**上海电力学院 LINUX/UNIX 操作系统**

**实验报告说明：**

**实验报告的一二三四部分阅读后不做改动。**

**第五部分实验结果记录，主要是每道题的答案的展示，按照题目要求，**

**有每一题的代码截图和程序运行结果的截图，以及问答题的文字答案，**

**流程图等。注意截图时截取有效信息。**

**第六部分实验结果分析，主要是每道题的解题过程的简要说明，以及**

**对实验结果的简要分析。**

**第七部分实验心得是对该实验的总体感悟。**

**实验时，所有以学号 20150001 为例的文本都要求改为你个人的学号。**

Linux 程序设计实验报告 5

——文件管理

**一、实验目的**

1．了解结构体 dirent，掌握目录文件的操作函数 opendir、closedir、readdir、scandir 的

使用。

2．掌握文件阻塞与非阻塞 I/O 的操作。

3．掌握函数 system 、chmod、stat 的应用。

1. 熟悉结构体 stat，掌握函数 stat 的应用。

二、实验任务与要求

1. 应用readdir函数显示文件和子目录。
2. 阻塞I/O文件操作的程序设计。
3. 创建文件，修改并读取文件的属性信息。
4. 判断文件的类型。

三、**实验工具与准备**

计算机 PC 机，Centos 操作系统或 Ubuntu 操作系统

**四、实验步骤与操作指导**

1. 调试下列程序。程序的功能是要求打印系统目录“/etc/rc.d”下所有的文件和子目录的名字。如果你的 LINUX 版本中没有/etc/rc.d 目录，打印/etc/下所有文件和子目录的名字。程序代码如下：

#include<stdio.h>

#include<sys/types.h>

#include<dirent.h>

#include<unistd.h>

int main()

{

DIR \* dir;

struct dirent \* ptr;

dir=opendir("/etc/rc.d");

printf("/etc/rc.d 目录中文件或子目录有:\n");

while((ptr = readdir(dir))!=NULL)

{

printf("%s\n",ptr->d\_name);

}

closedir(dir);

}

问题：

（1）修改程序，要求读取“/etc”目录下所有的目录结构，并依字母顺序排列。 #include<dirent.h>

考虑以下语句：

struct dirent \* \* namelist;

n=scandir("/etc",&namelist,0,alphasort);

程序段：

while(n--)

{

printf("%s\n", namelist[n]->d\_name);

free(namelist[n]);

}

2．调试下列程序。

程序主要运用一个非阻塞 I/O 的例子，程序打开当前终端文件/dev/tty，在打开时指定

O\_NONBLOCK 标志。程序运行时每隔一定时间（6 秒）等待用户从终端输入，如无输入时

循环 5 次，共等待 30 秒，每次等待时屏幕都有提示“try again”，30 秒后程序继续执行主程

序，输出以下图形后结束。

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

#include <unistd.h>

#include <fcntl.h>

#include <errno.h>

#include <string.h>

#include <stdlib.h>

#define MSG\_TRY "try again\n"

#define MSG\_TIMEOUT "timeout\n"

int main(void)

{

char buf[10];

int fd, n, i, j;

fd = open("/dev/tty", O\_RDONLY|O\_NONBLOCK); if(fd<0) {

perror("open /dev/tty");

exit(1);

}

for(i=0; i<5; i++)

{

n = read(fd, buf, 10);

if(n>=0)

break;

if(errno!=EAGAIN) {

perror("read /dev/tty");

exit(1);

}

sleep(6);

write(STDOUT\_FILENO, MSG\_TRY, strlen(MSG\_TRY));

}

if(i==5)

write(STDOUT\_FILENO, MSG\_TIMEOUT, strlen(MSG\_TIMEOUT));

else

write(STDOUT\_FILENO, buf, n);

for(i=0;i<5;i++)

{

for(j=0;j<=i;j++)

printf("%2c",'\*');

printf("\n");

}

close(fd);

return 0;

}

程序运行时，轮询等待用户的输入，等待期间如有输入，即转入主程序执行，如没

有输入，30 秒后执行主程序。

**[选做]：** 把以上程序改为阻塞I/O.

1. 程序设计。

设计程序应用 system 函数建立/home/liu 文件，应用 chmod 函数使文件 liu 具有 S\_IRUSR、S\_IWUSR、S\_IRGRP、S\_IROTH 属性，最后应用函数 stat 获取文件的大小与建立的时间。提示： （1）创建文件的命令 touch

（2）用到的函数： system 函数、chmod 函数、stat 函数

（3）参照例题：例 5.1，例 5.2，例 5.4，期中包括 system 函数、chmod 函数、stat

函数的使用。

（4）时间的显示用 ctime 函数，考虑使用 ctime(&buf.st\_ctim.tv\_sec).

