

习题课（1-6章）

第一章 概述

- P41/3, 5
- 3、Linux主要包括什么？
 - 内核、shell、文件系统和应用程序
- 5、什么是Linux的内核版本和发行版本
 - 内核版本：Linux开发者Linus Torvalds本人领导的系统内核的版本号，全球唯一，内核版本号由3个数字组成 r.x.y
 - 发行版本：一些厂家将Linux内核和应用软件包装起来，提供安装界面和配置工具，构成发行套件，例如，Ubuntu，Fedora，RedHat等。

第二章 文件系统

- 当前目录是/, 写出删除/home/jkx/abc目录的命令, 该目录中有文件。

`rm -rf /home/jkx/abc`

- Linux系统中, 文件系统的根目录为__/_
- 显示当前所在目录的命令是 __`pwd`__
- 比较两个文件不同之处的命令是 __`diff`__
- 文件属性-r-xr-xr--对应的8进制表示为 __`554`__。

第二章文件系统

- Vi编辑中，在命令模式下，将当前光标所在行开始的3行内容复制到文件开头的命令集合是_____。

3yy

:0

p

- 用grep命令在文件file.txt中查找包含头两个是大写字母，中间两个任意字符，以a结尾的单词的行，写出具体命令。

grep "[A-Z][A-Z]..a" file.txt

第二章文件系统

- 2,4,6,8,11,12,13,14,15

2、什么是虚拟文件系统，Linux为什么采用虚拟文件系统？ P47

答：虚拟文件系统是Linux 内核中的一个软件层，用于允许不同的文件系统共存，并给用户程序提供统一的文件函数接口。虚拟文件系统还允许跨文件系统操作。

4、什么是文件？Linux下主要有哪些类型文件？

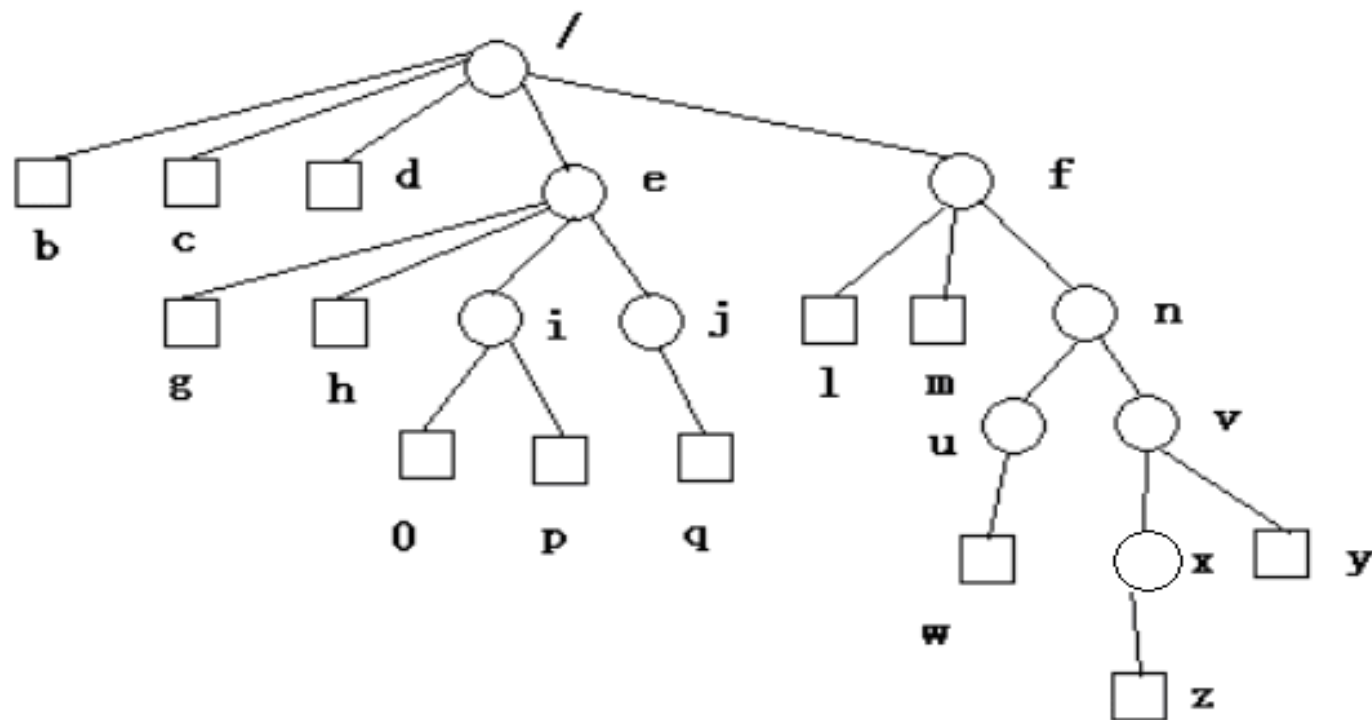
P47-48

答：文件是指由创建者定义的，具有文件名的一组相关元素的集合，文件可以是文本文档、图片、程序等。

Linux下的文件类型包括：

- 普通文件：Linux中最一般格式的文件；
- 目录文件：由文件目录信息构成的特殊文件，用来检索普通文件的目录信息；
- 设备文件：Linux将设备当文件来操作，分为字符设备文件和块设备文件，前者例如鼠标、键盘、屏幕这种输入输出低速设备，后者例如磁盘、光盘这种高速设备；
- 符号链接文件：一种特殊类型的文件，它的内容只是一个字符串，该字符串可能指向一个存在的文件也可能什么都不指向。

6、根据下图，圆圈代表目录，方框代表文件，当前目录为n，用相对路径法和绝对路径法分别写出文件g、o、z的路径。



8、将主目录下的文件.bashrc复制到/tmp下，并重命名为bashrc，用命令实现上述过程。

答： `cp ~/.bashrc /tmp/bashrc。`

或者 `cp ~/.bashrc /tmp/
