作业1

**161250041 侯韵晗**

1.用命令完成以下功能（写出对应的命令，要求命令尽可能简单[能一行命令完成的尽量一行命令完成, 不能使用课上教的Shell脚本或者编程方法]）

a)将当前目录下所有文件备份并压缩成tar.gz，如果成功再将备份拷贝到/backup目录下。

**mkdir ./backup; tar zcvf files.tar.gz ./\* && cp files.tar.gz ./backup**

b) 将上述备份压缩后的文件解压到原来的目录。

**tar zxvf ./backup/files.tar.gz -C ./**

c) 将当前目录下所有后缀为.html的文件的后缀改为.htm。

两种方法：

**rename  .html .htm \*.html //c**

**rename ‘s/\.html$/\.htm/g’ \*.html //perl**

d）将当前目录下所有子目录下以a开头的.cpp文件改名为以b开头的.c文件。

**find . -name "a\*.cpp" -exec rename 's/(.\*\/)a([^\/]\*)\.cpp$/$1b$2\.c/g' {} \;**

2.自己查阅Linux相关书籍资料，学习并完成以下作业题：

a) 请简要叙述DOS/Windows中的文本文件，UNIX/Linux中的文本文件以及Mac中的文本文件有何不同之处，并给出至少一种你认为较为实用的转换方案。(需要两两给出转换方案)

**三个系统的文本文件的换行不同**

DOS/Windows:’\r\n’ UNIX/Linux:’\n’ Mac:’\r’

**转换方案**

Windows->Linux : cat win.txt | sed -i 's/\r$//' > linux.txt 或 cat win.txt|tr -d "\r" > linux.txt （以下就省略sed方法）

Linux->Windows : cat linux.txt | sed -i 's/$/\r/' > win.txt

Windows->Mac : cat win.txt | tr -d "\n" > mac.txt

Mac->Windows : cat mac.txt | tr "\r" "\n" | sed 's/$/\r/' > win.txt

Linux->Mac : cat linux.txt | tr "\n" "\r" > mac.txt

Mac->Linux : cat mac.txt | tr "\r" "\n" > linux.txt

b) 请查阅Linux系统中/etc/fstab文件各字段功能，并对你自己安装的Linux系统中/etc/fstab的各个字段做出解释。

**/etc/fstab 文件包含了如下字段，通过空格或 Tab 分隔：**

**<file system> <dir> <type> <options> <dump> <pass>**

**<file systems>** 要挂载的分区或存储设备.

**<dir>** <file systems>的挂载位置。

**<type>** 要挂载设备或是分区的文件系统类型，支持许多种不同的文件系统：ext2,ext3,ext4,reiserfs,xfs,jfs,smbfs,iso9660,vfat,ntfs,swap及 auto。设置成auto类型，mount 命令会猜测使用的文件系统类型，对 CDROM 和 DVD 等移动设备是非常有用的。

**<options>**挂载时使用的参数，注意有些mount 参数是特定文件系统才有，参数auto noauto exec noexec …

**<dump>** dump 工具通过它决定何时作备份. dump 会检查其内容，并用数字来决定是否对这个文件系统进行备份。 允许的数字是 0 和 1 。0 表示忽略， 1 则进行备份。大部分的用户是没有安装 dump 的 ，对他们而言 <dump> 应设为 0。

**<pass>** fsck 读取 <pass> 的数值来决定需要检查的文件系统的检查顺序。允许的数字是0, 1, 和2。 根目录应当获得最高的优先权 1, 其它所有需要被检查的设备设置为 2. 0 表示设备不会被 fsck 所检查。

c) 用命令行实现： 从光盘制作一个.iso文件，如果成功将光盘弹出。

**cp /dev/cdrom /root/mydisk.iso && eject /dev/cdrom**

**或**

**dd if=/dev/cdrom of=/root/mydisk.iso && eject /dev/cdrom**

**附加题：**

假设当前Linux系统的目录/tmp/gsl下的文件结构如下图所示，其中有很多Makefile分散在各个子目录下(但并非每个子目录都存在Makefile，例如block目录下就没有)。

.

|-- INSTALL

|-- Makefile

|-- NEWS

|-- ...

|-- blas

| |-- ChangeLog

| |-- Makefile

| |-- TODO

| |-- blas.c

| |-- ...

|-- block

| |-- ChangeLog

| |-- block.c

|-- bspline

| |-- Makefile

| |-- ...

|-- ...

要求将这些Makefile拷贝到目录/home/usrname/gsl的对应子目录下(假设/home/usrname/gsl下已经存在blas，bspline等子目录)，使得拷贝后/home/usrname/gsl下的文件如下图所示：

.

|-- Makefile

|-- blas

| |-- Makefile

|-- bspline

| |-- Makefile

|-- ...

\*要求：该要求实际上是可以用一行命令完成的，难度较高。同学们实际做作业时，允许使用两到三行命令，但不允许使用任何编程实现(即不允许Shell脚本的循环，分支等控制流程，可以使用管道)。

\*提示：利用find，sed和sh命令。

**答**：

**find** /tmp/gsl -type f -name 'Makefile' -print | **sed** 's!\/tmp\/gsl(/.\*)!\/tmp\/gsl$1 \/Home\/usrname\/gsl$1!' | **sed** 's/^/cp /'|**sh** -x