

# Лабораторная работа №6

## Отчёт к лабораторной работе

Зайцева Анна Дмитриевна

### Table of Contents

#### Цель работы

Цель работы — Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

#### Задание

1. Осуществите вход в систему, используя соответствующее имя пользователя.
2. Запишите в файл `file.txt` названия файлов, содержащихся в каталоге `/etc`.  
Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
3. Выведите имена всех файлов из `file.txt`, имеющих расширение `.conf`, после чего запишите их в новый текстовый файл `conf.txt`.
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа `s`? Предложите несколько вариантов, как это сделать.
5. Выведите на экран (постранично) имена файлов из каталога `/etc`, начинающиеся с символа `h`.
6. Запустите в фоновом режиме процесс, который будет записывать в файл `~/logfile` файлы, имена которых начинаются с `log`.
7. Удалите файл `~/logfile`.
8. Запустите из консоли в фоновом режиме редактор `gedit`.
9. Определите идентификатор процесса `gedit`, используя команду `ps`, конвейер и фильтр `grep`. Как ещё можно определить идентификатор процесса?
10. Прочтите справку (`man`) команды `kill`, после чего используйте её для завершения процесса `gedit`.
11. Выполните команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man`.
12. Воспользовавшись справкой команды `find`, выведите имена всех директорий, имеющих в вашем домашнем каталоге.

## Выполнение лабораторной работы

1. Я осуществила вход в систему, используя соответствующее имя пользователя и пароль и перешла в терминал (Рис. [-@fig:001]):

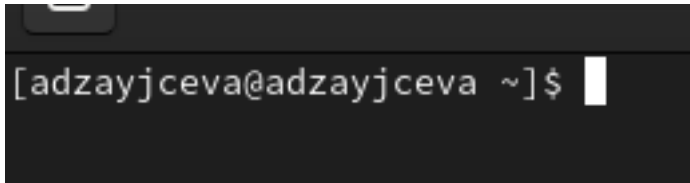


Рис. 1

2. Записала в файл `file.txt` названия файлов, содержащихся в каталоге `/etc` (команда: `ls -a /etc > file.txt`) (Рис. [-@fig:002]). Дописала в этот же файл названия файлов, содержащихся в вашем домашнем каталоге (команда: `ls -a ~ >> file.txt`) (Рис. [-@fig:003]). Проверку своих действий я осуществляла с помощью команды `cat file.txt`.

```
[adzayjceva@adzayjceva ~]$ ls -a /etc > file.txt
[adzayjceva@adzayjceva ~]$ cat file.txt
.
..
abrt
adjtime
aliases
alsa
alternatives
anaconda
anthy-unicode.conf
appstream.conf
asound.conf
audit
authselect
avahi
bash_completion.d
bashrc
bindresvport.blacklist
binfmt.d
bluetooth
brlapi.key
brltty
brltty.conf
ceph
chkconfig.d
chromium
chrony.conf
chrony.keys
```

*Puc. 2*

```
[adzayjceva@adzayjceva ~]$ ls -a ~ >> file.txt
[adzayjceva@adzayjceva ~]$ cat file.txt
.
..
abrt
adjtime
aliases
alsa
alternatives
anaconda
anthy-unicode.conf
appstream.conf
asound.conf
audit
authselect
avahi
bash_completion.d
bashrc
bindresvport.blacklist
binfmt.d
bluetooth
brlapi.key
brltty
brltty.conf
ceph
chkconfig.d
chromium
chrony.conf
chrony.keys
cifs-utils
containers
crypto-policies
crypttab
csh.cshrc
csh.login
cups
cupshelpers
dbus-1
dconf
default
depmod.d
```

