计算机基础

——Linux、Vim、Shell

荆茂强

高能物理研究所

中国科学院高能物理研究所

Linux

Linux是什么？

- Linux其实是一个操作系统平台，我们平时常用的操作系统叫作Windows。
- 也有不少同学使用的苹果电脑，苹果电脑所用的系统叫做MacOS。
- 也有一种系统叫作Unix，Unix是比Linux更加古老的一种系统，多用在服务器领域。
- Linux同样也用在服务器领域，大家熟知的BAT（百度，阿里，腾讯）公司、Google、Facebook等一线互联网大公司的服务器99%的操作系统都是Linux，大家用的Android手机也是Linux操作系统。→冬至饮雪

Linux系统目录结构

- **/bin**: bin是binary的缩写，该目录下存放的是最常用的命令。
- **/boot**: 该目录下存放的是启动Linux时使用的一些核心文件，包括一些链接文件以及镜像文件。
- **/dev**: dev是Device（设备）的缩写。该目录下存放的是Linux的外部设备。
- **/home**: 这是用户的家目录。在Linux中，每个用户都有一个自己的目录，一般该目录名是以用户的账号命名的。
- **/root**：该目录是系统管理员的用户家目录。
- **/usr**：这是一个非常重要的目录，类似于Windows下的Program Files目录，用户的很多应用程序和文件都存放在该目录下。
- **/usr/bin**：该目录存放的是系统用户使用的应用程序。

Linux命令行中常用的快捷键

- **Ctrl+C**:
  - 结束（终止）当前命令。如果你输入了一大串字符，但不想运行，可以按Ctrl+C组合键，此时光标将跳入下一行，而在刚刚的光标处留下一个^C的标记。
- **Tab**:
  - 实现自动补全功能。当你输入命令、文件或目录的前几个字符时，它会自动帮你补全。
- **Ctrl+Z**:
  - 暂停当前进程。这和Ctrl+C是有区别的，暂停后使用fg命令恢复该进程。
- **Ctrl+A（E）**:
  - 可以让光标移动到命令行的最前面（最后面）

Linux文件和目录管理

绝对路径和相对路径

- 路径就是文件存放的位置，只要告诉系统某个文件的路径，系统就可以找到这个文件，在Linux中存在着绝对路径和相对路径
  - 绝对路径：路径的写法一定是由根目录/写起的，例如/usr/local/bin。
  - 相对路径：路径的写法不是由根目录/写起的。例如，首先用户先进入到/home，然后再进入test，执行命令为