mv /tmp/.bashrc /tmp/bashrc`

11、将/home/stud1/wang目录做归档压缩，压缩后生成wang.tar.gz文件，并将此文件保存到/home目录下，用命令实现上述过程。

答： `tar zcvf /home/wang.tar.gz
/home/stud1/wang`

或者 `tar cvzf wang.tar.gz /home/stud1/wang
mv wang.tar.gz /home`

12、在Linux下有一文件列表内容如下：

（1）要完整显示如上文件类别信息，应该使用什么命令？

答：ls -l

（2）上述文件列表内容的第一列内容“lrwxrwxrwx”中的“l”是什么含义？对于其它类型的文件或目录还可能出现什么字符，它们分别表示什么含义？

答：“l”表示该文件是符号链接文件。“-”表示该文件是普通文件，“d”表示该文件是目录，“b”表示该文件是块设备文件，“c”表示该文件是字符设备文件。

(3) 上述文件列表内容的第一列内容“lrwxrwxrwx”中的第一、二、三个“rwx”分别代表什么含义？

答：第一个“rwx”表示该文件的文件主人具有读写执行的权限，第二个“rwx”表示该文件的主人同组用户具有读写执行的权限，第三个“rwx”表示该文件的其他用户具有读写执行的权限。

(4) 上述文件列表内容的第五列内容“6”是什么含义？

答：表示该文件的大小，6字节。

(5) 上述文件列表内容的最后一列内容“nurse2->nurse1”是什么含义？

答：表示文件nurse2是符号链接文件，其链接到文件nurse1。

13、在根目录下创建目录gdc，并设置权限为gdc的主人具有读写执行权限，与主人同组用户可读写，其它任何用户则只能读。

答：mkdir /gdc

chmod u=rwx,g=rw,o=r /gdc

或者 mkdir /gdc

chmod 764 /gdc

14、在用户user的主目录下新建目录software，并将路径/etc下所有以h开头的文件及目录拷贝到software中，用命令实现上述过程。

答：mkdir /home/user/software

cp /etc/h* /home/user/software

●

15. 什么是符号链接，什么是硬链接？符号链接与硬链接的区别是什么？

答： 符号链接文件包含到另一个文件的路径名。当用户需要在不同的目录下用到相同的文件时，不需要在每一个目录下都存放该文件，只要在某个固定的目录下存放，然后在其它的目录下用命令链接（link）它即可，不必重复的占用磁盘空间，

硬链接指通过索引节点来进行的链接。在Linux系统中，内核为每一个新创建的文件分配一个Inode（索引结点），文件属性保存在索引结点里，系统是通过索引节点（而不是文件名）来定位每一个文件。在Linux系统中，可以通过命令让多个文件名指向同一索引节点，这多个文件即为硬链接文件。

