

Федеральное государственное автономное образовательное учреждение высшего образования

«Санкт-Петербургский национальный исследовательский университет
информационных технологий, механики и оптики»

Лабораторная работа №5

по дисциплине “Системное программное обеспечение”

Выполнила: Калугина Марина

Группа: Р3202

г. Санкт-Петербург

2019 г.

Задание 1

1. Замените имя Jon на Jonathan

```
sed 's/Jon/Jonathan/' datebook
```

операция s заменяет первое встреченное соответствие шаблону

2. Удалите первые три строки

```
sed 1,3d datebook
```

операция d удаляет заданный диапазон строк

3. Выведите строки с 5-ой по 10-ю

```
sed -n 5,10p datebook
```

операция p печатает указанный диапазон строк. Ключ -n позволяет выводить только те строки, которые совпадают с шаблоном. Без использования ключа -n оператор p дублирует строки в указанном диапазоне.

4. Удалите строки, содержащие Lane

```
sed '/Lane/d' datebook
```

в данном случае ключ d удалит те строки, в которых был найден шаблон.

5. Вывести все строки с с днем рождения в ноябре или декабре

```
sed -n '/[0-9]\{5\}: [1][1,2]\{0-9\}\{1,2\}\{0-9\}\{2\}: [0-9]\{5\}$/p' datebook  
12345: 12 / 15 / 98 : 12345
```

ключ -n и оператор p описываются аналогично пункту 3. Конструкция `[0-9]\{5\}` означает, что ровно 5 цифр, далее идет двоеточие, за ним 2 числа: первое 1, второе - 1 или 2, далее идет число в диапазоне от 0 до 99, и после двоеточие и 5 цифр. Таким образом создается шаблон для поиска даты, месяц в которой имеет номер 11 или 12.

6. Добавить три символа * в конец строк, начинающихся с Fred

```
sed 's/^Fred.*$/&*\*/' datebook
```

в шаблоне ^ имеет значение начала строки, \$ - конца, оператор & добавляет к шаблону написанное после значение.

7. Замените строку, содержащую Jose на JOSE HAS RETIRED

```
sed 's/^.*Jose.*$/JOSE HAS RETIRED/' datebook
```

шаблон - строка от начала до конца, в которой содержится Jose. Оператор s заменяет найденную строку, на JOSE HAS RETIRED

8. Замените дату рождения Roreye на 11/14/46. При этом подразумевается, что вы не знаете значение даты, хранящейся в файле. Составьте и используйте регулярное выражение для ее поиска

```
sed '/^Roreye/s/[0-9]\{1,2\}[0-9]\{1,2\}[0-9][0-9]/11\14\46/' datebook
      3   /   19   /   35 =>/ 11 /14 /46
```

сначала производится поиск строки, начинающейся с Roreye, затем в ней оператор s заменяет по шаблону: две цифры, /, две цифры, /, две цифры на 11/14/46

9. Удалите все пустые строки

```
sed '/^$/d' datebook
```

в шаблоне ^ - начало строки, \$ - конец строки, оператор d удаляет строки, соответствующие шаблону

10. Напишите программу на языке редактора sed, которая:

1. вставляет перед первой строкой заголовок TITLE OF FILE
2. удаляет последнее поле, значение которого кратно 500
3. меняет местами имя и фамилию
4. добавляет к концу каждой строки фразу THE END

```
sed -e '1s/^/TITLE OF FILE\n/'
    -e 's:[0-9]\{2,3\}[0,5]00$/g'
    -e 's/^([A-Za-z]*) \([A-Za-z]*\):/2 \1:/g'
    -e '$s/$\nTHE END/'
datebook
```

оператор -e позволяет выполнять операции последовательно. В первой строке оператор 1s заменяет только первый найденный шаблон. Тут начало строки заменяется на TITLE OF FILE с переносом строки после.

Во второй строке: число кратно 500 в том случае, если его последние 3 цифры 000 или 500. Поэтому находим такое число перед концом строки, и если нашли его, то заменяем на пустую последовательность символов. Оператор g позволяет заменить все строки, удовлетворяющие условию.

В третьей строке: в запоминающие скобки 1 сохраняем первое слово до пробела. Во вторые - слово от пробела до двоеточия. С помощью оператора s меняем местами 1 и 2. С помощью оператора g выполняем это для каждой строки.

четвертая строка аналогична первой, только используем \$s для работы с последним вхождением.

Задание 2

1. sed '/north/p' datafile

выводит строки, содержащие 'north', так как нет никаких дополнительных ключей, то оператор p выводит текст целиком, дублируя строки, содержащие шаблон.

2. sed -n '/north/p' datafile

ключ -n подавляет автоматический вывод буфера после обработки, таким образом выводит только строки, содержащие образец.