    # cd /home

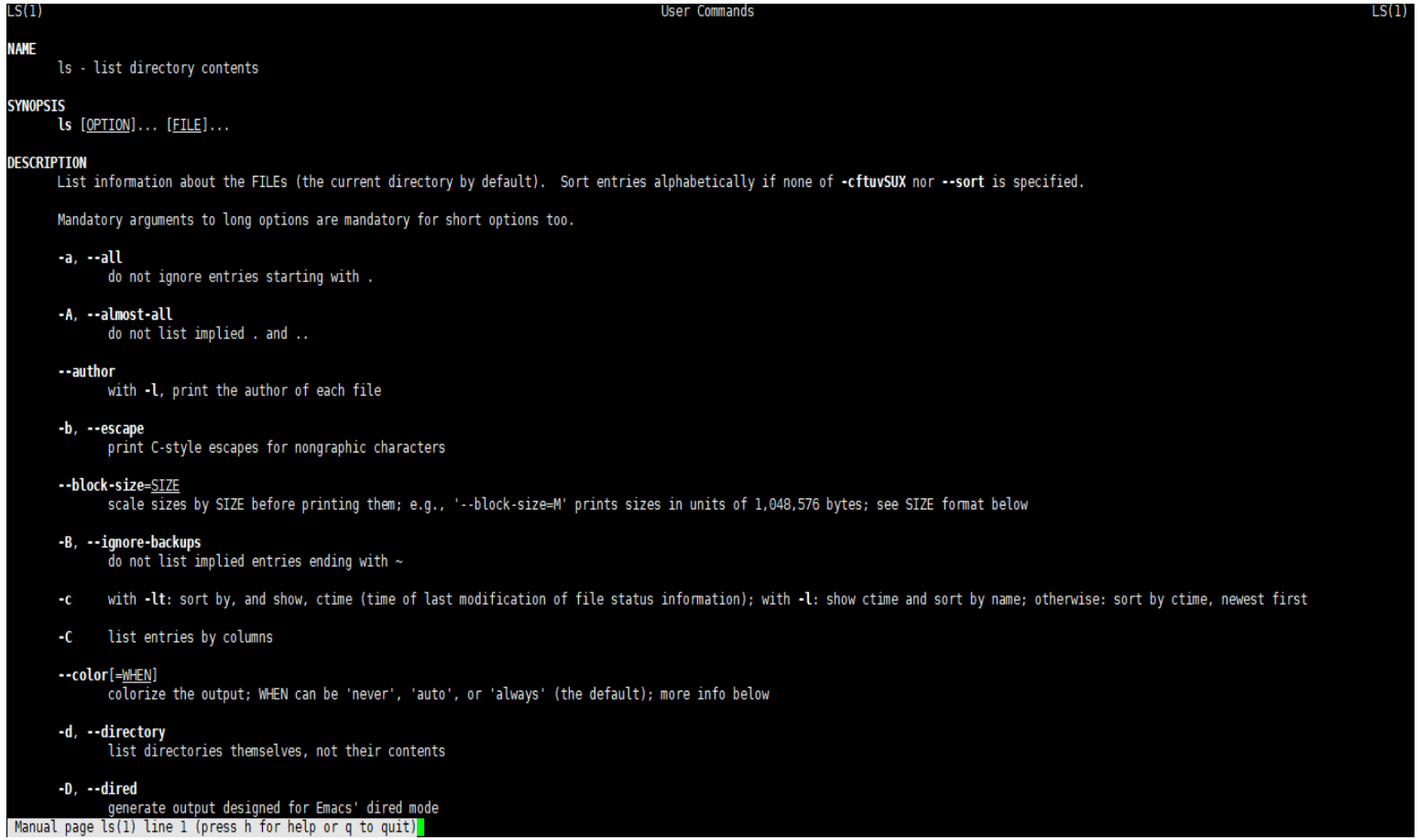
    # cd test

    此时用户所在的路径为/home/test。第一个cd命令后紧跟/home，前面有斜杠；而第二个cd命令后紧跟test，前面没有斜杠。这个test是相对于/home目录来讲的，所以称为相对路径。

学会查询帮助文档——man

- man命令用于查看命令的帮助文档，其格式为man 命令。例如，输入如下命令：
  - # man ls

这样可以查看ls命令的帮助文档。



如果屏幕不能显示完全，可以按空格键下翻，或者按上下方向键前后移动文本，若想退出帮助文档，按字母键q

命令——cd

- 命令cd（change dictionary的缩写）是用来变更用户所在的目录。
- 如果cd后面什么都不跟，就会直接进入当前用户的根目录下。

  # cd

  # pwd

  命令pwd用于显示当前所在的目录。
- 命令cd后面只能是目录名，如果跟了文件名，则会报错。
- 在Linux文件系统中，有两个特殊的符号也可以表示目录：“.”表示当前目录，“..”表示当前目录的上一级目录。
- cd ..：返回上一级目录。
- cd -：返回进入当前目录前的目录。

命令——ls

- 后面不加任何选项也不跟目录名或者文件名：会列出当前目录下的文件和目录，不包含隐藏文件。
- 后面加-a选项、不加目录名或者文件名：会列出当前目录下所有问价和目录，含有隐藏文件。
- 后面加-l选项，不加目录名或者文件名：会列出当前目录下除隐藏文件外的所有文件和目录的详细信息，包含其权限、所属主、所属组以及文件创建日期和时间。
- 后面不加选项、只跟文件名：会列出该文件，使用时通常都是加上-l选项，用来查看该文件的详细信息。
- 后面不加选项，只跟目录名：会列出指定目录下的文件和目录。
- 后面加-d选项，只跟目录名：只会列出指定的目录。

命令——mkdir

- 命令mkdir（make dictionary的缩写）用于创建目录，格式为mkdir [-mp] [目录名称]。
  - -m选项用于指定要创建目录的权限
  - -p选项非常有用：