Рис. 3

3. Вывела имена всех файлов из file.txt, имеющих расширение .conf (команда: *grep .conf file.txt*) (Рис. [-@fig:004]), после чего записала их в новый текстовый файл conf.txt (команда: *grep .conf file.txt > conf.txt*) (Рис. [-@fig:005]):

```
[adzayjceva@adzayjceva ~]$ grep .conf file.txt
anthy-unicode.conf
appstream.conf
asound.conf
brltty.conf
chkconfig.d
chrony.conf
dconf
dley-na-renderer-service.conf
dley-na-server-service.conf
dnsmasq.conf
dracut.conf
dracut.conf.d
extlinux.conf
fprind.d.conf
fuse.conf
host.conf
idmapd.conf
jwhois.conf
kdump.conf
krb5.conf
krb5.conf.d
ld.so.conf
ld.so.conf.d
libaudit.conf
libuser.conf
locale.conf
logrotate.conf
makedumpfile.conf.sample
man_db.conf
mke2fs.conf
mtools.conf
netconfig
nfs.conf
nfsmount.conf
nsswitch.conf
nsswitch.conf.bak
opensc.conf
opensc-x86_64.conf
passwdqc.conf
pkgconfig
```

*Рис. 4*

- 1 anthy-unicode.conf
- 2 appstream.conf
- 3 asound.conf
- 4 brltty.conf
- 5 chkconfig.d
- 6 chrony.conf
- 7 dconf
- 8 dleyna-renderer-service.conf
- 9 dleyna-server-service.conf
- 10 dnsmasq.conf
- 11 dracut.conf
- 12 dracut.conf.d
- 13 extlinux.conf
- 14 fprintd.conf
- 15 fuse.conf
- 16 host.conf
- 17 idmapd.conf
- 18 jwhois.conf
- 19 kdump.conf
- 20 krb5.conf
- 21 krb5.conf.d
- 22 ld.so.conf
- 23 ld.so.conf.d
- 24 libaudit.conf
- 25 libuser.conf
- 26 locale.conf
- 27 logrotate.conf
- 28 makedumpfile.conf.sample
- 29 man\_db.conf
- 30 mke2fs.conf



Рис. 5

4. Определила, какие файлы в моём домашнем каталоге имеют имена, начинающиеся с символа 'с' (команды: `find ~ -maxdepth 1 -name "с" -print`, `ls -a ~ | grep с**`) (Рис. [-@fig:006])(Рис. [-@fig:007]):

```
[adzayjceva@adzayjceva ~]$ find ~ -maxdepth 1 -name "с*" -print
/home/adzayjceva/conf.txt
[adzayjceva@adzayjceva ~]$
```

Рис. 6

```
[adzayjceva@adzayjceva ~]$ ls -a ~ | grep с*
conf.txt
[adzayjceva@adzayjceva ~]$
```

Рис. 7

5. Вывела на экран (постранично) имена файлов из каталога /etc, начинающиеся с символа h (команда: `find /etc -maxdepth 1 -name "h" -print*`) (Рис. [-@fig:008]):

```
[adzayjceva@adzayjceva ~]$ find /etc -maxdepth 1 -name "h*" -print
/etc/hp
/etc/httpd
/etc/host.conf
/etc/hosts
/etc/hostname
[adzayjceva@adzayjceva ~]$
```