3. `sed '3d' datafile`

оператор d удаляет указанную строку, в данном случае удаляет третью строку

4. `sed '3,$d' datafile`

удаляет строки с третьей до последней. (Если бы после запятой было число, то он указывал на конец интервала, а в данном случае конец интервала - конец файла)

5. `sed '$d' datafile`

удаляет последнюю строку

6. `sed '/north/d' datafile`

удаляет строку, содержащую north

7. `sed 's/west/north/g' datafile`

оператор s заменяет west на north, оператор g соответствует глобальной замене, а не первому вхождению.

8. `sed -n 's/^west/north/p' datafile`

-n выводит строки, соответствующие шаблону: в начале строки west, где этот шаблон заменяется на north.

9. `sed 's/[0-9][0-9]$/&.5/' datafile`

& - позволяет дописать к шаблону значение, указанное после него, таким образом, команда допишет .5 к строкам, заканчивающимся на 2 цифры.

10. `sed -n 's/Hemenway/Jones/gp' datafile`

ключ -n выведет только те строки, которые удовлетворяют шаблону, все (оператор g) 'Hemenway' заменятся на 'Jones' и будут выведены. Так как в файле datafile нет подстроки Hemenway, то выведено ничего не будет.

11. `sed -n 's/(Stag\)got/1ianne/p' datafile`

ключ -n выведет только те строки, которые удовлетворяют шаблону, круглые скобки сохраняют Stag в первый параметр, и заменяет Staggot на Stagianne (\1 - тот самый параметр, который был сохранен круглыми скобками). В результате работы команда ничего не выведет, потому что в файле datafile нет подстроки Staggot.

12. `sed 's/14#88#g' datafile`

кроме / разделять аргументы можно и #. Т.е. эта команда аналогична `sed 's/14/88/g' datafile` и она заменяет все 14 на 88.

13. `sed -n '/west/,/east/p' datafile`

ключ -n выведет только те строки, которые удовлетворяют шаблону, вывод строк в диапазоне от west до east (как если бы было написано 1,3/p, вывелись бы строки от первой до третьей).

14. sed -n '5,/^\northeast/p' datafile

ключ -n выведет только те строки, которые удовлетворяют шаблону, вывод строк с 5 до строки, удовлетворяющей шаблону (строка, начинающаяся с northeast)

15. sed '/west/,/east/s/\$/**WAKA**/' datafile

в диапазоне от строки, содержащей west до строки содержащей east символ конца строки заменяется на **WAKA**, т.е. добавляется в конец строки.

16. sed -e '1,3d' -e 's/Hemenway/Jones/' datafile

ключ -e добавляет скрипт для обработки, таким образом сначала удаляются первые 3 строки, потом у результата заменяется Hemenway на Jones.

17. sed '/Suan/r newfile' datafile

оператор r file добавляет к выводу текст, прочитанный из file. После всех строчек, с подстрокой Suan будет добавлен текст из newfile

18. sed -n '/north/w newfile' datafile

оператор w newfile записывает в newfile буфер. В файл newfile запишутся все строки, содержащие north.

19. sed '/^north /a\

--->THE NORTH SALES DISTRICT HAS MOVED<---' datafile

оператор a позволяет вывести в поток вывода после буфера указанную фразу. \ - позволяет делать многострочные фразы. В результате выполнения команды после строк со словом north в начале будет выведено --->THE NORTH SALES DISTRICT HAS MOVED<---

20. sed '/eastern/i\

NEW ENGLAND REGION\

-----' datafile

оператор i выводит в поток вывода перед буфером указанную фразу. \ служит для составления многострочных фраз. В результате работы данной команды перед строками с подстрокой eastern будет выведено NEW ENGLAND REGION...

21. sed '/eastern/c\

THE EASTERN REGION HAS BEEN TEMPORARILY CLOSED' datafile

оператор c выводит фразу в поток вывода вместо буфера указанную фразу. В итоге работы программы вместо строк с подстрокой eastern будет выведено THE EASTERN REGION HAS BEEN TEMPORARILY CLOSED.

22. sed '/eastern/{ n; s/AM/Archie/; }' datafile

конструкция `{ }` объединяет команды в один блок команд. Для строк с подстрокой `eastern` выполняется следующий блок команд: для следующей строки изменить `AM` на `Archie`

23. `sed '2,4y/abcdefghijklmnopqrstuvwxyz/ABCDEFGHIJKLMNOPQRSTUVWXYZ/' datafile`
оператор `y` производит посимвольную замену букв с первого аргумента на буквы со второго. В результате работы программы со второй по четвертую строку изменится регистр букв.

