

ЛАБОРАТОРНАЯ РАБОТА №9 ОБРАБОТКА СТРОК (РАБОТА С ТЕКСТОВЫМИ ДАННЫМИ)

Цель работы – практическое знакомство со способами эффективной обработки текста при помощи интерфейса командной строки и набора стандартных утилит

Задание 1:

Используя утилиты hexdump и strings, вывести на экран содержимое одного из перечисленных ниже файлов из каталога /bin. Позиция файла для распечатки определяется номером бригады. Имена файлов для выполнения задания 1: tar, sort, sed, ping, vi, unlink, uname, touch, sleep, sty.

Решение:

```
ubuntu@ubuntu:~$ hexdump -C /bin/sleep
00000000  7f 45 4c 46 02 01 01 00  00 00 00 00 00 00 00 00 |.ELF.....
00000010  03 00 3e 00 01 00 00 00  50 28 00 00 00 00 00 00 |...>....P(.....
00000020  40 00 00 00 00 00 00 00  d8 91 00 00 00 00 00 00 |@.....
00000030  00 00 00 00 40 00 38 00  0d 00 40 00 1e 00 1d 00 |....@.8....@....
00000040  06 00 00 00 04 00 00 00  40 00 00 00 00 00 00 00 |.....@.....
00000050  40 00 00 00 00 00 00 00  40 00 00 00 00 00 00 00 |@.....@.....
00000060  d8 02 00 00 00 00 00 00  d8 02 00 00 00 00 00 00 |.....
00000070  08 00 00 00 00 00 00 00  03 00 00 00 04 00 00 00 |.....
00000080  18 03 00 00 00 00 00 00  18 03 00 00 00 00 00 00 |.....
00000090  18 03 00 00 00 00 00 00  1c 00 00 00 00 00 00 00 |.....
000000a0  1c 00 00 00 00 00 00 00  01 00 00 00 00 00 00 00 |.....
000000b0  01 00 00 00 04 00 00 00  00 00 00 00 00 00 00 00 |.....
000000c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 |.....
000000d0  f0 14 00 00 00 00 00 00  f0 14 00 00 00 00 00 00 |.....
000000e0  00 10 00 00 00 00 00 00  01 00 00 00 05 00 00 00 |.....
000000f0  00 20 00 00 00 00 00 00  00 20 00 00 00 00 00 00 |. .....
00000100  00 20 00 00 00 00 00 00  21 3c 00 00 00 00 00 00 |. .....!<.....
00000110  21 3c 00 00 00 00 00 00  00 10 00 00 00 00 00 00 |!<.....
00000120  01 00 00 00 04 00 00 00  00 60 00 00 00 00 00 00 |.....
00000130  00 60 00 00 00 00 00 00  00 60 00 00 00 00 00 00 |`.....
00000140  38 1f 00 00 00 00 00 00  38 1f 00 00 00 00 00 00 |8.....8.....
00000150  00 10 00 00 00 00 00 00  01 00 00 00 06 00 00 00 |.....
00000160  b0 8b 00 00 00 00 00 00  b0 9b 00 00 00 00 00 00 |.....
00000170  b0 9b 00 00 00 00 00 00  d0 04 00 00 00 00 00 00 |.....
00000180  88 06 00 00 00 00 00 00  00 10 00 00 00 00 00 00 |.....
00000190  02 00 00 00 06 00 00 00  78 8c 00 00 00 00 00 00 |.....x.....
000001a0  78 9c 00 00 00 00 00 00  78 9c 00 00 00 00 00 00 |x.....x.....
000001b0  f0 01 00 00 00 00 00 00  f0 01 00 00 00 00 00 00 |.....
```

```
ubuntu@ubuntu:~$ strings -n 10 /bin/sort
/lib64/ld-linux-x86-64.so.2
libpthread.so.0
_ITM_deregisterTMCloneTable
_ITM_registerTMCloneTable
pthread_join
pthread_mutex_init
pthread_sigmask
pthread_cond_signal
pthread_cond_init
pthread_cond_wait
pthread_mutex_lock
__errno_location
pthread_mutex_unlock
pthread_create
__stpcpy_chk
__printf_chk
clearerr_unlocked
dcgettext
fflush_unlocked
__read_chk
__stack_chk_fail
program_invocation_name
__assert_fail
__ctype_get_mb_cur_max
posix_fadvise
```

Задание 2:

Подсчитать общее количество файлов (каталогов) в одном из перечисленных ниже каталогов. Каталог для подсчета количества определяется номером бригады. Имена каталогов для выполнения задания 2: /bin, /etc, /lib, /proc, /usr, /var, /dev, /sbin, /sys, /root

Решение:

```
ubuntu@ubuntu:~$ ls -1 /sbin | wc -l
522
ubuntu@ubuntu:~$
```

Задание 3:

Найти общее количество процессов, выполняющихся в системе в данный момент.

