

第二章：Linux命令行

姓名：张治卓 学号：517111910078

1. 当输入 `man date` 输出的手册的第一行中，有 `DATE(1)` 的出现，其中 `1` 表示的是指令的类型。常用的指令类型有哪些，分别表示哪种对应的类型？

答：
`1 - commands`
`2 - system calls`
`3 - library calls`
`4 - special files`
`5 - file formats and conversions`
`6 - games for linux`
`7 - macro packages and conventions`
`8 - system management commands`
`9 - 其他`

2. 可以在环境变量 `PATH` 中声明可执行文件的路径，如

```
export PATH=/usr/local/bin:/usr/bin:/usr/local/sbin:${HOME}/bin:${PATH}
```

如果在不同的路径下存在相同的指令，那么指令的优先级是如何体现的呢？

答：
对于多个路径下相同的指令，按照环境变量 `PATH` 中声明的顺序执行，越靠前优先级越高。

3. 使用 `ls -l` 查看一个文件时，会显示这个文件的信息，例如 `drwxr-xr-x. 3 root root 17 May 6 00:14 .config`，请解释这里所包含的信息。

答：
`drwxr-xr-x` 代表该文件为目录文件，文件所有者、同组用户、其他用户对该文件的读取权限分别为（读、写、执行）、（读、执行）、（读、执行）。
`3` 代表该目录文件第一级子目录个数。
`root root` 代表该文件所有者名为root，组名为root。
`17` 代表该文件大小为17字节。
`May 6 00:14` 代表最后修改时间为五月六日00:14。

4. 文件有3种时间戳（timestamp），分别是 `mtime`，`ctime` 和 `atime`，请说明这三种时间戳分别代表的意义以及被修改的场景。

答：
`atime(access time)`:最近访问内容的时间；
`mtime(modify time)`:最近修改内容的时间；
`ctime(change time)`:最近更改文件的时间，包括文件名、大小、内容、权限、属主、属组等。
`atime`会在文件被查看、被读取时改变；
`mtime`会在文件内容被修改时改变；
`ctime`会在文件的各项属性被更改时改变，包括文件内容被改变时引起的文件大小改变。

5. 某文件的权限为 `drw-r--r--`，用八进制表示该权限，该文件的文件属性是什么？如果需要为该目录添加执行权限，命令是什么？

答：
八进制：644
`chmod a+x file_name`

6. 如何用 `umask` 修改文件或目录的默认权限？对于特定文件，我们可以用 `chmod` 修改文件的权限，请问如何进行为一个目录下的所有内容添加该目录属主的写和执行的权限。

答：
`umask` 是设置用户创建文件或目录时默认权限的补码，这样当创建新文件或目录时，就会按照 `umask` 对应的补码设置权限，其中文件为 `(666)-umask`，目录为 `(777)-umask`。
`chmod -R u+wx file_name`

7. 什么是索引节点（index node, inode）？其意义是什么？inode能够存储哪些文件信息？

答：
`inode` 是 UNIX/Linux 操作系统中的一种数据结构，其本质是结构体，它包含了与文件系统中各个文件相关的一些重要信息，用来存放档案及目录的基本信息，包含时间、档名、使用者及群组等。
信息主要分为

- 文件类型，权限，UID，GID
- 链接数（指向这个文件名路径名称个数）
- 该文件的大小和不同的时间戳
- 指向磁盘上文件的数据块指针
- 有关文件的其他数据。

8. 说出硬链接（hard link）和符号链接（symbolic link）的差别，并举例说明。Hint：特别要注意Inode的部分。

答：
硬链接是通过索引节点(inode index)来进行链接的。在linux中，多个文件名指向同一索引点的链接就是硬链接。硬链接的作用是允许一个文件拥有多个有效路径名，这样用户就可以建立硬链接到重要文件，起到防止“误删”的功能。但是不能链接目录。
硬链接例子： `ln -s source_file softlink_file`

符号链接文件类似于windows的快捷方式,实际上是特殊文件的一种。在符号链接中，文件实际上是一个文本文件，他包含了另一个文件的位置信息。
符号链接例子： `ln existfile newfile`

