***Лабораторна робота №15***

***«Робота з потоковим редактором sed»***

***Теоретичні відомості***

Потоковий редактор **Stream EDitor** використовується для виконання базових перетворень в тексті, який зчитується з файлу або з конвеєра. Результат відправляється на стандартне виведення. Синтаксис команди ***sed*** не дозволяє вказати вихідний файл, але результат можна зберегти в файлі за допомогою перенаправлення виведення. Редактор не дозволяє змінити вихідний вхідний потік.

Редактор ***sed*** відрізняється від інших редакторів тим, що з його допомогою можна фільтрувати текст, що подається з конвеєра. Вам не потрібно управляти редактором під час його роботи, тому редактор ***sed*** іноді називається редактором потокової обробки. Ця особливість дозволяє вказувати команди редагування всередині скриптів, що значно полегшує виконання завдань, що повторюються. Коли в великій кількості файлів потрібно виконати текстову заміну, ***sed*** стає гарною підмогою.

***Команди sed***

За допомогою програми ***sed*** можна виконувати текстові заміни і видалення за зразком, а також можна використовувати точно такі ж регулярні вирази, що і з командою ***grep***.

Команди редагування аналогічні тим, що використовуються в редакторі ***vi***.

Табл. 1. Команди редагування ***sed***

|  |  |
| --- | --- |
| **Команда** | **Результат** |
| ***a\*** | Додавання тексту під поточним рядком |
| ***c\*** | Заміна тексту в поточному рядку новим текстом |
| ***d*** | Видалення тексту |
| ***i\*** | Вставка тексту над поточним рядком |
| ***p*** | Видача тексту |
| ***r*** | Читання файлу |
| ***s*** | Пошук і заміна тексту |
| ***w*** | Запис в файл |

Крім команд редагування, можна в ***sed*** вказувати додаткові параметри. Їх огляд наведено в таблиці нижче.

Табл. 2. Параметри редактора ***sed***

|  |  |
| --- | --- |
| **Параметр** | **Дія** |
| ***-e SCRIPT*** | До набору команд, які повинні бути запущені при обробці вхідного потоку, додаються команди, які вказуються в параметрі ***SCRIPT*** |
| ***-f*** | До набору команд, які повинні бути запущені при обробці вхідного потоку, додаються команди, що знаходяться в файлі ***SCRIPT-FILE*** |
| ***-n*** | Режим мовчання (у стандартний потік виведення нічого не видається) |
| ***-V*** | Видається інформація про версії і відбувається вихід з програми |

У документації по ***sed*** міститься більше інформації, тут ми перерахували тільки найбільш часто використовувані команди і параметри.

***Інтерактивне редагування***

***Видача рядків, що містять шаблон***

Подібне, звичайно, можна зробити за допомогою команди ***grep***, але з її допомогою ми не зможемо виконати дію "Знайти і замінити". З цього ми тільки почнемо.

Нижче наведено наш текстовий файл ***example***:

***$ cat -n example***

***1 This is the first line of an example text.***

***2 It is a text with erors.***

***3 Lots of erors.***

***4 So much erors, all these erors are making me sick.***

***5 This is a line not containing any errors.***

***6 This is the last line.***

Ми хочемо за допомогою редактора ***sed*** знайти всі рядки, що містять наш шаблон пошуку, в даному випадку "***erors***". Щоб знайти результат, ми використовуємо ***p***:

***$ sed '/erors/p' example***

***This is the first line of an example text.***

***It is a text with erors.***

***It is a text with erors.***

***Lots of erors.***

***Lots of erors.***

***So much erors, all these erors are making me sick.***

***So much erors, all these erors are making me sick.***

***This is a line not containing any errors.***

***This is the last line.***

Як можна помітити, ***sed*** видає весь файл, а рядки, що містять шуканий рядок, виводяться двічі. Це не те, що нам потрібно. Щоб виводити тільки ті рядки, які відповідають нашому шаблоном, потрібно використати параметр ***-n***:

***$ sed -n '/erors/p' example***

***It is a text with erors.***

***Lots of erors.***

***So much erors, all these erors are making me sick.***

***Видалення рядків, що містять шаблон***

Ми використовуємо той же самий текстовий файл ***example***. Тепер ми хочемо бачити лише рядки, в яких немає шуканого рядка:

***$ sed '/erors/d' example***

***This is the first line of an example text.***

***This is a line not containing any errors.***

***This is the last line.***

Команда ***d*** вказує, що знайдені рядки не показуються.