    # mkdir /tmp/test/123

    mkdir: 无法创建目录 ‘/tmp/test/123’ ：没有那个文件或目录

    # mkdir –p /tmp/test/123

    # ls /tmp/test

    123
  - 查看目录的属性：

    ls –ld /tmp/test/123
- 命令rmdir（remove directory的缩写）用于删除空目录。

命令——rm

- 命令rm是最常用的，它也有很多选项。
  - -r：删除目录用的选项，类似于rmdir，但可以删除非空目录。

    # mkdir –p /tmp/test/123

    # rm –r /tmp/test/123

    rm：是否删除目录 ‘/tmp/test/123’ ？y
  - -f：表示强制删除。他不再询问是否删除，而是直接删除。如果后面跟一个不存在的文件或者目录，则不会报错。但如果要删除一个存在的目录时，即使加上-f选项也会报错。所以使用命令rm删除目录时，一定要加上-r选项。
- 命令rm 使用最多的是-rf选项，这样删除文件或目录比较方便，但请大家千万要注意，rm –rf命令后面不能加 “/” ！

环境变量——PATH

- 在讲环境变量之前，需要先介绍下命令which，它用于查找某个命令的绝对路径。

```
# which rmdir
/usr/bin/rmdir
# which rm
alias rm=  ‘rm -i’
    /usr/bin/rm
# which ls
alias ls=  ‘ls –color=auto’
    /usr/bin/ls
```

rm和ls是两个特殊的命令，我们使用alias命令做了别名，我们实际上是rm –i，加上-i选项后，删除文件或者命令时都会询问是否确定要删除。

环境变量——PATH

- 在上页的示例中，用which查到rm命令的绝对路径为/usr/bin/rm，为什么我们使用命令时，只是直接打出了命令，而没有使用这些命令的绝对路径呢？这是环境变量PATH在起作用。

  # echo $PATH

  /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin

  值得注意的是，由于PATH里没有/root目录，如果你将ls移动到/root目录下，当执行ls命令时，系统自然就找不到可执行文件了，它会提示command not found！

命令——cp

- cp是copy（即复制）的简写，该命令的格式为：cp [选项] [来源文件] [目的文件]
  - -r：如果要复制一个目录，必须加-r选项，否则不能复制，这类似于rm命令
  - -i：这是安全选项，如果遇到一个已存在的文件，会询问是否覆盖，这也与rm命令类似。在RadHat/CentOS系统中，使用cp命令其实是cp –i，可以通过which命令查看：

```
# which cp
alias cp=  ‘cp -i’
           /bin/cp
```

命令——mv

- mv是move的简写，该命令的格式为：mv [源文件或目录] [目的文件或目录]

命令——cat

- 命令cat是比较常用的一个命令，用于查看一个文件的内容并将其显示在屏幕上。Cat可以不加任何选项，直接跟文件名，下面介绍它的两个常用选项
  - -n：查看文件时，把行号也显示在屏幕上。

    # echo ‘11111111’ > filee

    # echo ‘22222222’ >> filee

    # cat dirb/filee

    # cat –n filee
  - -A：显示所有内容包括特殊字符。

    # cat –A filee

命令——tac

- 和命令cat一样，命令tac（正好是命令cat的反序写法）也是把文件的内容显示在屏幕上，只不过是先显示最后一行，然后显示倒数第二行，最后才显示第一行。

命令——more

- 命令more也用于查看一个文件的内容，后面直接跟文件名。当文件内容太多时，按空格键可以继续看下一屏，看完所有内容后就会退出，按ctrl+B可以向上翻屏，按ctrl+F向下翻屏（同空格），如果想提前退出，按q。

命令——less

- 命令less的作用和命令more一样，后面直接跟文件名，但比more功能更多一点，按空格可以翻页，按j键可以向下移动一行，按k键可以向上移动一行。在使用more和less查看某个文件时。可以按一下/键并输入一个字符，然后回车，进行查找。

命令——head

- 命令head用于显示文件的头10行，后面直接跟文件名，如果加-n选项，则显示文件的前几行。

  # head –n10 test.txt

  # head test.txt

命令——tail

- 命令tail的作用和命令head类似，后面直接跟文件名，用于显示文件的最后10行，如果加-n选项，则显示文件的最后几行。另外命令tail的-f选项也常用，可以动态显示文件的最后10行。