1. 程序设计。

设计一个程序，要求判断“/etc/passwd”的文件类型。

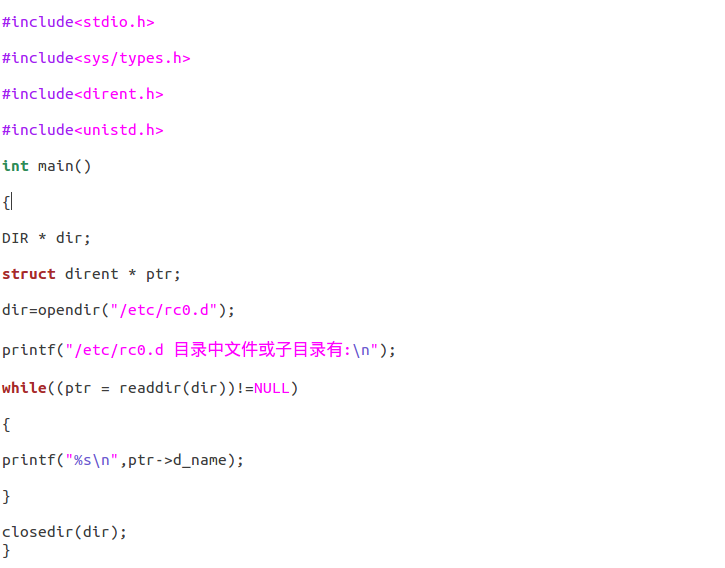
提示：读取 stat 结构体中的 st\_mode 属性，可以使用几个宏来判断：S\_ISLNK(st\_mode)是否是一个连接，S\_ISREG 是否是一个常规文件 S\_ISDIR 是否是一个目录，S\_ISCHR 是否是一个字符设备，S\_ISBLK 是否是一个块设备，S\_ISFIFO 是否是一个 FIFO 文件，S\_ISSOCK

