

-----oOo-----

Phần 1: Thực hành

Sed trong Linux là một trình soạn thảo luồng (*Stream-oriented Editor*), dùng để thao tác trực tiếp với văn bản như thay thế, xóa dòng, in ra một số dòng và một số thao tác khác. Sed có một vài lệnh, nhưng phần lớn mọi người chỉ sử dụng lệnh substitute: s. Lệnh *substitute* có tác dụng thay đổi toàn bộ nội dung của một biểu thức thành một biểu thức mới. Ví dụ như thay đổi "day" thành "night":

Ví dụ:

`$ echo day | sed s/day/night/` → Thay thế *day* bằng *night*

Kết quả: *night*

Hoặc: `sed s/day/night/ <old >new` → Thay thế *day* trong tệp tin *old* bằng *night* trong tệp tin *new*

Ví dụ:

`$ echo "day Thursday" | sed 's/day/night/g'` → Thêm "g" để thay thế tất cả *day* bằng *night*

Kết quả: *night Thursnight*

Cú pháp:

`sed [options] 'lệnh' FILE`

Hoặc: `sed [options] -f script FILE`

Trong đó

- options:
 - `-e` : đọc một lệnh sed khác từ cửa sổ dòng lệnh
 - `-f` : đọc một lệnh sed khác từ file script
 - `-n` : ngăn chặn đầu ra của lệnh sed, khi sử dụng `-n` bạn phải sử dụng câu lệnh `p` để in ra màn hình. Ví dụ: `sed -n '/^/*..$/p' filename`
- Một lệnh có dạng: `[địa chỉ],[địa chỉ] [!]lệnh[tham số]`

- Tham số:
 - d: xóa
 - p: in (sử dụng với option -n)
 - Script là file chứa lệnh.
 - Địa chỉ có thể là: Số dòng, ví dụ: 3 (dòng số 3), hoặc các biểu thức chính quy
 - Ví dụ: *sed '3,8 d' filename*
 - Ví dụ: */BEGIN/*
 - Nếu có kí hiệu ! đặt sau địa chỉ có nghĩa là trừ phần địa chỉ ra
- Ví dụ:

```
$cat sedtest.txt
This is line #1
It is line #2
That is line #3
While, this is line #4
It's line #5
I am line #6
Myself line #7
It's me, line #8
Hello, I am line #9
Last line, line #10
```

Thực thi:

```
$ sed '/It/ d' sedtest.txt
$ sed '/It/ !d' sedtest.txt
```

1. Lệnh *substitute*:

Cú pháp:

sed 's/mau/thaythe/[tùy chọn]' [<dauvao] [>daura]

Dùng để thay thế *mau* thành *thaythe*. Nếu không có tùy chọn *g*, chỉ thay thế một lần cho 1 dòng. Có thể sử dụng *_* hoặc *:* để thay thế cho */*.

- Ví dụ: *s_mau_thaythe_g* hoặc *s:mau:thaythe:g*

2. Chuyển hướng đầu ra của lệnh *sed*

Có thể chuyển hướng đầu ra của lệnh *sed* sử dụng *>*

Ví dụ:

- Thay thế Linux thành UNIX (system v) trong file example.txt và ghi ra file file.out: *sed 's/Linux/UNIX(system v)/' example.txt > file.out*
- Xóa dòng trống: *sed '/^\$/d' example.txt*

3. Ký tự slash (/)

Ký tự (/) sau lệnh *s* có vai trò như một ký tự phân cách. Không nhất thiết phải sử dụng slash(/) mà thay vào đó có thể sử dụng bất cứ ký tự đặc biệt nào mà bạn muốn.

Thay vì sử dụng ký tự (/) , có thể sử dụng ký tự gạch dưới thay vì ký tự (_) như sau:

```
sed 's_/usr/local/bin_/common/bin_' <old >new
```

Một số người thích dùng dấu hai chấm (:)

```
sed 's:/usr/local/bin:/common/bin:' <old >new
```

Một số khác thì sử dụng ký tự "|".

```
sed 's|usr/local/bin|common/bin|' <old >new
```

4. Viết sed scripts

Lệnh sed cũng có thể nhóm lại với nhau thành một tệp tin văn bản, được gọi là sed script.

Ví dụ:

```
cat > example1.txt
```

```
Pen      5    20.00
Pencil  10     2.00
Rubber   3     3.50
Cock     2    45.50
```

Có thể viết:

```
cat > sed1.txt
```

```
1i\
Price of all items changes from 1st-April-2001
/Pen/s/20.00/19.5/
/Pencil/s/2.00/2.60/
/Rubber/s/3.50/4.25/
/Cock/s/45.50/51.00/
```

- Để chèn một dòng text vào một dòng trong tệp tin, ta dùng cú pháp:

```
[line-address]i\
text
```

- Để nối dòng **a** và để thay đổi dòng **c**

Chạy script này bằng cách dùng lệnh: *sed -f sed1.txt example1.txt*

5. Ký tự &

Ký tự & để đại diện cho nội dung cho các string khớp với mẫu để tìm kiếm.

Ví dụ:

```
Sed 's/abc/(abc)/' <old >new
```

Nhưng cách này không hiệu quả khi bạn không biết chính xác string bạn tìm là gì mà chỉ biết mẫu của nó. Giải pháp chính là sử dụng ký tự đặc biệt "&". Ký tự này đại diện cho mẫu được tìm thấy.