Рис. 8

6. Запустила в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log (команда: `find / -name "log" > logfile &`) (Рис. [-@fig:009]). Проверила правильность выполнения команды с помощью команды `cat logfile*` (Рис. [-@fig:010]):

```
[adzayjceva@adzayjceva ~]$ find / -name "log*" > logfile &
[4] 6498
[adzayjceva@adzayjceva ~]$ find: '/boot/loader/entries': Отказано в доступе
find: '/boot/lost+found': Отказано в доступе
find: '/boot/efi': Отказано в доступе
find: '/boot/grub2': Отказано в доступе
find: '/proc/tty/driver': Отказано в доступе
find: '/proc/1/task/1/fd': Отказано в доступе
find: '/proc/1/task/1/ns': Отказано в доступе
find: '/proc/1/fd': Отказано в доступе
find: '/proc/1/map_files': Отказано в доступе
find: '/proc/1/ns': Отказано в доступе
find: '/proc/2/task/2/fd': Отказано в доступе
find: '/proc/2/task/2/ns': Отказано в доступе
find: '/proc/2/fd': Отказано в доступе
find: '/proc/2/map_files': Отказано в доступе
find: '/proc/2/ns': Отказано в доступе
find: '/proc/3/task/3/fd': Отказано в доступе
find: '/proc/3/task/3/ns': Отказано в доступе
find: '/proc/3/fd': Отказано в доступе
find: '/proc/3/map_files': Отказано в доступе
find: '/proc/3/ns': Отказано в доступе
find: '/proc/4/task/4/fd': Отказано в доступе
find: '/proc/4/task/4/ns': Отказано в доступе
find: '/proc/4/fd': Отказано в доступе
find: '/proc/4/map_files': Отказано в доступе
find: '/proc/4/ns': Отказано в доступе
find: '/proc/6/task/6/fd': Отказано в доступе
find: '/proc/6/task/6/ns': Отказано в доступе
find: '/proc/6/fd': Отказано в доступе
find: '/proc/6/map_files': Отказано в доступе
find: '/proc/6/ns': Отказано в доступе
find: '/proc/9/task/9/fd': Отказано в доступе
find: '/proc/9/task/9/ns': Отказано в доступе
find: '/proc/9/fd': Отказано в доступе
find: '/proc/9/map_files': Отказано в доступе
find: '/proc/9/ns': Отказано в доступе
find: '/proc/10/task/10/fd': Отказано в доступе
find: '/proc/10/task/10/ns': Отказано в доступе
find: '/proc/10/fd': Отказано в доступе
```

Рис. 9

```
[adzajjceva@adzajjceva ~]$ cat logfile
/dev/log
/home/adzajjceva/.mozilla/firefox/mqkhvisp.default-release/logins-backup.json
/home/adzajjceva/.mozilla/firefox/mqkhvisp.default-release/logins.json
/home/adzajjceva/.local/share/keyrings/login.keyring
/home/adzajjceva/laboratory/.git/logs
/home/adzajjceva/laboratory/2021-2022/OS/lab03/academic-laboratory-report-template/.git/logs
/home/adzajjceva/laboratory/2021-2022/OS/lab03/academic-presentation-markdown-template/.git/logs
/home/adzajjceva/.cabal/logs
/home/adzajjceva/blog/.git/logs
/home/adzajjceva/blog/.git/modules/public/logs
/home/adzajjceva/adzajjceva.github.io/.git/logs
/home/adzajjceva/logfile
/proc/sys/dev/scsi/logging_level
/proc/sys/net/ipv4/conf/all/log_martians
/proc/sys/net/ipv4/conf/default/log_martians
/proc/sys/net/ipv4/conf/enp0s3/log_martians
/proc/sys/net/ipv4/conf/lo/log_martians
/proc/1/task/1/loginuid
/proc/1/loginuid
/proc/2/task/2/loginuid
/proc/2/loginuid
/proc/3/task/3/loginuid
/proc/3/loginuid
/proc/4/task/4/loginuid
/proc/4/loginuid
/proc/6/task/6/loginuid
/proc/6/loginuid
/proc/9/task/9/loginuid
/proc/9/loginuid
/proc/10/task/10/loginuid
/proc/10/loginuid
/proc/11/task/11/loginuid
/proc/11/loginuid
/proc/12/task/12/loginuid
/proc/12/loginuid
/proc/13/task/13/loginuid
/proc/13/loginuid
/proc/14/task/14/loginuid
/proc/14/loginuid
```

Рис. 10

7. Удалила файл ~/logfile (команда: *rm logfile*). Правильность выполнения команды проверила командой *ls* (Рис. [-@fig:011]):

```
[adzajjceva@adzajjceva ~]$ rm logfile
[adzajjceva@adzajjceva ~]$ ls
abc1      australia  conf.txt  file.txt  my_os      pandoc-crossref-linux.tar.xz  ski.places  Документы  Изображения  Общедоступные  Шаблоны
adzajjceva.github.io  blue      feathers  laboratory  pandoc-2.18-linux-arm64.tar.gz  play        Видео      Загрузки    Музыка        'Рабочий стол'
```