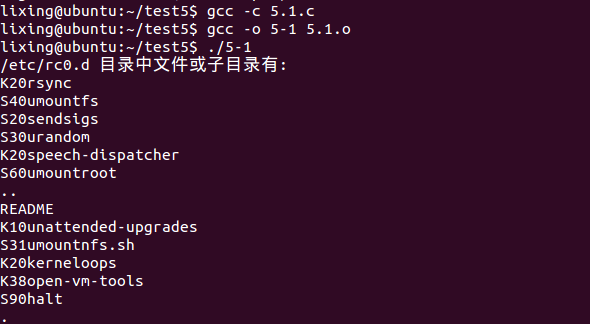
是否是一个 SOCKET 文件。

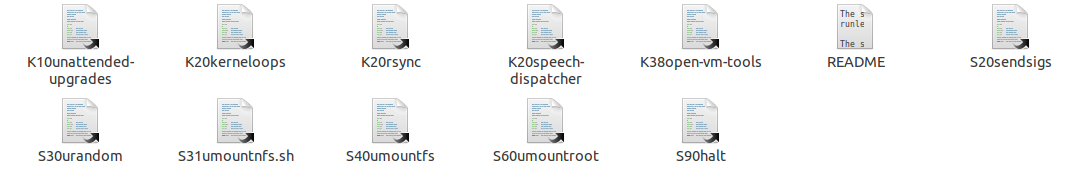
参考程序： 例 5.4 后的程序扩展阅读。

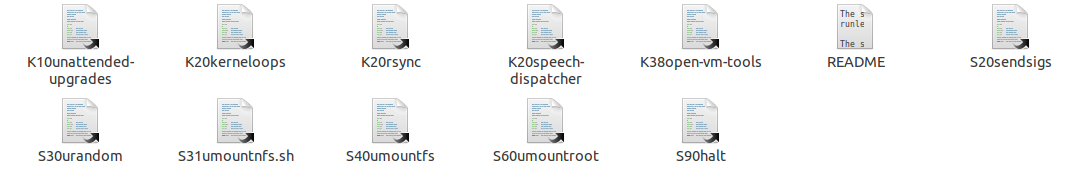
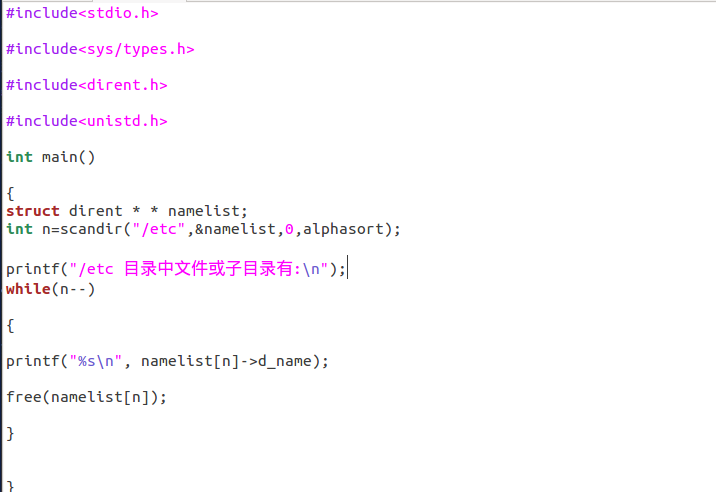
【选做】修改程序，使得程序可以判断用户输入的任意文件的文件类型。

**五、实验结果记录**

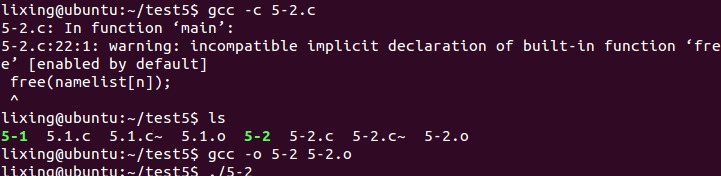
1.

