

Отчет по второй лабораторной работе №6

Тулеева Валерия, НБИбд-01-20

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук Кафедра прикладной
информатики и теории вероятностей

1 Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Оглавление:

1. Введение:
 - a) Цель работы
2. Описание результатов выполнения задания;
3. Вывод;
4. Контрольные вопросы.

2 Введение:

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.

Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

Команда `find` используется для поиска и отображения имён файлов, соответствующих заданной строке символов.

Найти в текстовом файле указанную строку символов позволяет команда `grep`.

Команда `df` показывает размер каждого смонтированного раздела диска.

Команда `du` показывает число килобайт, используемое каждым файлом или каталогом.

Любой команде, выполняемой в системе, присваивается идентификатор процесса (process ID). Получить информацию о процессе и управлять им, пользуясь идентификатором процесса, можно из любого окна командного интерпретатора.

Команда `ps` используется для получения информации о процессах.

3 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.

4 Описание результатов выполнения

Задания:

1. Записала в файл file.txt названия файлов, содержащихся в каталоге /etc (рис 1.1):

```
cat >> file.txt
valeriya@valeriya-Aspire-V3-572G:~$ cat >> file.txt
[...]
valeriya@valeriya-Aspire-V3-572G:~$ cat >> file.txt
adduser.conf
anacrontab
apg.conf
appstream.conf
bash.bashrc
bash_completion
bindresvport.blacklist
brlapi.key
login.defs
logrotate.conf
lsb-release
ltrace.conf
machine-id
magic
magic.mime
mailcap
mailcap.order
manpath.config
mime.types
mke2fs.conf
[...]
```

Рис 1.1. Запись файлов из каталога /etc в файл

Дописала в файл названия файлов, содержащихся в домашнем каталоге:

```
cat >> file.txt
valeriya@valeriya-Aspire-V3-572G:~$ cat >> file.txt
equiplist2
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 1.2. Добавление в файл file.txt файлов из домашнего каталога

2. Вывела имена всех файлов из file.txt, имеющих расширение .conf(рис 2.1):

grep .conf file.txt

```
valeriya@valeriya-Aspire-V3-572G:~$ grep .conf file.txt
debconf.conf
deluser.conf
e2scrub.conf
fprintd.conf
fuse.conf
pnm2ppa.conf
popularity-contest.conf
rc.conf.save
gai.conf
hdparm.conf
rsyslog.conf
rygel.conf
sensors3.conf
signond.conf
host.conf
kernel-lmg.conf
kerneloops.conf
ld.so.conf
libao.conf
libaudit.conf
updatedb.conf
usb_modeswitch.conf
xattr.conf
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 2.1. Вывод всех файлов имеющих расширение .conf

3. Записала выведенные имена в новый текстовый файл conf.txt:

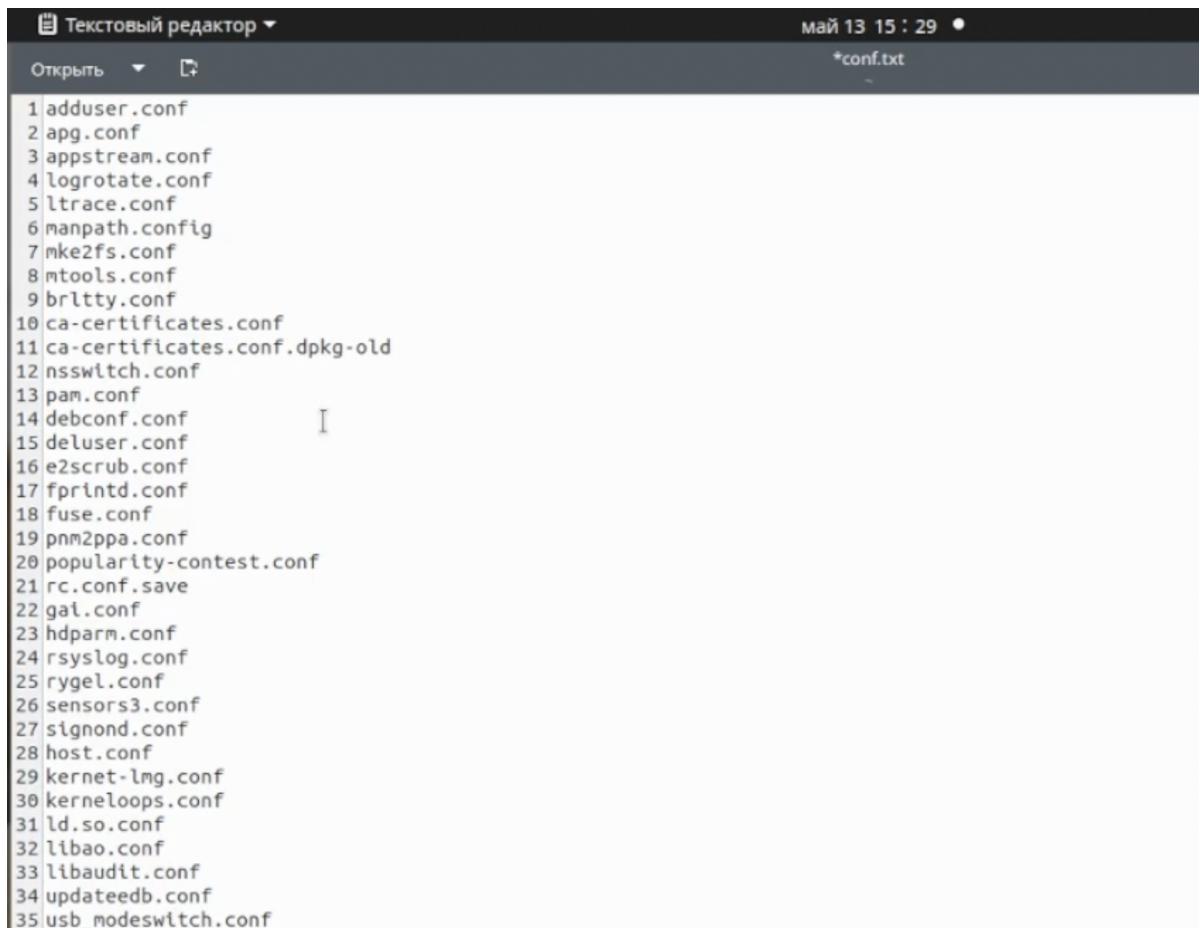
Создала текстовый файл: (рис 3.1)

touch conf.txt

```
valeriya@valeriya-Aspire-V3-572G:~$ touch conf.txt
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 3.1. Создание текстового файла

Добавила имена файлов в новый файл: (рис 3.2)



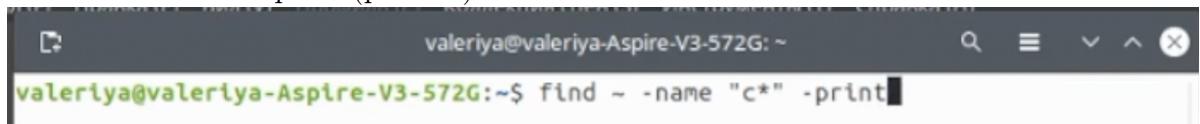
```
1 adduser.conf
2 apg.conf
3 appstream.conf
4 logrotate.conf
5 ltrace.conf
6 manpath.config
7 mke2fs.conf
8 mtools.conf
9 brltty.conf
10 ca-certificates.conf
11 ca-certificates.conf.dpkg-old
12 nsswitch.conf
13 pam.conf
14 debconf.conf
15 deluser.conf
16 e2scrub.conf
17 fprintd.conf
18 fuse.conf
19 pnm2ppa.conf
20 popularity-contest.conf
21 rc.conf.save
22 gai.conf
23 hdparm.conf
24 rsyslog.conf
25 rygel.conf
26 sensors3.conf
27 signond.conf
28 host.conf
29 kernel-lmg.conf
30 kerneloops.conf
31 ld.so.conf
32 libao.conf
33 libaudit.conf
34 updateedb.conf
35 usb_modeswitch.conf
```

Рис 3.2. Добавление в файл

4. Определила, какие файлы в моем домашнем каталоге имеют имена, начинающиеся с символа с:

Первый вариант:

find ~ -name "c*" -print (рис 4.1)