符号链接可以建立对于文件和目录的链接，硬链接只能建立对文件的链接；符号链接可以跨文件系统，即可以跨磁盘分区，硬链接不可以跨文件系统；符号链接的文件类型位是l，链接文件具有新的i节点，硬链接的文件类型位是“-”，且硬链接文件的i节点同被链接文件相同。

第三章系统管理

- 2,6,7,9

2、Linux分为几个级，分别有什么特点？ P91

6、Linux用户管理相关的文件有哪些？解释文件内容。

答：在Linux系统中，主要使用/etc目录下的三个文件来维护用户及用户组的相关信息。

(1) /etc/passwd文件，其中存放关于账户相关的信息

(2) /etc/shadow文件，/etc/passwd的影子文件，来存放用户的加密口令。

(3) /etc/group文件，其中存放用户组相关的信息资料。

第三章 系统管理

7、使用useradd命令添加用户student，然后使用userdel删除该用户。

答：

useradd student

passwd student

userdel student

第三章 系统管理

9、假设U盘的标识为sdb3，则把挂载U盘到目录/mnt/usb下的命令是什么？假设当前已经有/mnt/usb目录，是否会丢失原来该目录信息？如何卸载U盘？

答：

`#mount /dev/sdb3 /mnt/usb`

不会

`#umount /dev/sdb3 或者 umount /mnt/usb`

第四章 网络管理

- 实现samba服务器配置：
 - 三个目录/home/share/teacher, /home/share/class1, /home/share/class2
 - 五个用户： teacher, admin_class1,class1,admin_class2,class2
 - teacher用户对/home/share/teacher目录具有可写权限，对它其它目录具有只读权限
 - admin_class1用户对/home/share/class1目录具有可写权限，对它其它目录具有只读权限
 - admin_class2用户对/home/share/class2目录具有可写权限，对它其它目录具有只读权限
 - class1和class2用户对这三个用户具有只读权限
- 答案

第五章 Shell编程

- P182/2, 4, 9, 10

2,如何编辑一个脚本, 执行脚本有哪些方式?

vi或者gedit进行编辑

三种执行方式, sh 脚本名; .脚本名;

chmod+x 脚本名, ./脚本名

4、shell中都有哪些引号，他们之间的区别是什么？

答：

在shell中引号分为三种：单引号，双引号和反引号。

单引号（'）：括起来的字符都作为普通字符出现。特殊字符也失去原有意义，而只作为普通字符处理。

双引号（"）：除\$、\、'(反引号)特殊字符保留其特殊功能外，其余字符当做普通字符对待。

反引号（`）：括起来的字符串被shell解释为命令，在执行时，shell首先执行该命令，并以它的输出结果取代整个反引号（包括两个反引号）部分。

习题9

- 设计一个shell脚本：求命令行上所有整数和。
 - 预期：
 - 输入： ./q9.sh 1 a 2 3 b ， 输出 6
 - 输出： ./q9.sh 3 4 5， 输出 12
 - 两种方案：
 - for
 - 参考P175/for2_script
 - While
 - 将第一个参数进行累加
 - 巧用shift，将参数向左移，直至参数全部移走

For方案

```
sum=0
```

```
for var in $*
```

```
do
```

```
    let sum=sum+var
```

```
done
```

```
echo "$sum"
```

while方案

```
sum=0
```

```
while [ $# -gt 0 ]
```

```
do
```

```
    let sum=sum+$1
```

```
    shift
```

```
done
```

```
echo "$sum"
```

习题10

- 设计一个shell脚本：判断当前工作目录下所有的文件类型，如果是目录显示其包含的文件名，如果是文件查看文件内容，如果都不是，显示提示信息。
 - For file in ./*
 - ./*表示当前目录下的所有文件
 - 遍历当前目录下的每个文件，再进行判断,参考p171/elif1_script


```
for file in ./*
do
    if [ -f $file ] ; then
        echo "$file is ordinary file, its content :"
        cat "$file"
    elif [ -d $file ] ; then
        echo "$file is a directory, it includes:"
        ls "$file"
    else
        echo "$file is not ordinary or directory"
    fi
done
```