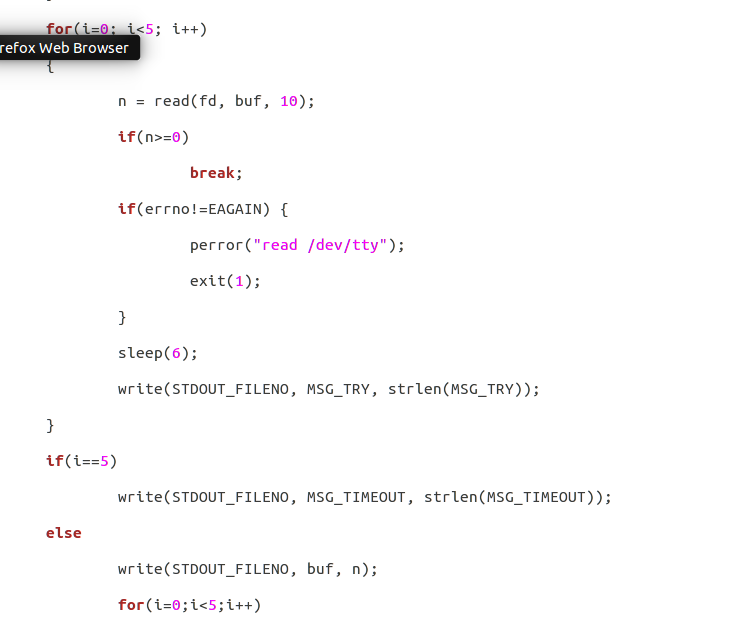




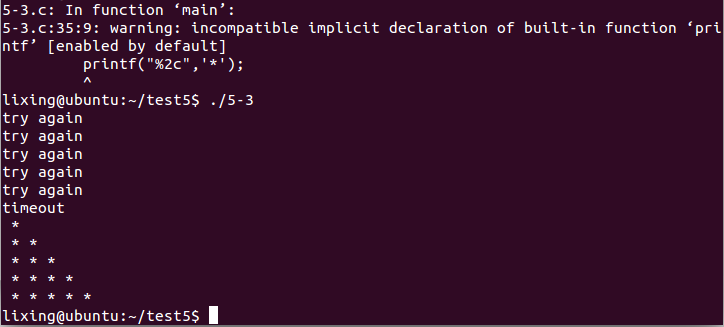
1.2

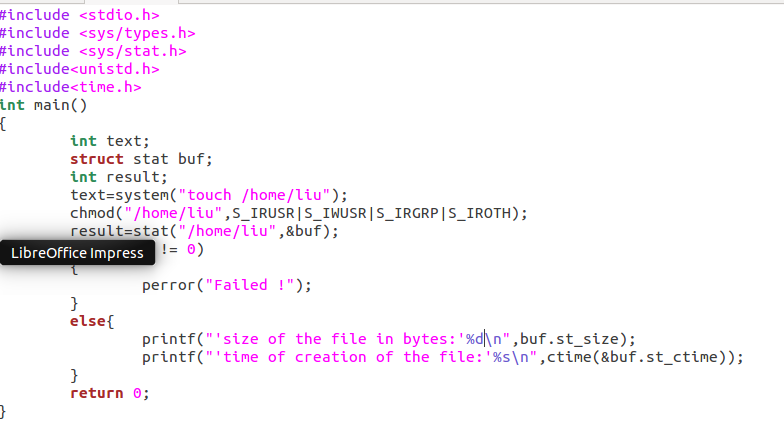


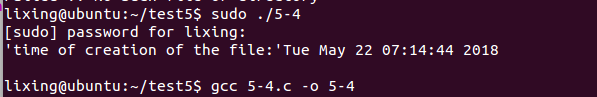
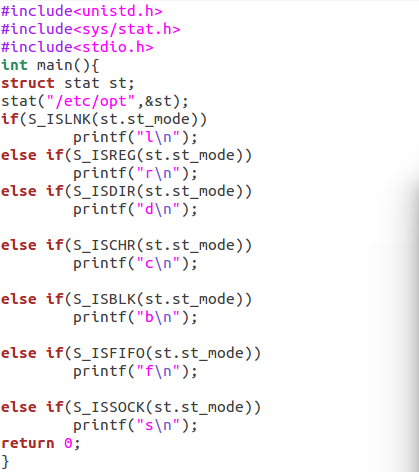




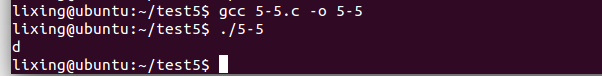
2.1

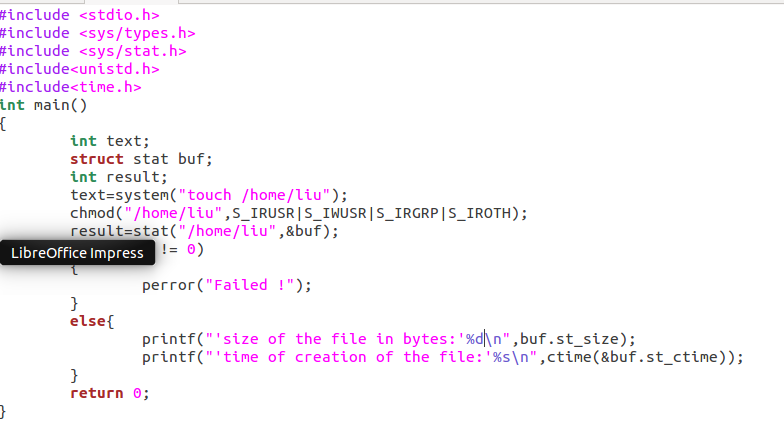
3





4





**六、实验结果分析**

1.readdir函数显示文件和目录 n=scandir(“/etc”,&namelist,0,alphasort)进行文件名排序

2.

3.遇到问题 warning: incompatible implicit declaration of built-in function ‘printf’ [enabled by default]

printf("%2c",'\*');

网上说缺少#include<stdio.h> ，我写上之后仍存在同样的问题，只是warning,对实验结果没有影响。

4.遇到问题 warning: format ‘%s’ expects argument of type ‘char \*’, but argument 2 has type ‘\_\_off\_t’ [-Wformat=]

printf("'size of the file in bytes:'%s\n",buf.st\_size);

1. **实验心得**

遇到很多问题，搞了好久， warning: format ‘%s’ expects argument of type ‘char \*’, but argument 2 has type ‘\_\_off\_t’ [-Wformat=]

printf("'size of the file in bytes:'%s\n",buf.st\_size);

这个问题还是没有解决，类型错误吧，加了&也是不行，需要把课本重新看一遍了。