```
valeriya@valeriya-Aspire-V3-572G:~$ find ~ -name "c*" -print
```

```
valeriya@valeriya-Aspire-V3-572G: ~
/home/valeriya/snap/snap-store/common/.cache/gnome-software/icons/c8aef6476c95e2f699a4754681bb630c7154fd70-icon_smKeoSb.png
/home/valeriya/snap/snap-store/common/.cache/gnome-software/icons/cce3c494341bf36f5439cbdd344b24b196b58cd1-256.png
/home/valeriya/snap/snap-store/common/.cache/gnome-software/cssresource
/home/valeriya/snap/snap-store/common/.cache/gnome-software/cssresource/c8aef6476c95e2f699a4754681bb630c7154fd70-icon_smKeoSb.png
/home/valeriya/snap/snap-store/481/.local/share/icons/Adwaita/cursor.theme
/home/valeriya/snap/snap-store/481/.local/share/icons/ubuntu-mono-dark/categories
/home/valeriya/snap/snap-store/481/.local/share/icons/Humanity/categories
/home/valeriya/snap/snap-store/481/.local/share/icons/Humanity/categories@2
/home/valeriya/snap/snap-store/481/.local/share/icons/ubuntu-mono-light/categories
/home/valeriya/snap/snap-store/current
/home/valeriya/snap/snap-store/518/.local/share/icons/Adwaita/cursor.theme
/home/valeriya/snap/snap-store/518/.local/share/icons/ubuntu-mono-dark/categories
/home/valeriya/snap/snap-store/518/.local/share/icons/Humanity/categories
/home/valeriya/snap/snap-store/518/.local/share/icons/Humanity/categories@2
/home/valeriya/snap/snap-store/518/.local/share/icons/ubuntu-mono-light/categories
/home/valeriya/snap/zoom-client/common
```

Рис 4.1. Нахождение файлов начинающихся с с

Второй вариант:

ls -l | grep c* (рис 4.2)

```
valeriya@valeriya-Aspire-V3-572G: ~
valeriya@valeriya-Aspire-V3-572G: ~$ ls -l | grep c*
-rw-rw-r-- 1 valeriya valeriya 497 май 13 15:29 conf.txt
valeriya@valeriya-Aspire-V3-572G: ~$ cd /home
valeriya@valeriya-Aspire-V3-572G:/home$ ls -l | grep c*
итого 12
-rwxr--r-- 1 root      root      0 май 11 19:26 australia
-rw-r--r-- 1 root      root      0 май 11 19:12 equiplist2
---xr-xr-- 1 root      root      0 май 11 19:30 my_os
drwxr-xr-x  3 root      root    4096 май 11 19:45 play
drwxr-xr-x  4 root      root    4096 май 11 19:20 ski.plases
drwxr-xr-x 28 valeriya valeriya 4096 май 13 15:33 valeriya
valeriya@valeriya-Aspire-V3-572G:/home$
```

Рис 4.2. Нахождение файлов начинающихся с с

5. Вывела на экран (постранично) имена файлов из каталога /etc, начинающихся с символа h: (рис 5.1)

Перешла в каталог /etc:

```

cd /etc
ls -l h* |less
valeriya@valeriya-Aspire-V3-572G:/etc$ ls -l h* |less

-rw-r--r-- 1 root root 6748 янв 21 2018 hddtemp.db
-rw-r--r-- 1 root root 5060 авг 21 2019 hdparm.conf
-rw-r--r-- 1 root root 92 дек 5 2019 host.conf
-rw-r--r-- 1 root root 4 июл 31 2020 hostid
-rw-r--r-- 1 root root 24 сен 13 2020 hostname
-rw-r--r-- 1 root root 238 сен 13 2020 hosts
-rw-r--r-- 1 root root 411 июл 31 2020 hosts.allow
-rw-r--r-- 1 root root 711 июл 31 2020 hosts.deny

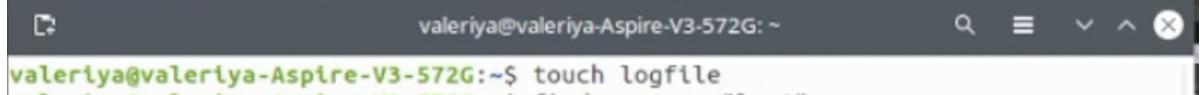
hp:
ИТОГО 4
-rw-r--r-- 1 root root 973 июл 31 2020 hplip.conf
(END)

```

Рис 5.1. Вывод на экран имен файлов из каталога /etc, начиняющихся с символа h

6. Создала файл logfile и запустила в фоновом режиме процесс, который записал в новый файл имена, которые начинаются с log:

touch logfile (рис 6.1)



