

作业须知：请特别注意截止时间，老师会在教学系统里多设定一周，以防止因系统不稳定而导致有的同学未能及时提交，请同学们务必提早一周提交。超过截止时间将不再收取作业，请不要通过Email发送作业。

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。

b) 将上述备份压缩后的文件解压到原来的目录。

c) 将当前目录下所有后缀为.html的文件的后缀改为.htm。

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。

这边不是，是。

前面是压缩到文件的名称 后面是要压缩的目录

a. tar czvf ./file.tar.gz ./ && cp ./file.tar.gz /backup/file.tar.gz

b. tar xzvf file.tar.gz -C

c. rename html hml *.html

(或使用find: find . -name "*.html" | xargs mv -i -t ./

{ }.html ./ { }.hml)

d. find -name "a*.cpp" | xargs mv -i -t { }/a { }.cpp { }/b { }.c

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）

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。

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。

a. DOS/Windows中的文本文件的换行符是\r\n；Unix/Linux中的换行符是\n；Mac中的文本文件中的换行符是\r。

Dos -> Linux: sed -i 's\r&/' ' file.txt

Linux -> Dos: sed -i 's/\$^r/' ' file.txt

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

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

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

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

win.txt

b. 文件/etc/fstab存放的是系统中的文件系统信息

字段名	功能
fs_spec	定义希望加载的文件系统所在的设备或远程文件系统
fs_file	描述希望的文件系统加载的目录点

fs_type	定义啦设备上的文件系统（Linux下一一般为ext2）
fs_options	指定加载该设备的文件系统是否需要特定的参数选项，参数选项如下：
no	以只读模式加载系统
sync	不对该设备的写操作进行写缓冲处理
user	允许普通用户加载该文件系统
quota	强制在该文件系统上进行磁盘定额配置
noauto	不再使用mount -a命令来加载该文件系统
fs_dump	检查一个文件系统应该以多快的频率进行转储，若不需要转储则置0
fs_pass	决定在启动时需要被扫描的文件系统的顺序（根文件系统对应的值应该为1，其他文件系统应该为2，若无需启动时扫描则置0）

我的系统中的各个字段

文件系统名 fs_spec	挂载目录点 fs_file	文件系统 fs_type	选项 fs_options	fs_dum p	fs_pass
/dev/mapper/ centos-root	/	xfs	defaults	0	0
UUID=6ead2962-1b31-4618-9a9a-0c79e8358a69	/boot	xfs	defaults	0	0
/dev/mapper/ centos-swap	/swap	xfs	defaults	0	0

c. 将光盘放入光驱，执行 dd if=/dev/cdrom of=/home/sunky/mydisk.iso && eject

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/username/gsl的对应子目录下(假设/home/username/gsl下已经存在blas, bspline等子目录)，使得拷贝后/home/username/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。

```
find /gsl -name 'Makefile' | awk '{n=$0; sub(/\gsl/, "/home/username/gsl");
system("cp" "n" "$0")}'
```