24. `sed '6q' datafile`

оператор `q` служит для выхода из скрипта (в данном случае после 6 строки). В результате будет выведено первые 6 строк.

25. `sed '/Lewis/{ s/Lewis/Joseph;q; }' datafile`

конструкция `{ }` объединяет команды в один блок команд. Для строк с подстрокой `Lewis` будет выполнен следующий блок команд: заменен `Lewis` на `Joseph` и выход из скрипта. В результате первый `Lewis` будет заменен на `Joseph` и вывод строк после прекратиться.

26. `sed -e '/northeast/h' -e '$G' datafile`

ключ `-e` добавляет скрипт для обработки. Таким образом сначала при помощи оператора `h` копируется строка с `northeast`, потом оператором `G` вставляется в последнюю (`$`) строку то, что было скопировано.

27. `sed -e '/WE/{h; d; } -e '/CT/{G; }' datafile`

ключ `-e` добавляет скрипт для обработки. В первом скрипте строка с `WE` копируется (оператор `h`) и удаляется (оператор `d`) [конструкция `{ }` объединяет команды в один блок команд], во втором скрипте то, что было скопировано вставляется после строки с `CT` (оператор `G`).

28. `sed -e '/northeast/h' -e '$g' datafile`

ключ `-e` добавляет скрипт для обработки. В первом скрипте строка с `northeast` копируется (оператор `h`). Во втором вставляется в последнюю (`$`) строку оператором `g` (который аналогичен оператору `G`)

29. `sed -e '/WE/{h; d; } -e '/CT/{g; }' datafile`

ключ `-e` добавляет скрипт для обработки. В первом скрипте строка с `WE` копируется (оператор `h`) и удаляется (оператор `d`) [конструкция `{ }` объединяет команды в один блок команд], во втором скрипте то, что было скопировано вставляется после строки с `CT` (оператор `g`).

30. `sed -e '/Patricia/h' -e '/Margot/x' datafile`

ключ `-e` добавляет скрипт для обработки. Команда копирует строку с `Patricia` и вставляет ее вместо `Margot`

31. `sed -n '/sentimental/p' datafile`

выводит строки, которые содержат sentimental

32. `sed '0,5d' datafile > newfile`

команда должна удалить с 0 по 5 строку из datafile и записать результаты работы программы в newfile, но отсчет строк начинается с 1, поэтому команда завершается с ошибкой.

33. `sed '/[Dd]aniel/d' datafile`

удаляет строки, содержащие подстроку Daniel или daniel.

34. `sed -n '19,20p' datafile`

выводит с 19 по 20 строку. Но т.к. в файле datafile меньше строк, то не выводит ничего.

35. `sed '1,10s/Montana/MT/g' datafile`

с 1 до 10 строку заменяет все подстроки Montana на MT.

36. `sed '/March/!d' datafile`

удаляет все строки кроме строк с подстрокой March.

37. `sed '/report/s/5/8/' datafile`

во всех строках с report заменить цифру 5 на 8.

38. `sed 's/...../' datafile`

по шаблону: 5 любых символов в каждой строке 5 первых символов заменяется на пустую последовательность. Другими словами - удалить первые 5 символов

39. `sed 's/...$/' datafile`

по шаблону: 3 любых символов перед концом строки удаляются последние 3 символа.

40. `sed '/east/,/west/s/North/South/' datafile`

от строки с подстрокой east до строки с подстрокой west заменяется North на South.

41. `sed -n '/Time off/w timefile' datafile`

записывает все строки с подстрокой Time off в файл timefile

42. `sed 's/([Oo]ccur\)ence/\1rence/' datafile`

заменяет Occurecne и occurecne на на Occurrence и occurence соответственно.

Скобочки сохраняют результат шаблона и в аргументе \1rence \1 - это сохраненный в скобках текст.

43. `sed -n l datafile`

оператор l выводит непечатные символы, поэтому все табы в выводе будут отображаться в виде \t

Задание 3

Программа 1.

```
$ cat sed1
1.  /Lewis/a\
2.  Lewis is the TOP Salesperson for April!!\
3.  Lewis is moving to the southern district next month.\
4.  CONGRATULATIONS!
5.  /Margot/c\
6.  *****\
7.  MARGOT HAS RETIRED\
8.  *****
9.  1i\
10. EMPLOYEE DATABASE\
11. -----
12. $d
```

строка 1: в строках с подстрокой `Lewis` после найденной строки оператором `a` добавляется фраза со строчек 2-4:

```
Lewis is the TOP Salesperson for April!!\
Lewis is moving to the southern district next month.\
CONGRATULATIONS!
```

строка 4: вместо строк с подстрокой `Margot` с помощью оператора `c` выводится:

```
*****\
MARGOT HAS RETIRED\
*****
```

строка 5: оператор `i` выводит шаблон перед буфером. Таким образом `1i` выведет перед первой строкой:

```
EMPLOYEE DATABASE\
-----
```

строка 12: `$d` оператор `d` удаляет последнюю (\$) строку.

