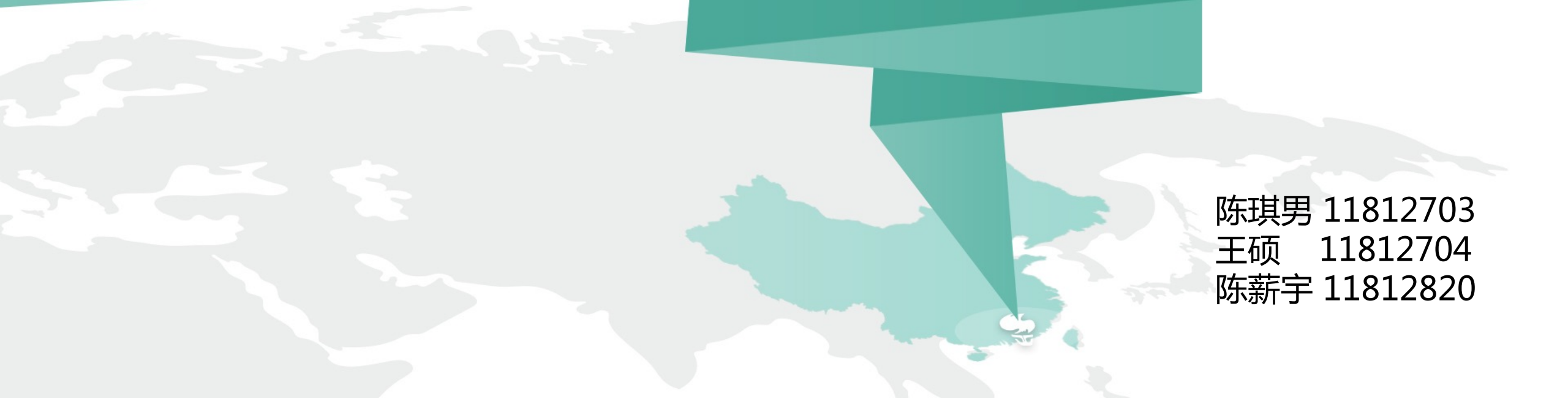


Shell Enhancement

Group 49



陈琪男 11812703
王硕 11812704
陈薪宇 11812820

Division

陈琪男

Command Auto-completion
Find

王硕

Command
Grep

陈薪宇

Pipe
Sed



Command Auto-completion Find



Command Auto-completion

1. 匹配关键字超过一种

```
[root@sylixos:/root]# ch  
ch      chmod
```

2. 匹配关键字只有一种

```
[root@sylixos:/root]# chm  → [root@sylixos:/root]# chmod 
```

Find

1. -name 通过文件名搜索文件 (支持*和?两种通配符)

```
[root@sylixos:/root]# find -name a*  
./aaa  
./aaa/ab  
./bbb/a  
./aaa/ab/a  
[root@sylixos:/root]# find aaa/ -name ?  
aaa/ab/d  
aaa/ab/c  
aaa/ab/b  
aaa/ab/a  
[root@sylixos:/root]#
```

2. -type 通过文件类型搜索文件

```
[root@sylixos:/root]# find aaa/ -type d  
aaa/ba  
aaa/ab  
[root@sylixos:/root]# find aaa/ -type f  
aaa/ab/d  
aaa/ab/c  
aaa/ab/b  
aaa/ab/a
```

Find



3. -links 通过文件硬链接数量搜索

```
[root@sylixos:/root]# find bbb/ -links 1
bbb/b
bbb/a
bbb/c
[root@sylixos:/root]# find bbb/ -links +1
[root@sylixos:/root]#
```

4. -size 通过文件大小搜索文件(支持byte, kb, mb)

```
[root@sylixos:/root]# find -size +1k
./aaa
./bbb
./aaa/ba
./aaa/ab
./bbb/c
[root@sylixos:/root]#
```

Find



5. -empty 找到所有的空文件或文件夹

```
[root@sylixos:/root]# find -empty
./aaa/ba
./bbb/a
./bbb/c
./aaa/ab/d
./aaa/ab/c
./aaa/ab/b
./aaa/ab/a
[root@sylixos:/root]#
```

6. -perm 通过文件权限搜索文件

```
[root@sylixos:/root]# find -perm 777
./aaa
./bbb/b
[root@sylixos:/root]#
```

Find

7. -mtime(-mmin) 根据最近修改日期(分钟)搜索文件

```
[root@sylixos:/root]# find bbb/ -mtime -1
bbb/b
bbb/c
[root@sylixos:/root]# find bbb/ -mtime +1
bbb/a
[root@sylixos:/root]# find bbb/ -mmin -300
bbb/b
[root@sylixos:/root]#
```


Command Help

Grep



Command Help

```
[root@sylixos:/root]# grep ?
grep          [-args] [pattern] [file name]
[root@sylixos:/root]# cd ?
cd            path
[root@sylixos:/root]# help cd
set current working path
cd path
[root@sylixos:/root]# touch ?
touch         [-amc] file name
```

Grep

Feature:

1. 按文件名查询支持通配符
2. 字符串匹配支持部分正则
3. 可以查询指定目录
4. 支持递归查询
5. 用KMP提高匹配效率

Future work:

1. 大文件读取
2. 高效字符串匹配
3. 内存管理
4. 输出对齐

```
[root@sylixos:/root]# grep -r -R in.* ?.c
.2.c      int main(){
[root@sylixos:/root]# grep -r -R in.* ?.c*
.2.c      int main(){
.1.cpp    in
[root@sylixos:/root]#
```

Grep

Feature:

1. 按文件名查询支持通配符
2. 字符串匹配支持部分正则
3. 可以查询指定目录
4. 支持递归查询
5. 用KMP提高匹配效率

```
[root@sylixos:/root]# grep -r -R in.* ./
./2.c      int main(){
./1.cpp    in
./dirl/wang.txt  in out cd !@ *?
./dirl/dirl1/neww      in in in
[root@sylixos:/root]#
```

Future work:

1. 大文件读取
2. 高效字符串匹配
3. 内存管理
4. 输出对齐

Grep

Feature:

1. 按文件名查询支持通配符
2. 字符串匹配支持部分正则
3. 可以查询指定目录
4. 支持递归查询
5. 用KMP提高匹配效率

Future work:

1. 大文件读取
2. 高效字符串匹配
3. 内存管理
4. 输出对齐

```
[root@sylixos:/root]# grep -r in ./
./1.cpp in
./2.c #include <stdio.h>
./2.c int main(){
./2.c printf("hello WS\n");
./dirl/wang.txt in out cd !@ *?
./dirl/dirl1/neww in in in
[root@sylixos:/root]# grep -r in /root/
/root/1.cpp in
/root/2.c #include <stdio.h>
/root/2.c int main(){
/root/2.c printf("hello WS\n");
/root/dirl/wang.txt in out cd !@ *?
/root/dirl/dirl1/neww in in in
[root@sylixos:/root]#
```

Grep



Feature:

1. 按文件名查询支持通配符
2. 字符串匹配支持部分正则
3. 可以查询指定目录
4. 支持递归查询
5. 用KMP提高匹配效率

Future work:

1. 大文件读取
2. 高效字符串匹配
3. 内存管理
4. 输出对齐

Pipe Sed



Pipe



`pwd | ls`

```
[root@sylixos:/root]# ls
d      c      a      pipe  os
b
```

```
[root@sylixos:/root]# pwd | ls
d      c      a      pipe  os
b
```

`cat c | cat`

```
[root@sylixos:/root]# cat c
d
[root@sylixos:/root]# cat d
d 1 the first line
d 2 the second line
[root@sylixos:/root]# cat c | cat
d 1 the first line
d 2 the second line
```


Sed



1. 检测sed命令：

```
[root@sylixos:/root]# cat a
first line
second line
third line
4th line
```

```
[root@sylixos:/root]# sed a
first line
second line
third line
4th line
```

2. sed s 替换脚本命令

此命令的基本格式为：

`sed [address]s/pattern/replacement/flags file`

```
[root@sylixos:/root]# sed 's/i/l/' a
first line
second line
third line
4th line
```

```
[root@sylixos:/root]# sed 's/i/l/2' a
first llne
second line
third llne
4th line
```

```
[root@sylixos:/root]# sed 's/i/l/g' a
flrst llne
second llne
thlrd llne
4th llne
```

```
[root@sylixos:/root]# sed 's/f/l/p' a
1first line
```

Sed

3. sed d

d 命令表示删除满足条件的行，
此命令的基本格式为
[address]d

```
[root@sylixos:/root]# sed 'd' a
[root@sylixos:/root]# sed '2,3d' a
first line
4th line
[root@sylixos:/root]# sed '2d' a
first line
third line
4th line
[root@sylixos:/root]# sed '3,$d' a
first line
second line
```

4. sed p 打印脚本命令

p 命令表示搜索符号条件的行，并输出该行的内容，
此命令的基本格式为：
[address]p

```
[root@sylixos:/root]# sed '2,3p' a
first line
second line
second line
third line
third line
4th line
[root@sylixos:/root]# sed '2p' a
first line
second line
second line
third line
4th line
```

5. sed a 和 i 脚本命令

a 命令

表示在指定行的后面附加一行，i 命令表示在指定行的前面插入一行，它们的基本格式完全相同，如下所示：

[address]a (或 i) \新文本内容

```
[root@sylixos:/root]# sed '3a\addedline' a
first line
second line
third line
addedline
4th line
[root@sylixos:/root]# sed '3i\addedline' a
first line
second line
addedline
third line
4th line
```

6. sed c 替换脚本命令

c 命令

表示将指定行中的所有内容，替换成该选项后面的字符串。

该命令的基本格式为：
[address]c\用于替换的新文本

```
[root@sylixos:/root]# sed '3c\addedline' a
first line
second line
addedline
4th line
```

Sed



SUSTech

Southern University
of Science and
Technology

7. sed y 转换脚本命令

y 转换命令是唯一可以处理单个字符的 sed 脚本命令，其基本格式如下：
[address]y/inchars/outchars/

```
[root@sylixos:/root]# sed 'y/fir/123/' a
123st 12ne
second 12ne
th23d 12ne
4th 12ne
```

8. sed w 脚本命令

w 命令用来将文本中指定行的内容写入文件中，此命令的基本格式如下：

[address]w filename

```
[root@sylixos:/root]# sed '1,2w b' a
first line
second line
[root@sylixos:/root]# sed b
first line
second line
[root@sylixos:/root]# sed '1w b' a
first line
[root@sylixos:/root]# sed b
first line
```

9. sed r 脚本命令

r 命令用于将一个独立文件的数据插入到当前数据流的指定位置，该命令的基本格式为：
[address]r filename

```
[root@sylixos:/root]# sed d
d 1 the first line
d 2 the second line
[root@sylixos:/root]# sed '3r d' a
first line
second line
third line
d 1 the first line
d 2 the second line
4th line
```

10. sed q 退出脚本命令
q 命令的作用是使 sed 命令在第一次匹配任务结束后，退出 sed 程序，不再进行对后续数据的处理。

```
[root@sylixos:/root]# sed '2q' a
first line
second line
```

Summary



Thank You