```

valeriya@valeriya-Aspire-V3-572G:~$ touch logfile

```

Рис 6.1. Создание файла

Запустила в фоновом режиме процесс, который добавил в файл logfile имена которые начинаются с log: (рис 6.2)

```

find ~ -name "log*" -print > logfile &
valeriya@valeriya-Aspire-V3-572G:~$ find ~ -name "log*" -print > logfile &
[1] 9933
valeriya@valeriya-Aspire-V3-572G:~$ 

```

```
 logfile
1 /home/valeriya/laboratory/.git/logs
2 /home/valeriya/snap/zoom-client/143/.zoom/logs
3 /home/valeriya/.local/share/keyrings/login.keyring
4 /home/valeriya/.git/logs
5 /home/valeriya/.mozilla/firefox/1p9z448z.default-release/logins.json
6 /home/valeriya/.mozilla/firefox/1p9z448z.default-release/logins-backup.json
7 /home/valeriya/logfile
8 /home/valeriya/.zoom/logs
9 /home/valeriya/.config/obs-studio/logs
10 /home/valeriya/work/laboratory/.git/logs
```

Рис 6.2. Запуск в фоновом режиме процесса

7. Удалила файл logfile: (рис 7.1)

```
rm -r logfile
/[1]+ Завершён          find ~ -name "log*" -print > logfile
/[1]+ [1] valeriya@valeriya-Aspire-V3-572G:~$ ls
/[1]+ [1] '2021-05-13 14-58-12.mkv'   '2021-05-13 15-41-17.mkv'   snap
/[1]+ [1] '2021-05-13 15-14-00.mkv'   '2021-05-13 15-45-45.mkv'   work
/[1]+ [1] '2021-05-13 15-14-41.mkv'   conf.txt
/[1]+ [1] '2021-05-13 15-15-57.mkv'   file.txt
/[1]+ [1] '2021-05-13 15-17-53.mkv'   home
/[1]+ [1] '2021-05-13 15-25-38.mkv'   lab
/[1]+ [1] '2021-05-13 15-28-53.mkv'   lab07.asm-
/[1]+ [1] '2021-05-13 15-30-25.mkv'   laboratory
/[1]+ [1] '2021-05-13 15-33-29.mkv'   monthly
/[1]+ [1] valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 7.1. Удаление файла

8. Запустила в фоновом режиме редактор gedit: (рис 8.1 и 8.2)