Рис. 11

8. Запустила из консоли в фоновом режиме редактор gedit (команда: *gedit &*) (Рис. [-@fig:012]). Открылось отдельное окно с редактором с файлом “Без имени 1”:

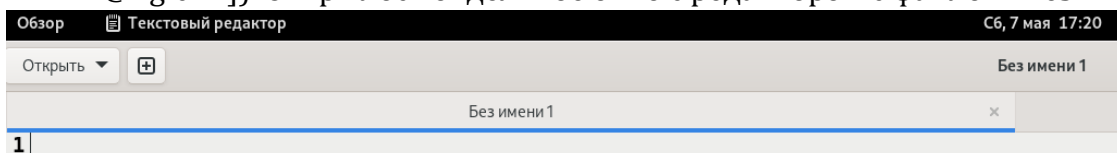


Рис. 12

9. Определила идентификатор процесса gedit, используя команду *ps*, конвейер и фильтр *grep* (команда: *ps | grep -i "gedit"*). Также можно определить идентификатор процесса с помощью команд *pgrep gedit* и *pidof gedit* (Рис. [-@fig:013]):

```
[adzayjceva@adzayjceva ~]$ ps | grep -i "gedit"
[4]   Завершён      gedit
[adzayjceva@adzayjceva ~]$ pgrep gedit
4497
[adzayjceva@adzayjceva ~]$ pidof gedit
4497
[adzayjceva@adzayjceva ~]$
```

Рис. 13

10. Прочла справку (man) команды kill (команда: *man kill*), после чего использовала её для завершения процесса gedit (команда: *kill 4497*), поскольку в 9 пункте я получила именно такой идентификатор для процесса gedit (Рис. [-@fig:014]):

```
[adzayjceva@adzayjceva ~]$ pidof gedit
4497
[adzayjceva@adzayjceva ~]$ man kill
[adzayjceva@adzayjceva ~]$ kill 4497
[adzayjceva@adzayjceva ~]$
```

Рис. 14

11. Выполнила команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man (команды: *man df* и *man du*) (Рис. [-@fig:015]):

```

[adzayjceva@adzayjceva ~]$ man df
[adzayjceva@adzayjceva ~]$ man du
[adzayjceva@adzayjceva ~]$ df
Файловая система 1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs          986140            0   986140            0% /dev
tmpfs             1006532          20908   985624            3% /dev/shm
tmpfs             402616          1408    401208            1% /run
/dev/sda2         82836480        6921916   74280308           9% /
tmpfs             1006532          184    1006348            1% /tmp
/dev/sda2         82836480        6921916   74280308           9% /home
/dev/sda1         996780         175308    752660           19% /boot
tmpfs             201304           140     201164            1% /run/user/1000
/dev/sr0          59642            59642            0          100% /run/media/adzayjceva/VBox_GAs_6.1.26
[adzayjceva@adzayjceva ~]$ du
8      ./mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
8      ./mozilla/extensions
0      ./mozilla/plugins
0      ./mozilla/firefox/Crash Reports/events
4      ./mozilla/firefox/Crash Reports
0      ./mozilla/firefox/Pending Pings
0      ./mozilla/firefox/mqkhvisp.default-release/minidumps
0      ./mozilla/firefox/mqkhvisp.default-release/crashes/events
4      ./mozilla/firefox/mqkhvisp.default-release/crashes
2696   ./mozilla/firefox/mqkhvisp.default-release/security_state
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files/journals
772    ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3561288849sdhlie.files
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/1451318868ntouromlalnodry--epcr.files
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/1657114595AmcateirvtiSty.files
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/2823318777ntouromlalnodry--naod.files
0      ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/2918063365piupsah.files
9180   ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb
9184   ./mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome
9184   ./mozilla/firefox/mqkhvisp.default-release/storage/permanent
0      ./mozilla/firefox/mqkhvisp.default-release/storage/temporary
0      ./mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-5210-4dfe-acef-12b3a140b812^use
44     ./mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-5210-4dfe-acef-12b3a140b812^use
48     ./mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-5210-4dfe-acef-12b3a140b812^use
12     ./mozilla/firefox/mqkhvisp.default-release/storage/default/https+++yandex.ru/ls
16     ./mozilla/firefox/mqkhvisp.default-release/storage/default/https+++yandex.ru
12     ./mozilla/firefox/mqkhvisp.default-release/storage/default/https+++accounts.google.com/ls
16     ./mozilla/firefox/mqkhvisp.default-release/storage/default/https+++accounts.google.com
12     ./mozilla/firefox/mqkhvisp.default-release/storage/default/https+++mail.google.com/ls