Рядки, що починаються заданим шаблоном і закінчуються другим шаблоном, показуються в такий спосіб:

***$ sed -n '/^This.\*errors.$/p' example***

***This is a line not containing any errors.***

Відзначимо, що, насправді, для останньої точки потрібно вказати, що це не спецсимвол і не шукати з нею збіги. У нашому прикладі відповідність шукається для будь-якого останнього символу, в тому числі і точки.

***Діапазони рядків***

На цей раз ми хочемо отримати рядки, в яких є помилки. У прикладі такими є рядки з ***2*** по ***4***. Позначимо цей діапазон адрес і команду ***d***:

***$ sed -n '2,4d' example***

***This is the first line of an example text.***

***This is a line not containing any errors.***

***This is the last line.***

Щоб видати тільки початкові рядки з файлу і виключити рядки, починаючи з певного рядка і до кінця файлу, потрібно використати команду, подібну до наступної:

***$ sed '3,$d' example***

***This is the first line of an example text.***

***It is a text with erors.***

В результаті будуть виведені тільки перші два рядки файлу ***example***.

Наступна команда видає рядки, починаючи з першого, в якій виявлений шаблон "***a text***", і до того рядка, в якій виявлений шаблон "***a line***":

***$ sed -n '/a text/,/This/p' example***

***It is a text with erors.***

***Lots of erors.***

***So much erors, all these erors are making me sick.***

***This is a line not containing any errors.***

***Пошук і заміна за допомогою sed***

Тепер ми будемо в файлі ***example*** шукати помилки і замінювати їх, а не тільки шукати рядки, що містять (або не містять) шуканий рядок.

***$ sed 's/erors/errors/' example***

***This is the first line of an example text.***

***It is a text with errors.***

***Lots of errors.***

***So much errors, all these erors are making me sick.***

***This is a line not containing any errors.***

***This is the last line.***

Як бачимо, це не зовсім те, що нам потрібно: в рядку 4 заміна була виконана тільки для першого входження, і там ще залишилася рядок '***eror***'. Потрібно використати команду ***g***, щоб вказати ***sed***, що він повинен перевіряти весь рядок, а не зупинятися після того, як виявить перше входження відповідного рядка:

***$ sed 's/erors/errors/g' example***

***This is the first line of an example text.***

***It is a text with errors.***

***Lots of errors.***

***So much errors, all these errors are making me sick.***

***This is a line not containing any errors.***

***This is the last line.***

Вставка рядка на початку кожного рядка:

***$ sed 's/^/> /' example***

***> This is the first line of an example text.***

***> It is a text with erors.***

***> Lots of erors.***

***> So much erors, all these erors are making me sick.***

***> This is a line not containing any errors.***

***> This is the last line.***

Вставка рядка у кінці кожного рядка:

***$ sed 's/$/EOL/' example***

***This is the first line of an example text.EOL***

***It is a text with erors.EOL***

***Lots of erors.EOL***

***So much erors, all these erors are making me sick.EOL***

***This is a line not containing any errors.EOL***

***This is the last line.EOL***

Можна задати відразу кілька команд пошуку і заміни, випереджаючи кожну параметром ***-e***:

***$ sed -e 's/erors/errors/g' -e 's/last/final/g' example***

***This is the first line of an example text.***

***It is a text with errors.***

***Lots of errors.***

***So much errors, all these errors are making me sick.***

***This is a line not containing any errors.***

***This is the final line.***

Треба мати на увазі, що за замовчуванням ***sed*** видає свої результати в стандартний потік виведення, тобто, як правило, у термінальне вікно. Якщо ми хочемо зберегти результат у файлі, то потрібно виконати перенаправлення.