也就是说，硬链接记录的inode number和源文件是一致的，不会占有空间，不能跨越系统；符号链接新建了一个inode number，是占有一定空间的文件。

9. 在使用 `find` 搜索文件时，常常使用时间选项，例如 `-ctime +30` 和 `-ctime -30` 分别代表什么样的时间信息？

答：
`-ctime +30` 代表距现在30天以前发生过更改的文件的文件名
`-ctime -30` 代表距现在30天以内发生过更改的文件的文件名

10. 对于扩展名分别为 `tar.bz2` 和 `tar.xz` 的文件，需要用什麼命令进行解包和解压缩？

答：
`tar -xvjf xxxx.tar.bz2`
`tar -xvJf xxxx.tar.xz`

11. 命令 `cat` 和 `tac`、`less` 和 `more` 有什么异同？

答： `cat` 和 `tac` 都是用来逐行显示文件内容的命令，但前一个为正序，后一个为反序。
`less` 和 `more` 都是用来分页显示文件内容的命令，但前一个只能向后翻，后一个前后翻都可以。

12. 写出查看文件 `seqs.fq` 前16行的命令以及输出结果；写出查看该文件最后20行的命令；写出查看该文件第25-40行的命令；这个文件包含多少条测序序列？

答：(1)
`head -n 16 seqs.fq`
输出结果：
[517111910078@mgt01 ~]\$ head -n 16 seqs.fq
@SRR002051.1 :8:1:325:773 length=33
AAAGAACATTAAAGCTATATTATAAGCAAAGAT
+SRR002051.1 :8:1:325:773 length=33
IIIIIIIIIIIIIIIIIIIIII'II@I\$)-
@SRR002051.2 :8:1:409:432 length=33
AAGTTATGAAATTGTAATTCCAATATCGTAAGC
+SRR002051.2 :8:1:409:432 length=33
IIIIIIIIIIIIIIIIIIIIIIIIIIII07
@SRR002051.3 :8:1:488:490 length=33
AATTCTTACCATATTAGACAAGGCACTATCTT
+SRR002051.3 :8:1:488:490 length=33
IIIIIIIIIIIIIIIIIIIIIIIIIIII&I
@SRR002051.4 :8:1:899:554 length=33
AGATTCTAATATGGTTAAGAAGCGAACTTTT
+SRR002051.4 :8:1:899:554 length=33
IIIIIIIIIIIIIIIIIIII?IIIIII<IIIIII

(2)
`tail -n 20 seqs.fq`
输出结果：
[517111910078@mgt01 ~]\$ tail -n 20 seqs.fq
@SRR002051.49998 :8:4:24:372 length=33
AGCAGCTTCTAAAGCAGCAACGGCAGTAGAAGC
+SRR002051.49998 :8:4:24:372 length=33
III:IIIIII505ID:13"%<+)/#/)%"\$/
@SRR002051.49999 :8:4:339:854 length=33
AGAAAAAAAAACAAAAAAAAAAAAACAAAAAA
+SRR002051.49999 :8:4:339:854 length=33
I+I(.'I+8'I-/8F&-&.7I&0-&,8)%,+.
@SRR002051.50000 :8:4:242:540 length=33
AGATCGGTGGTATTGGTACTGTGCCAGTCCACA
+SRR002051.50000 :8:4:242:540 length=33
IIIIIII+II-*A6*+,0"0*&(", "#%"&%"#
@SRR002051.50001 :8:4:172:536 length=33
TATATGTATAAGGTATGTGTATGTATACATATC
+SRR002051.50001 :8:4:172:536 length=33

```
IIIIIIIIIIII/III*I3I.7?I)&$4'2&
@SRR002051.50002 :8:4:908:834 length=33
TCGGAAGAGCTCGTATGCGGTCTTCTGCTTTAA
+SRR002051.50002 :8:4:908:834 length=33
IIIIII5I,I-I.E3-8#( )%C+)%$17$%&"
```

(3)

```
sed -n '25,40p' seqs.fq
```

输出结果：

```
[517111910078@mgt01 ~]$ sed -n '25,40p' seqs.fq
@SRR002051.7 :8:1:559:202 length=33
ATAAATAGTTCATTATATGTAAGTGTATATTAA
+SRR002051.7 :8:1:559:202 length=33
IIIIIIIIIIIIIIIIIIIIII2IIIIAIIF'
@SRR002051.8 :8:1:662:787 length=33
ATATAACAATATCTATATATATGTATGTACACC
+SRR002051.8 :8:1:662:787 length=33
IIIIIIIIIIIIIIIIIIIIIIII$IIII5+E
@SRR002051.9 :8:1:168:654 length=33
AAGAAGAAGAACAATATCAAGTTGTACGTT
+SRR002051.9 :8:1:168:654 length=33
IIIIIIIIIIIIIIIIIIIIII:I;7//(:2
@SRR002051.10 :8:1:941:449 length=33
ACAAATATATTAGTATTTAACAGGAAAACATAA
+SRR002051.10 :8:1:941:449 length=33
IIIIIIIIIIIIIIIIIIIIII;IICIIII
```

共50002条测序序列。