Linux文件属性

- 用ls -l命令查看当前目录下的文件时，共显示了9列内容（用空格划分列）

  # ls -ld bes

  drwxr-xr-x 3 jingmq physics 2048 Mar 25 20:21 bes

  - 第1列：包含该文件的类型、所有者、所属组以及其它用户对该文件的权限。第1列共10位具体描述如下
    - d表示该文件为目录，-表示该文件为普通文件，l表示该文件为链接文件……
    - 文件类型的后面9位，每3位为一组，其中r代表可读，w代表可写，x代表可执行。前3位为所有者（user）的权限，中间3位为所属组（group）的权限，最后3位为其他非本群组用户（others）的权限。
  - 第2列：表示该文件占用的节点（inode）。
  - 第3列：表示该文件的所有者。

Linux文件属性

- 用ls -l命令查看当前目录下的文件时，共显示了9列内容（用空格划分列）

  # ls -ld bes

  drwxr-xr-x 3 jingmq physics 2048 Mar 25 20:21 bes

  - 第4列：表示该文件的所有组。
  - 第5列：表示该文件的大小。
  - 第6列、第7列和第8列：表示该文件最后一次被修改的时间（mtime），依次为月份、日期以及时间。
  - 第9列：表示文件名。

命令——chmod

- 为了方便更改文件权限，Linux使用数字代替rwx，具体规则为：r等于4，w等于2，x等于1，-等于0。例如rwxrwx---用数字代表就是770。
- 命令chmod（change mode的缩写）用于改变用户对文件/目录的读写执行权限，其格式为：chmod [-R] xyz 文件名（这里的xyz代表数字）。其中，-R选项的作用为级联更改，即不仅更改当前目录，连目录里的目录或者文件也全部更改。
- chmod还支持使用rwx的方式来设置权限，用u、g和o来代表user、group和others的属性，用a代表all（全部）。

  # chmod u=rwx/og=rx test.txt

- 还可以针对u、g、o和a，增加或者减少它们的某个权限。

  # chmod u-x test.txt

  # chmod u+x test.txt

命令——find

- find是常用的搜索工具，其格式为：find [路径] [参数]。
  - -atime +n/-n：表示访问或执行时间大于或小于n天的文件。
  - -ctime +n/-n：表示写入、更改inode属性（如更改所有者、权限或者链接）的时间大于或小于n天的文件。
  - -mtime +n/-n：表示写入时间大于或小于n天的文件，该参数用的最多。

    # find ./ -mtime -1

    stat命令可用来列出文件的atime、ctime和mtime。
  - -name filename：表示直接查找该文件名的文件（支持通配符）。

    # find ./ -name test.txt

    # find ./ -name “test*”
  - -type filetype：表示通过文件类型查找文件。

Linux磁盘管理

命令——df

- 命令df（disk filesystem的简写）用于查看已挂载磁盘的总容量、使用容量、剩余容量等，可以不加任何参数，默认以KB为单位显示。
  - -h：表示使用合适的单位显示，例如GB。

    # df –h
  - -k, -m：分别表示以KB和MB为单位显示。

    # df –k

    # df -m

命令——du

- 命令du（disk usage）用于查看某个目录或文件所占空间的大小，其格式为du [-abckmsh] [文件或者目录]。
  - -a：表示表示全部文件和目录的大小都列出来。默认显示单位“KB”。
  - -b：表示列出的值以B为单位输出。
  - -k：表示以KB为单位输出，这和默认不加任何现象的输出值是一样的。
  - -m：表示以MB为单位输出。
  - -h：表示系统自动调节单位。
  - -c：表示最后加总。
  - -s：表示只列出总和，组合用法： du –sh filename。

文本编辑工具Vim

Vim的3种常用模式——一般模式

- 当使用命令vim filename编辑文件时，默认进入该文件的一般模式。在这个模式下，可以做的操作有：上下移动光标、删除某个字符、删除某行以及复制粘贴一行或者多行。
  - gg：移动到首行。
  - G：移动到尾行。
  - x和X：x表示向后删除一个字符，X表示向前删除一个字符。
  - dd：删除/剪切光标所在的那一行。
  - yy：复制光标所在行。
  - p：从光标所在行开始，向下粘贴已经复制或者粘贴的内容。
  - P：从光标所在行开始，向上粘贴已经复制或者粘贴的内容。
  - u：还原上一步操作。
  - v：按v后移动光标会选中指定字符，然后可以实现复制、粘贴等操作。

