world!".

```
ln -s srcfile softlink
# softlink 的文件大小为7字节，软链接存储源文件的路径
ln srcfile hardlink
# srcfile硬链接数加1
```

(2) 查看 `srcfile` 链接文件的索引节点号和文件内容。接下来修改源文件、硬链接文件、软链接文件，查看其他两个文件内容的变化。然后删除源文件，观察硬链接文件和软链接文件的变化，请给出操作过程的截图以及得出的结论。

硬链接文件hardlink: A terminal window showing the command `ls -li` output for the file `hardlink`, displaying "13 Oct 20 17:22 hardlink".

- 索引节点号: 13

- 文件内容: A terminal window with a black background and yellow text displaying "Hello, world!".

软链接文件softlink:

A terminal window showing the command `ls -li` output for the file `softlink`, displaying "7 Oct 20 17:25 softlink -> srcfile".

- 索引节点号: 7

- 文件内容: A terminal window with a black background and yellow text displaying "Hello, world!".

修改源文件: A terminal window with a black background and yellow text displaying "Hello, world!" followed by "Talk is cheap, show me the code." on the next line.

查看硬链接文件hardlink的内容:

A terminal window with a black background and yellow text displaying "Hello, world!" followed by "Talk is cheap, show me the code." on the next line.

查看软链接文件softlink的内容:

A terminal window with a black background and yellow text displaying "Hello, world!" followed by "Talk is cheap, show me the code." on the next line.

修改硬链接文件hardlink:

A terminal window with a black background and yellow text displaying "Hello, world!" followed by "Talk is cheap, show me the code." and "Change of hardlink" on the next line.

```
Hello, world!  
Talk is cheap, show me the code.  
Change of hardlink  
~  
~
```

```
Hello,world!  
Talk is cheap,show me the code.  
Change of hardlink  
~  
~
```

```
Hello, world!  
Talk is cheap, show me the code.  
Change of hardlink  
Change of softlink
```

```
Hello, world!  
Talk is cheap, show me the code.  
Change of hardlink  
Change of softlink
```

```
Hello, world!  
Talk is cheap, show me the code.  
Change of hardlink  
Change of softlink  
~  
~
```

```
rm -f srcfile
```

```
Hello,world!  
Talk is cheap,show me the code.  
Change of hardlink  
Change of softlink
```

- 硬链接文件的大小和源文件的大小一样，但是软链接文件的大小通常是源文件名的字节数
- 不论硬链接文件还是软链接文件，内容都和源文件srcfile一样
- 修改硬连接文件或者符号链接文件，另两个文件同样变化
- 删除源文件，硬链接文件依然存在，不受影响，但是软链接文件变成死链接，文件打不开