gedit &

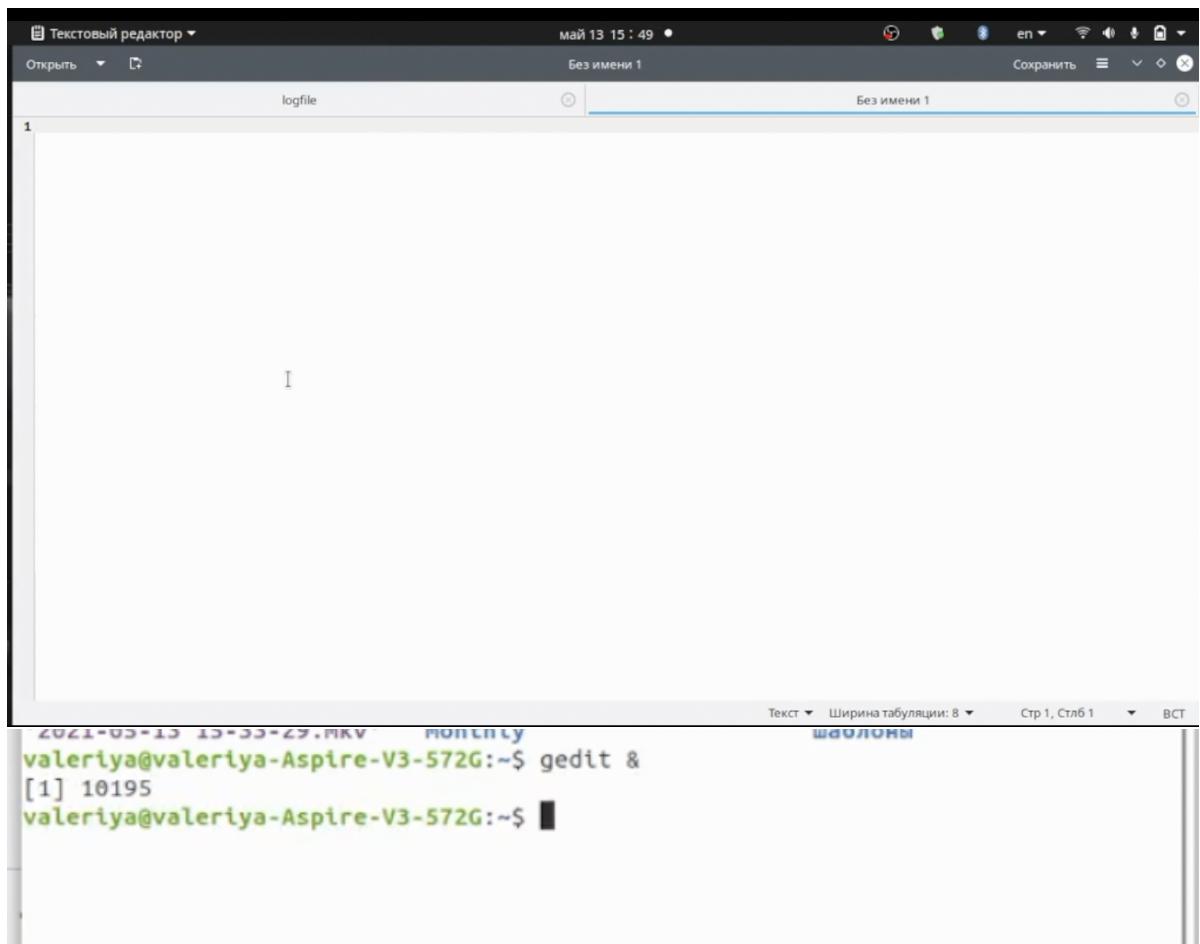


Рис 8.1 и 8.2. Запуск редактора в фоновом режиме

9. Определила идентификатор процесса gedit, используя команду ps и фильтр grep:

Для получения информации о процессах, управляемых вами и запущенных (работающих или остановленных) на вашем терминале, используйте опцию aux.

ps aux (рис 9.1 и 9.2)

```
valeriya@valeriya-Aspire-V3-572G:~$ ps aux
```

```

root      5990  0.0  0.0      0      0 ?          I    15:15   0:01 [kworker/2:3-
root      6869  0.2  0.0      0      0 ?          I    15:23   0:03 [kworker/1:0-
root      6871  0.0  0.0      0      0 ?          I    15:23   0:00 [kworker/u16:
root      7246  0.1  0.0      0      0 ?          I    15:29   0:01 [kworker/3:1-
root      7307  0.2  0.0      0      0 ?          I    15:29   0:02 [kworker/2:0-
root      7572  0.0  0.0      0      0 ?          I    15:31   0:00 [kworker/u16:
root      8441  0.4  0.0      0      0 ?          I    15:35   0:03 [kworker/1:2-
root      8462  0.0  0.0      0      0 ?          I    15:36   0:00 [kworker/u16:
valeriya 8625  0.0  0.2  630392 19696 ?        SNl   15:41   0:00 /usr/lib/x86_
root      8638  0.0  0.0      0      0 ?          I    15:41   0:00 [kworker/1:1-
root      8659  0.0  0.0      0      0 ?          I    15:41   0:00 [kworker/0:0-
valeriya 8678  0.5  0.9  840904 73816 ?        Ssl   15:41   0:02 /usr/libexec/
valeriya 8699  0.0  0.0  14340  6008 pts/0       Ss    15:41   0:00 bash
valeriya 9451  0.0  0.0  2616   536 ?          S     15:45   0:00 sh -c "/usr/b
valeriya 9453  0.4  0.3  168432 31104 ?        SL    15:45   0:01 /usr/bin/obs-
valeriya 9454  0.8  0.3  437844 27772 ?        Ssl   15:45   0:02 /usr/libexec/
valeriya 9471  1.9  1.1  2182644 93184 ?       SNsl  15:45   0:04 /usr/libexec/
root      9492  0.2  0.0      0      0 ?          I    15:45   0:00 [kworker/3:0-
root      9494  0.0  0.0      0      0 ?          I    15:45   0:00 [kworker/1:3-
root      10085  0.2  0.0      0      0 ?          I    15:48   0:00 [kworker/2:1-
root      10262  0.2  0.0      0      0 ?          I    15:49   0:00 [kworker/0:1-
valeriya 10392 0.0  0.0  14064  3440 pts/0       R+   15:49   0:00 ps aux
[1]+  Завершён          gedit
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 9.1 и 9.2. Получение информации о процессах, управляемых вами и запущенных на вашем терминале

Команда ps используется для получения информации о процессах.

ps (рис 9.2)