```

Рис. 15

12. Воспользовавшись справкой команды `find` (команда: `man find`), вывела имена всех директорий, имеющихя в моём домашнем каталоге (команда: `find ~ -type d`) (Рис. [-@fig:016]):

```
[adzayjceva@adzayjceva ~]$ man find
[adzayjceva@adzayjceva ~]$ find ~ -type d
/home/adzayjceva
/home/adzayjceva/.mozilla
/home/adzayjceva/.mozilla/extensions
/home/adzayjceva/.mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
/home/adzayjceva/.mozilla/plugins
/home/adzayjceva/.mozilla/firefox
/home/adzayjceva/.mozilla/firefox/Crash Reports
/home/adzayjceva/.mozilla/firefox/Crash Reports/events
/home/adzayjceva/.mozilla/firefox/Pending Pings
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/minidumps
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/crashes
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/crashes/events
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/security_state
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3870112724rsegr
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3870112724rsegr
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/3561288849sdhli
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/1451318868ntour
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/1657114595Amcat
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/2823318777ntour
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/permanent/chrome/idb/2918063365piups
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/temporary
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-521
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-521
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default/moz-extension+++ee641be0-521
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default/https+++yandex.ru
/home/adzayjceva/.mozilla/firefox/mqkhvisp.default-release/storage/default/https+++yandex.ru/ls
```

Рис. 16

## Ответы на контрольные вопросы

1. В системе по умолчанию открыто три специальных потока: – stdin – стандартный поток ввода (по умолчанию: клавиатура), файловый дескриптор 0; – stdout – стандартный поток вывода (по умолчанию: консоль), файловый дескриптор 1; – stderr – стандартный поток вывод сообщений об ошибках (по умолчанию: консоль), файловый дескриптор 2. Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода stdout.
2. “>” Перенаправление вывода в файл “>>” Перенаправление вывода в файл и открытие файла в режиме добавления (данные добавляются в конец файла).
3. Конвейер служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий: команда 1 | команда 2 (это означает, что вывод команды 1 передаётся на ввод команде 2).
4. Процесс рассматривается операционной системой как заявка на потребление всех видов ресурсов, кроме одного – процессорного времени. Этот последний важнейший ресурс распределяется операционной системой между другими единицами работы – потоками, которые и получили свое название благодаря тому, что они представляют собой последовательности (потоки выполнения)

команд. Процесс – это выполнение программы. Он считается активной сущностью и реализует действия, указанные в программе. Программа представляет собой статический набор команд, а процесс это набор ресурсов и данных, использующихся при выполнении программы.