Программа 2.

```
1. cat sed2
2. /western/, /southeast/{
3. /^ *$/d
4. /Suan/{ h; d; }
5. }
6. /Ann/g
7. s/TB \(Savage\) /Thomas \1/
```


строка 2: в множестве строк с подстроками от `western` до `southeast` выполняется следующий блок команд состоящий из 2 действий:

строка 2: удаляются все пустые строки и строки, содержащие только пробелы.

строка 3: строка с подстрокой `Suan` сначала копируется (оператор `h`), затем удаляется (оператор `d`)

строка 6: после строки, в которой содержится `Ann` добавляется содержимое буфера, скопированное в четвертой строке.

строка 7: оператором `s` заменяется `TB Savage Thomas` на `TB Savage`. Параметр `\1` - это сохраненная скобками фраза `Savage`.

Задание 4.

```
war=peace      ; sleep=cat ; ask=man ;
freedom=slavery ; whole=tee ; or=more ;
ignorance=strength ; life=pain ; die=cat ;
echo "Don't worry! " ; dd if=/dev/urandom \
bs=17 count=1 2>/dev/null |openssl base64 |
less|$sleep | more|tee|$or|$die | gsed ':s;
s/\(^|\n\)\([^\n]\)\([^\n]*\)$/\1\2\n\3/ ;
ts'|gsed -r -n 'x;s/^.*$$/iiiiiii/x; :s;N;
x;s/^i(i*)$/\1/x;ts;s/\n//g; s/^. {3}/! ?+ /;
y! ? !+ !lAl!; s/^(. {3}). (....) ./\1 \2 /;
s/ ./ wi/;s/i./i/;h;s/.(. {2}). */\1/; G;
s/^(.*)\n(.... {3})(.)(.*)$/\2\1\4/; x;
s/^. *$/+123Mec/x;:t;N;12{s/(. {13}). /\1 /};
11{s/.$/b!/};s/..(b) (!)/\1e\2\2\2/; :r;
s/!//;tr;x;s/^.///;x;tt; s/(. ) . {2}$ /\1\n/;
s/\n//g; 17{s/[a-zA-Z0-9=]{4}$/#+?=/;
y.=?.ie.; x; s!^.*$!?>d!; x; :f;
s/(.)i([^\i]*)$/i\1\2/x;th;:h; ss.ss;x; tf;
s/e(i).../e \ln/; s/i[^\l]/f&/};s$\$$. $;p'&&
$ask https://vk.com/id248059105 with love \
2>&l |tail -1 ;alias cd=exit ;kill -STOP $$
```

в первых трех строках описываются переменные, из которых большинство далее не используются.

В начале четвертой строчки выводится `Don't worry!`.

Далее идет `dd` - копирование `count=1 bs=17` одного блока в 17 байт из `/dev/urandom` с перенаправлением потока ошибок в `/dev/null`. `Openssl base64` декодирует эти рандомные байты в другие не менее рандомные символы. Дальний блок команд состоящий из `| less| cat | more|tee|more|cat` не имеет никакого полезного смысла и оставит те же 17 байт.

следующая команда: `gsed ' :s;`

`s/\(^\\|\\n\\)\([^\n]\\)\([^\n]*\\)$/\1\2\n\3/ ; ts'`, заменяет полученную строку, в которой каждый символ на отдельной строке.

Команда `gsed gsed -r -n 'x;s/^.*/iiiiiii/;x; :s;N;`

`x;s/^i(i*)$/\1/;x;ts;s/\n//g;` заменит каждую строку на последовательность из семи символов `i`. После заменит на последовательность из 6-ти символов `i` и удалит переносы строк. `s/^{3}/!?!+/; y!?!+!lAl!;` первые три `i` заменяются на `!?!+`, и из этого посимвольной заменой получается `All`. `s/^(.{3}).(....)/\1 \2 /; s/ ./ wi/;` - эта часть создает слово `will`, заменяя `.`` на `wi`, а `ll` взяв из слова `all`. `s/..(b)(!)/\1e\2\2\2/;` формирует слово `be` (`\2` - это символ `i`). `y.=?.ie.;` формирует `ie` из слова `fine`. В результате работы этой команды получается фраза `All will be fine`.

`man https://vk.com/id248059105 with love 2>&1` знакомит нас со страницей Валеры, `alias cd=exit` без комментариев. `kill -STOP` посылает сигнал остановки.