Vim的3种常用模式——编辑模式

- 在一般模式下不可以修改某一个字符，如果要修改字符，只能进入编辑模式，当进入编辑模式时，在屏幕的尾行会显示INSERT或REPLACE的字样。从编辑模式回到一般模式，只需按Esc键即可。
  - i：在当前字符前插入。
  - I：在光标所在行的行首插入。
  - a：在当前字符后插入。
  - A：在光标所在行的行尾插入。
  - o：在当前行的下一行插入新的一行。
  - O：在当前行的上一行插入新的一行。

Vim的3种常用模式——命令模式

- 在一般模式下，输入：或者/即可进入命令模式，在该模式下，我们可以搜索某个字符或者字符串，也可以实现保存、替换、退出显示行号等操作。
  - /word：在光标之后查找一个字符串word，按n向后继续搜索。
  - ？word：在光标之前查找一个字符串word，按n向前继续搜索。
  - :n1,n2s/word1/word2/g：在n1和n2行之间查找word1并替换为word2。
  - :w：保存文本。
  - :q：退出Vim。
  - :w!：强制保存，在root用户下，即使文本只读也可以完成保存。
  - :q!：强制退出，所有改动不生效。
  - :wq：保存并退出。
  - :set nu，:set nonu：显示行号，不显示行号。

Shell基础知识

什么是Shell？

- Shell是系统跟计算机硬件交互时使用的中间介质，它只是系统的一个工具。实际上在shell和计算机硬件之间还有一层东西——系统内核。如果把计算机硬件比作一个人的躯体，那系统内核就是人的大脑。至于shell，把它比作人的五官似乎更贴切。用户直接面对的不是计算机硬件而是shell，用户把指令告诉shell，然后shell再传输给系统内核，接着内核再去支配计算机硬件去执行各种操作。
- !是与命令历史有关的特殊字符，该字符按常用的应用有以下三个：
  - ！！：连续两个！表示执行上一条指令

    # pwd

    # !!
  - !n：这里的n是数字，表示执行命令历史中的第n条指令

    # history | grep 1002　　# !1002

什么是Shell？

- !是与命令历史有关的特殊字符，该字符按常用的应用有以下三个：
  - ！字符串（字符串大于等于1）:表示执行命令历史中最近一次以pw开头的命令

    # ！pw

命令和文件名补全

- 可以通过alias把一个常用的并且很长的指令领取一个简单易记的指令，如果不想用了，还可使用unalias命令解除别名功能。

  # alias rl= ‘root -l’

- 通配符：在bash下可以使用*来匹配零个或多个字符，用？匹配一个字符。

  # ls –d /tmp/4_6/test*

  # ls –d /tmp/4_6/test?

- 输入/输出重定向：输入重定向用于改变命令的输入，输出重定向用于改变命令的输出。输出重定向更为常用，它经常用于将命令的结果输入到文件中，而不是屏幕上。输入重定向的命令是<，输出重定向的命令是>。另外，还有错误重定向2>以及追加重定向命令>>。

命令和文件名补全

- 管道符：用于将前一个指令的输出作为后一个指令的输入。

  # cat /etc/passwd | wc -l

- 作业控制：当运行进程时，可以使它暂停（按Ctrl+Z组合键），然后使用fg（foreground的缩写）命令恢复它，或是利用bg（background）命令使它到后台运行。此外也可以使它终止（按Ctrl+C组合键）。多个被暂停的任务会有编号，使用jobs命令可以看到任务，使用bg命令或者fg命令时，需要加上显示出的任务编号。

变量

- 前面提到过的环境变量PATH是shell预设的一个变量，通常shell预设的变量都是大写的。

  变量就是使用一个较简单的字符串来代替某些具有特殊意义的设定以及数据。

  ```
  # echo $PATH
  ```

- 命令env：使用env可以列出系统预设的全部系统变量。
  - HOSTNAME：表示主机名称
  - SHELL：表示当前主机的shell类型
  - HISTSIZE：表示历史记录数
  - PATH：该变量决定了shell将到哪些目录中寻找命令或程序
  - PWD：表示当前目录
  - HOME：表示当前用户的家目录
  - LOGNAME：表示当前用户的登录名。