***Неінтерактивне редагування***

***Читання команд sed з файлу***

Кілька команд ***sed*** можна об'єднати в файл і виконати файл з використанням параметра ***-f***. Коли створюється такий файл, переконайтеся в наступному:

* в кінці рядків не повинно бути пробілів;
* не повинні використовуватися лапки;
* коли вводиться текст для додавання або заміни, перед усіма символами кінця рядків, крім останнього, повинен додаватися зворотний слеш.

***Запис вихідних файлів***

Запис результату, що виводиться, виконується за допомогою оператора перенаправлення ***>***. Нижче наведено приклад скрипта, який використовується для створення дуже простого файлу HTML зі звичайного текстового файлу.

***$ cat script.sed***

***<html>***

***<head><title>sed generated html</title></head>\***

***<body bgcolor="#ffffff">\***

***<pre>***

***$a\***

***</pre>\***

***</body>\***

***</html>***

***$ cat txt2html.sh***

***#!/bin/bash***

***echo "converting $1..."***

***SCRIPT="/home/sandy/scripts/script.sed"***

***NAME="$1"***

***TEMPFILE="/var/tmp/sed.$PID.tmp"***

***sed "s/\n/^M/" $1 | sed -f $SCRIPT | sed "s/^M/\n/" > $TEMPFILE***

***mv $TEMPFILE $NAME***

***echo "done."***

У змінній ***$1*** зберігається перший аргумент даної команди, в даному випадку це ім'я файлу, використовуваного у перетворенні:

***$ cat test***

***line1***

***line2***

***line3***

Виклик сценарію:

***$ txt2html.sh test***

***converting test...***

***done.***

***$ cat test***

***<html>***

***<head><title>sed generated html</title></head>***

***<body bgcolor="#ffffff">***

***<pre>***

***line1***

***line2***

***line3***

***</pre>***

***</body>***

***</html>***

Потужні редактори, що підтримують підсвічування синтаксису, можуть розпізнавати синтаксис ***sed***. Це може бути великою підмогою для тих, хто забуває вказувати зворотний слеш і т.д.

***Завдання***

1. Створіть список файлів, що є в директорії ***/usr/bin***, в яких в якості другого символу використовується буква "***а***". Помістіть результат в тимчасовий файл.
2. Видаліть перші три рядки кожного тимчасового файлу.
3. Видайте в стандартне виведення тільки ті рядки, в яких є шаблон "***an***".
4. Створіть файл з командами ***sed***, які виконують два попередні завдання. Додайте в цей файл ще одну команду, яка додає рядок, наприклад, ***"\*\*\* This might have something to do with man and man pages \*\*\*"*** перед кожним входженням рядка ***"man"***. Перевірте результати.
5. Створіть довгий список файлів, використавши в якості входу кореневий каталог ***"/"***. Створіть файл з командами ***sed***, які перевіряють, чи є файл символічним посиланням або звичайним файлом. Якщо файл є символічним посиланням, поставте перед ним рядок ***"--This is a symlink--"***. Якщо файл є звичайним файлом, додайте в той самий рядок, наприклад, коментар ***"<--- this is a plain file"*** (.
6. Створіть скрипт, який показує в файлі рядки, що містять проміжки в кінці рядка. В цьому сценарії повинен використовуватися ***sed*** і видаватися зрозумілий користувачам результат.

***Контрольні питання***

1. Який формат має команда ***sed***?
2. Яке призначення команди ‘***s***’ у редакторі ***sed***?
3. Як прочитати команди редактору ***sed*** з файлу?
4. Як замінити символ за допомогою потокового редактора ***sed***?
5. Для чого призначений ключ ***–e*** у команді ***sed***?
6. Як додати текст під поточним рядком за допомогою редактора ***sed***?
7. Як додати текст над поточним рядком за допомогою редактора ***sed***?
8. Як у редакторі ***sed*** вказати діапазон рядків?