Решение:

```
ubuntu@ubuntu:~$ ps aux | wc -l
194
ubuntu@ubuntu:~$
```

Задание 4:

Вывести список выполняющихся процессов, в именах которых присутствует слово manager и отсутствует слово grep

Решение:

```
ubuntu@ubuntu:~$ ps aux | grep manager | grep -v grep
root          126  0.0  0.0      0     0 ?        I<   13:32   0:00 [charger_ma
ager]
ubuntu@ubuntu:~$
```

Задание 5:

Создать текстовый файл, содержащий набор строк вида: 123

178

176

755

713

873

С помощью утилиты grep найти строки, в которых есть цифра 7, после которой находится одна из цифр — 1, 3 или 5.

Решение:

```
ubuntu@ubuntu:~$ nano z5.txt
ubuntu@ubuntu:~$ grep "7[1|3|5]" z5.txt
755
713
873
ubuntu@ubuntu:~$
```

Задание 6:

Создать текстовый файл, содержащий набор строк вида: starfish

starless samscripter stellar microsrar ascender sacrifice scalar

С помощью утилиты grep найти строки, начинающиеся на букву s и заканчивающиеся на букву r

Решение:

```
GNU nano 4.8                               6.txt                                Modified
starfish
starless
samscripter
stellar
microsrar
ascender
sacrifice
scalar

^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify
^X Exit          ^R Read File      ^\ Replace       ^U Paste Text    ^T To Spell

ubuntu@ubuntu:~$ nano 6.txt
ubuntu@ubuntu:~$ grep "\b[Ss]\w*[Rr]\b" 6.txt
samscripter
stellar
scalar
ubuntu@ubuntu:~$
```

Задание 7:

Создать текстовый файл, содержащий простейшие адреса электронной почты вида username@website.com. С помощью утилиты grep найти строки, содержащие правильные простейшие адреса. Проверить возможность использования более сложного регулярного выражения для распознавания адресов, содержащих другие допустимые символы.

Решение:

```
ubuntu@ubuntu:~$ nano 7.txt
ubuntu@ubuntu:~$ grep -E -o "\b[A-Za-z0-9._%?^+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,6}\b" 7.txt
wqrqwt@mail.ru
152315324@yandex.ru
asgasekjg@gmail.com
ubuntu@ubuntu:~$
```

Задание 8:

На произвольном примере продемонстрировать работу утилиты tr Создать текстовый

файл, содержащий допустимые и недопустимые IP-адреса, например 127.0.0.1

255.255.255.255

12.34.56

123.256.0.0

1.23.099.255

0.79.378.111

С помощью утилиты grep и руководства man найти строки, содержащие допустимые четырехбайтовые IP адреса.

Решение:

```
ubuntu@ubuntu:~$ nano 8.txt
ubuntu@ubuntu:~$ grep -E "(25[0-5]|2[0-4][0-9]| [01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]| [01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]| [01]?[0-9][0-9]?)\.(25[0-5]|2[0-4][0-9]| [01]?[0-9][0-9]?)" 8.txt
127.0.0.1
255.255.255.255
1.23.099.255
ubuntu@ubuntu:~$
```

Задание 9:

Создать текстовый файл, содержащий корректные и некорректные номера телефонов ведомственной АТС объемом 399 номеров, номера с 000 до 399 – корректные, 0, 400, 900 –некорректные.

С помощью утилиты grep и руководства man найти строки, содержащие допустимые номера телефонов.

Решение:

```
ubuntu@ubuntu:~$ nano 9.txt
ubuntu@ubuntu:~$ grep -E "(0[0-9][0-9]|1-2[0-9][0-9]|3[0-9][0-9])" 9.txt
000
399
ubuntu@ubuntu:~$ cat 9.txt
000
400
777
1
2
56
900
999
399

ubuntu@ubuntu:~$
```