Ví dụ:

```
echo 'abcd1234' | sed 's/[a-z]*/(&)/'
```

Bạn có thể sử dụng số lượng "&" tùy ý trong string thay thế. Bạn có thể gấp đôi mẫu, e.g. những chữ số đầu tiên của dòng.

Ví dụ:

```
echo "123 abc" | sed 's/[0-9]*/& &/'
```

Kết quả: 123 123 abc

6. Sử dụng \1 để giữ lại một phần của mẫu

"\1" ghi nhớ mẫu đầu tiên là sed tìm kiếm được, "\2" ghi nhớ mẫu thứ 2 sed tìm kiếm được. Sed nhớ được lên tới 9 mẫu khớp với mẫu sử dụng để tìm kiếm.

Nếu bạn đã giữ được từ đầu tiên của 1 line và xóa đi phần còn lại, hãy đánh dấu phần quan trọng bằng dấu ngoặc tròn như sau:

```
sed 's/([a-z]*\).*\1/'
```

Biểu thức sử dụng để tìm cố gắng so khớp nhiều mẫu nhất có thể: "[a-z]*" khớp với null string và tất cả những ký tự thường đầu tiên mà sed tìm thấy. Vì mẫu đầu tiên khớp với mẫu rồi nên tất cả những thứ còn lại sẽ là khớp với mẫu đối lập với mẫu đó. Do đó nếu bạn gõ lệnh

```
echo abcd123abc | sed 's/([a-z]*\).*\1/'
```

Sẽ trả về output "abcd" và xóa đi tất cả những thứ không thỏa mãn điều kiện là string đầu tiên khớp với mẫu đưa ra để tìm kiếm.

Nếu bạn muốn trao đổi vị trí của 2 string khớp với mẫu:

```
echo abc def | sed 's/([a-z]*)\ ([a-z]*)/2 \1/'
```

Chú ý rằng có dấu cách giữa 2 mẫu được ghi nhớ. Điều này được sử dụng để chắc chắn rằng 2 từ đã được tìm thấy. Tuy nhiên, điều này không có tác dụng khi một từ đơn được tìm thấy hay tất cả các lines đều không có ký tự nào. Bạn có thể muốn viết rõ là mẫu bắt buộc bắt đầu bởi 1 ký tự bằng cách sử dụng:

```
sed 's/([a-z][a-z]*)\ ([a-z][a-z]*)/2 \1/'
```

Nếu bạn muốn phát hiện các từ trùng nhau, bạn có thể sử dụng

```
sed -n '/([a-z][a-z]*)\ /p'
```

hay cách sử dụng các biểu thức tìm kiếm dạng mở rộng

```
sed -n '/([a-z]+)\ /p'
```

Trong trường hợp này, sed được sử dụng với chức năng như một filter, với công dụng là hiển thị ra các dòng có các từ trùng nhau.

Lượng biến mà sed cho phép ghi nhớ lên tới 9 biến: "\1" tới "\9". Nếu bạn muốn đảo ngược 3 ký tự trong 1 dòng, bạn có thể sử dụng:

```
echo a b c | sed -r 's/([a-z]+) ([a-z]+) ([a-z]+)/3\2\1/'
```

7. Một số ví dụ về thao tác với sed

- 10d Xóa dòng thứ mười.
- /^\$/d Xóa các dòng trống.
- 1,/^\$/d Xóa từ dòng đầu tiên cho đến khi gặp dòng trống đầu tiên.
- /NAM/p Chỉ in ra dòng chứa "NAM" (phải dùng cùng với tùy chọn -n).
- s/ *\$// Xóa tất cả các khoảng trắng ở cuối mỗi dòng.
- /NAM/d Xóa tất cả các dòng chứa "NAM".
- s/NAM//g Chỉ thay thế chữ "NAM", các phần còn lại giữ nguyên.

Phần 2: Bài tập

Bài 1. Cho tệp tin friends.txt có nội dung như sau:

```

-----
Name      DOB      Hobby      Phone      #
-----
Ha Lan    5/12/93    Food, Music  98220-5678  5/22, Street 4, A'bad, MH, INDIA.

Vu Ha    15/6/72    Computers, Book Reading  98220-333322, Thanh xuan, VietNam.

M.M. Kale  2/1/91    Food, Drinks, Lifestyle  98220-6823  6/21, A'bad, MH, USA.

Ngoc Le   9/10/90    Colletion of Old coins  98220-6877  Flat No.9, Sing.

N.K. Kulkarni 1/2/94    Computer Games  98220-988888, Tu Liem, HN, VietNam.
-----

```

Yêu cầu:

1. Thay thế tất cả các chuỗi “A’bad” bằng chuỗi “Aurangabad”
2. Thay thế tất cả các chuỗi “MH” bằng chuỗi “Maharastra”
3. Thay thế tất cả các dòng trống bằng “=====”
4. Tìm kiếm và in ra thông tin của những người có họ là “Ha”
5. Tìm kiếm và in ra số đầu tiên trong tệp tin có nhiều hơn 3 số 8 (đứng cạnh nhau hoặc không đứng cạnh nhau)
6. In ra 4 dòng đầu tiên trong file friends.txt
7. Thêm vào thông tin của hai người bạn của bạn vào 2 dòng cuối trong file friends.txt
8. Thay đổi dòng thứ 2 bởi thông tin người bạn thứ 3 của bạn

Phần 3: Liên lạc

STT	Họ và tên	Email	ĐT
1	Hà Mỹ Linh	halinh.hus@gmail.com	
2	Trần Thị Hương	tranthihuong.hus@gmail.com	