附加题：设计一个shell脚本：求命令行上整数的最大值和最小值。

```
min=$1
max=$1
shift
while [ $# -gt 0 ]
do
    if [ $1 -lt $min ];then
        min=$1
    fi
    if [ $1 -gt $max ];then
        max=$1
    fi
    shift
done
echo "the max is $max, the min is $min"
```

附加课堂KTQ2

- 编写一个shell脚本，输入一个数字 x ($x>0$)，显示从0开始的前 $x+2$ 个Fibonacci序列
- 0,1,1,2,3,5,8,13.....

```
echo -n " 0 1"
x1=0
x2=1
cnt=$1
while [ $cnt -gt 0 ]
do
    let x=x1+x2
    echo -n " $x"
    let x1=x2
    let x2=x
    let cnt=cnt-1
done
echo ""
exit 0
```

第五章课堂作业

(1) 当前目录有一名为script的可执行脚本，不可以运行的方法是：

- A、 ./script B、 sh script
- C、 . script D、 script

(2) 命令ls /home/`whoami`/ 的执行结果是什么？

whoami:打印出当前正在操作的用户名

(3) 写出命令，查找当前用户主目录下所有以d开头的文件并排序，只查找一层，不递归

find ~ -maxdepth 1 -name "d*" | sort

第五章其他作业的解读

P182/5

```
#!/bin/bash
```

```
# finduser----寻找是否有第一个参数指定的用户登录
```

```
if [ $# -ne 1 ]
```

```
then
```

```
echo usage: finduser username
```

```
exit 1
```

```
fi
```

```
who | grep $1
```

who命令能够打印 当前都有谁登录到系统中的相关信息。who命令只会显示真正登录到系统中的用户。它不会显示那些通过**su**命令切换用户的登录者。

习题6

```
echo "enter username: "  
read user  
until who | grep "$user" > /dev/null  
do  
sleep 30  
done
```

习题6

- `> /dev/null`

把一个文件move到/dev/null，那他就永远消失了，就想掉进了黑洞里。并不是什么具体的设备。等价于删除

- `sleep 30`

bash中sleep还支持睡眠（分，小时）

`sleep 1` 睡眠1秒

`sleep 1s` 睡眠1秒

`sleep 1m` 睡眠1分

`sleep 1h` 睡眠1小时

习题7

```
#!/bin/bash
if [ $# -ne 2 ]
then
    echo "usage: $0 mdays size " 1>&2
    exit 1
fi
if [ $1 -lt 0 -o $1 -gt 30 ]
then
    echo "mdays is out of range"
    exit 2
fi
```

习题7

```
if [ $2 -le 20 ]
```

```
then
```

```
    echo "size is out of range"
```

```
    exit 3
```

```
fi
```

```
find ./ -xdev -mtime -$1 -size +$2c -print
```

- find命令的几个选项含义：

- -xdev: 仅仅会在当前文件系统中搜索
- -mtime -n +n : 按文件更改时间来查找文件，-n指n天以内，+n指n天以前
- -size n[c]: 查长度为n块[或n字节]的文件

习题7

- 整个shell脚本的意思：从当前目录开始查找所有\$1天内修改，大小小于\$2块的文件。 1块=512字节

第六章 Linux下C编程

- P230/ 5, 6
- 当前目录是/home/jkx/work, 里面有一个名为test.c的文件, test.c依赖位于/home/jkx/includes目录下的头文件defs.h, 和位于/home/jkx/libs目录下的库文件libmy.so。请写出将test.c生成可执行文件test的gcc命令。
- `gcc -o test test.c -I../includes -L../libs -lmy`

5、makefile文件的作用是什么？其书写规则是怎样的？

答：要用make维护一个程序，必须创建一个makefile文件，makefile文件告诉make以何种方式编译源代码和链接程序。makefile有自己的书写格式、关键字、函数，像C语言有自己的格式、关键字和函数一样，makefile描述规则组成如下所示。

目标：依赖文件

[TAB]命令

6、设某个程序由四个C语言源文件组成，分别是a.c、b.c、c.c、d.c，其中b.c和d.c都使用了defs.h中的声明，最后生成的可执行文件名为pgm。试为该程序编写相应的makefile文件。

```
pgm : a.o b.o c.o d.o
    gcc -o pgm a.o b.o c.o d.o
a.o : a.c
    gcc -c a.c
b.o : b.c defs.h
    gcc -c b.c
c.o : c.c
    gcc -c c.c
d.o : d.c defs.h
    gcc -c d.c
```