5. `pid`: идентификатор процесса (PID) процесса (process ID), к которому вызывают метод `gid`: идентификатор группы UNIX, в котором работает программа.
6. Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать знак амперсанда `&`. Запущенные фоновые программы называются задачами. Ими можно управлять с помощью команды `jobs`, которая выводит список запущенных в данный момент задач.
7. `top` – это консольная программа, которая показывает список работающих процессов в системе. Программа в реальном времени отсортирует запущенные процессы по их нагрузке на процессор. `htop` – это продвинутый консольный мониторинг процессов. Утилита выводит постоянно меняющийся список системных процессов, который сортируется в зависимости от нагрузки на ЦПУ. Если делать сравнение с `top`, то `htop` показывает абсолютно все процессы в системе, время их непрерывного использования, загрузку процессоров и расход оперативной памяти.
8. `find` – это команда для поиска файлов и каталогов на основе специальных условий. Ее можно использовать в различных обстоятельствах, например, для поиска файлов по разрешениям, владельцам, группам, типу, размеру и другим подобным критериям. Команда `find` имеет такой синтаксис: `find [папка] [параметры] критерий шаблон [действие]` Папка – каталог в котором будем искать Параметры – дополнительные параметры, например, глубина поиска, и т.д. Критерий – по какому критерию будем искать: имя, дата создания, права, владелец и т.д. Шаблон – непосредственно значение по которому будем отбирать файлы. Основные параметры: `-R` никогда не открывать символические ссылки `-L` - получает информацию о файлах по символическим ссылкам. Важно для дальнейшей обработки, чтобы обрабатывалась не ссылка, а сам файл. `-maxdepth` - максимальная глубина поиска по подкаталогам, для поиска только в текущем каталоге установите 1. `-depth` - искать сначала в текущем каталоге, а потом в подкаталогах `-mount` искать файлы только в этой файловой системе. `-version` - показать версию утилиты `find` `-print` - выводить полные имена файлов `-type f` - искать только файлы `-type d` - поиск папки в Linux Основные критерии: `-name` - поиск файлов по имени `-perm` - поиск файлов в Linux по режиму доступа `-user` - поиск файлов по владельцу `-group` - поиск по группе `-mtime` - поиск по времени модификации файла `-atime` - поиск файлов по дате последнего чтения `-nogroup` - поиск файлов, не принадлежащих ни одной группе `-nouser` - поиск файлов без владельцев `-newer` - найти файлы новее чем указанный `-size` - поиск файлов в Linux по их размеру Примеры: `find ~ -type d` поиск директорий в домашнем каталоге `find ~ -type f -name “.*”` поиск скрытых файлов в домашнем каталоге
9. Файл по его содержимому можно найти с помощью команды `grep`: «`grep`-г “слово/выражение, которое нужно найти”».

10. Утилита `df`, позволяет проанализировать свободное пространство на всех подключенных к системе разделах.
11. При выполнении команды `du` (без указания папки и опции) можно получить все файлы и папки текущей директории с их размерами. Для домашнего каталога: `du ~/`
12. Основные сигналы (каждый сигнал имеет свой номер), которые используются для завершения процесса:

**SIGINT** – самый безобидный сигнал завершения, означает `Interrupt`. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш `Ctrl+C`. Процесс правильно завершает все свои действия и возвращает управление;

**SIGQUIT** – это еще один сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей что нужно завершиться и программа может выполнить корректное завершение или проигнорировать сигнал. В отличие от предыдущего, она генерирует дампы памяти. Сочетание клавиш `Ctrl+\/`;

**SIGHUP** – сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом;

**SIGTERM** – немедленно завершает процесс, но обрабатывается программой, поэтому позволяет ей завершить дочерние процессы и освободить все ресурсы;

**SIGKILL** – тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными.

Также для передачи сигналов процессам в Linux используется утилита `kill`, её синтаксис: `kill [-сигнал] pid_процесса`. Сигнал представляет собой один из выше перечисленных сигналов для завершения процесса. Перед тем, как выполнить остановку процесса, нужно определить его PID. Для этого используют команды `ps` и `grep`. Команда `ps` предназначена для вывода списка активных процессов в системе и информации о них. Команда `grep` запускается одновременно с `ps` (в канале) и будет выполнять поиск по результатам команды `ps`.

Утилита `killall` – это оболочка для `kill`, она ведет себя точно так же, и имеет тот же синтаксис, только в качестве идентификатора процесса ей нужно передать его имя.

`killall` работает аналогично двум предыдущим утилитам. Она тоже принимает имя процесса в качестве параметра и ищет его PID в директории `/proc`. Но эта утилита обнаружит все процессы с таким именем и завершит их.

## Вывод

В ходе лабораторной работы я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.