变量

- 在用户主目录下的.bashrc文件的最后一行加入export myname=jingmq，然后运行source .bashrc就可以生效了，既可以每次登录时自动设置myname变量。
- 自定义变量的规则：
  - 设定变量的格式为a=b，其中a为变量名，b为变量的内容，等号两边不能有空格
  - 变量名只能由字母、数字以及下划线组成，而且不能以数字开头
  - 当变量内容带有特殊字符（如空格）时，需要加上单引号

```
# myname=  ‘$LOGNAME’ jingmq
# echo $myname
```

系统环境变量与个人环境变量的配置文件

- .bash_profile：该文件定义了用户的个人化路径与环境变量的文件名称。每个用户都可使用该文件输入专属于自己的shell信息，当用户登录时，该文件仅仅执行一次。
- .bashrc：该文件包含专属于自己的shell的bash信息，当登录或每次打开新的shell时，该文件会被读取。例如，你可以将用户自定义的别名或者自定义变量写到这个文件中。
- .bash_history：该文件用于记录历史命令。
- .bash_logout：当退出shell时，会执行该文件。可以将一些清理的工作放到这个文件中。

Linux shell中的特殊符号

- *：代表零个或多个任意字符

  # ls /tmp/4_6/test*

- ?：代表一个任意字符

  # ls –d /tmp/4_6/test?

- #：注释符号
- \：转义字符
- |：管道符
  - 命令cut，用来截取一个字段，格式为cut –d ‘分割字符’ [-cf] n：
    - -d：后面跟分隔符，分隔符要用单引号括起来
    - -c：后面接的是第几个字符
    - -f：后面接的是第几个区块

Linux shell中的特殊符号

- |：管道符
  - 命令cut：
    - -d：后面跟分隔符，分隔符要用单引号括起来

      # cat /etc/passwd | cut –d ‘:’ –f1 | head -5
  - 命令wc：wc命令用于统计文档的行数、字符数或词数，该命令的常用选项有-l（统计行数），-m（统计字符数）和-w（统计词数）。

    # wc /etc/passwd

    # wc -l /etc/passwd

    # wc -m /etc/passwd

    # wc –w /etc/passwd

Linux shell中的特殊符号

- $：！$表示上条命令中的最后一个变量

  # ls test.txt

  # ls !$

- ;：通常，我们都是在一行中输入一个命令，然后回车就运行了。如果想在一行中运行两个或两个以上的命令，需要在命令之间加符号;。

  # mkdir testdir; touch test1.txt; touch test2.txt; ls –d test*

- ~：符号~表示用户的家目录。

Linux shell中的特殊符号

- []：中括号内为字符组合，代表字符组合中的任意一个，可以是一个范围（1-3，a-z）

  # ls –d test*

  test1.txt test2.txt testb.txt testdir

  # ls –d test[1-3].txt

  # ls –d test1[a-z].txt

- &&和||：
  - Command1 ； Command2：不管Command1是否执行成功，都会执行Command2
  - Command1 && Command2： 只有Command1执行成功后，Command2才会执行，否则Command2不执行
  - Command1 || Command2：Command1执行成功后则Command2不执行，否则执行Command2，即Command1和Command2总由一条指令会执行。

命令——grep

- 该命令的格式为：grep [-cinvABC] ‘word’ filename：
  - -c：表示打印符合要求的行数
  - -i：表示忽略大小写
  - -n：表示输出符合要求的行及其行号

    grep –n ‘root’ /etc/passwd
  - -v：表示打印不符合要求的行

    grep –nv ‘root’ /etc/passwd
  - -A：后面跟一个数字，例如-A2表示打印符合要求的行以及下面的两行
  - -B：后面跟一个数字，例如-B2表示打印符合要求的行以及上面的两行
  - -C：后面跟一个数字，例如-C2表示打印符合要求的行以及上下各两行

命令——sed

- 替换字符或者字符串

  # sed ‘1，2s/ot/to/g’ test.txt

  # sed ‘s/[0-9]//g’ test.txt

- 直接修改文件的内容

  # sed –i ‘s/ot/to/g’ test.txt

Shell脚本

什么是Shell脚本？

- Shell脚本不能作为正式的编程语言，因为它是在Linux的shell中运行的，所以称为shell脚本。事实上shell脚本就是一些命令的集合。
- 假如我想实现这样的操作：

  （1）进入/tmp/目录；

  （2）列出当前目录中所有的文件名；

  （3）把所有当前的文件复制到/root/目录下；

  （4）删除当前目录下所有的文件。

  完成以上简单的4步需要在shell窗口中输入4次命令，按4次回车，这不算太难。但如果是输入复杂的命令，一次一次敲键盘会很麻烦。我们不妨把所有的操作都记录到一个文档中，然后去调用文档中的命令，这样一步操作就可以完成，其实这个文档就是shell脚本，只是这个shell脚本有它特殊的格式。

shell脚本的创建和执行

- 脚本文件first.sh的第1行要以#! /bin/bash开头，表示该文件使用的是bash语法。
- 执行脚本：

  # sh first.sh

  #./first.sh

  该运行方法的前提是脚本本身具有可执行权限，所以需要给脚本一个x权限，另外，使用sh命令执行一个shell脚本时，可以加-x选项来查看这个脚本的执行过程，这样有利于我们调试这个脚本：

  #sh –x first.sh

/workfs/bes/jingmq/USC_AS/first.sh

命令——date

- date命令在shell脚本中最常见的几个用法如下：
  - date +%Y：表示以四位数字格式打印年份
  - date +%y：表示以两位数字格式打印年份
  - date +%m：表示月份
  - date +%d：表示日期
  - date +%H：表示小时
  - date +%M：表示分钟
  - date +%S：表示秒
  - date +%w：表示星期，结果显示0则表示周日

    # date +” %Y-%m-%d %H-%M-%S”

    # date –d “-1 hour” + %H && date –d “-1 min” + %M

shell脚本中的变量

- 定义变量的格式为：“变量名=变量的值”，在脚本引用变量是需要加上符号$。

/workfs/bes/jingmq/USC_AS/variable.sh

- 脚本中使用了反引号，它的作用是将引号中的字符串当成shell命令执行，返回命令的执行结果，d和d1在脚本中作为变量出现。

数学运算

/workfs/bes/jingmq/USC_AS/sum.sh

- 数学计算要用[]括起来，并且前面要加符号$。

和用户交互

/workfs/bes/jingmq/USC_AS/read.sh

- read命令用于和用户交互，它把用户输入的字符串作为变量值。

脚本预设变量

/workfs/bes/jingmq/USC_AS/option.sh → # ./option.sh 1 2

shell脚本中的逻辑判断

常用的文件判断运算符如下：

-e 文件是否存在

-f 文件是否是普通文件（不是目录、设备文件、链接文件）

-s 表示文件大小不为0

-d 表示文件是否是目录

-b 表示是块设备（光驱、软盘等）

-c 表示是字符设备（键盘、声卡等）

-p 表示是管道

-h 表示是符号链接

-S 表示是否是socket

-r、-w、-x表示文件是否有可读、可写、可执行权限（指运行这个测试命令的用户）

f1 -nt f2　　f1是否比f2新（new than）

f1 -ot f2　　f1是否比f2旧（old than）

f1 -ef f2　　f1和f2是否是相同文件的硬链接

/workfs/bes/jingmq/USC_AS/if1.sh

/workfs/bes/jingmq/USC_AS/if2.sh

/workfs/bes/jingmq/USC_AS/if3.sh

/workfs/bes/jingmq/USC_AS/case.sh

对比数字使用既能使用-eq、-ne、-gt、-ge、-lt、-le，也能使用==、<、>、!=。其中-eq的意思是equal，-ne是unequal，-gt是greater than，-ge是greater than or equal to，-lt是less than，-le是less than or equal to

练习

- 创建一串目录（例如test/1/2）。
- 使用less命令查看文件/etc/passwd，搜索下共出现了几个root？按哪个键可以向上/向下逐行移动？
- 用find找出/var目录下最近一天内变更的文件，再用find找出/root目录下一个天内变更的文件。
- 用find找出/etc目录下一年内从未变更过的文件。
- Linux下的一个点“.”和两个点”..”分别代表什么？
- cd –表示什么含义？

练习

- 设计一个shell脚本，实现如下目标：
  - 输入路径，统计输入路径下的文件和文件夹个数
  - 输入路径，输出输入路径下的具体文件和文件夹（不用显示文件夹下的文件）
  - 设计良好的报错机制