```

valeriya@valeriya-Aspire-V3-572G:~$ ps
 PID TTY      TIME CMD
 8699 pts/0    00:00:00 bash
 10551 pts/0    00:00:00 ps
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 9.2. Получение информации о процессах

Фильтр grep:

ps aux | grep gedit(рис 9.3)

```

valeriya@valeriya-Aspire-V3-572G:~$ ps aux | grep gedit
valeriya 10948  0.0  0.0  11608   736 pts/0    S+   15:53   0:00 grep --color=
auto gedit
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 9.3. Фильтр grep

10. С помощью команды man определила опции команды kill:

man kill (рис 10.1)

```
valeriya@valeriya-Aspire-V3-572G:~$ man kill
```

```
KILL(1) User Commands KILL(1)
```

```
NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]
```

```
DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available
    signals. Particularly useful signals include HUP, INT, KILL, STOP,
    CONT, and 0. Alternate signals may be specified in three ways: -9,
    -SIGKILL or -KILL. Negative PID values may be used to choose whole
    process groups; see the PGID column in ps command output. A PID of -1
    is special; it indicates all processes except the kill process itself
    and init.
```

```
OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
```

```
Manual page kill(1) line 1 (press h for help or q to quit)
```

Рис 10.1. Команда man kill

Завершила процесс gedit с помощью команды kill:

kill %gedit (рис 10.2)

```
valeriya@valeriya-Aspire-V3-572G:~$ gedit &
[1] 11559
valeriya@valeriya-Aspire-V3-572G:~$
(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:71:
44: Invalid number for color value

(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:72:
44: Invalid number for color value

(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:74:
53: Invalid number for color value

(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:75:
53: Invalid number for color value

(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:76:
56: Invalid number for color value

(gedit:11559): Gtk-WARNING **: 15:55:53.149: Theme parsing error: colors.css:77:
65: Invalid number for color value

valeriya@valeriya-Aspire-V3-572G:~$ kill %gedit
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 10.2. Завершение процесса с помощью команды kill

11. С помощью команды man определила опции команды df:

man df (рис 11.1 и 11.2)

```
valeriya@valeriya-Aspire-V3-572G:~$ man df
```

```
DF(1) User Commands DF(1)

NAME
    df - report file system disk space usage

SYNOPSIS
    df [OPTION]... [FILE]...

DESCRIPTION
    This manual page documents the GNU version of df. df displays the amount of disk space available on the file system containing each file name argument. If no file name is given, the space available on all currently mounted file systems is shown. Disk space is shown in 1K blocks by default, unless the environment variable POSIXLY_CORRECT is set, in which case 512-byte blocks are used.

    If an argument is the absolute file name of a disk device node containing a mounted file system, df shows the space available on that file system rather than on the file system containing the device node. This version of df cannot show the space available on unmounted file systems, because on most kinds of systems doing so requires very non-portable intimate knowledge of file system structures.

Manual page df(1) line 1 (press h for help or q to quit)
```

Рис 11.1 и 11.2. Команда man df

12. Используя команду man для просмотра описания команды du:

man du; (рис 12.1 и рис 12.2)

```
valeriya@valeriya-Aspire-V3-572G:~$ kill %gedit
valeriya@valeriya-Aspire-V3-572G:~$ man df
[1]+  Завершено      gedit
valeriya@valeriya-Aspire-V3-572G:~$ man du
```

```
valeriya@valeriya-Aspire-V3-572G:~
```

DU(1) User Commands DU(1)

NAME

du - estimate file space usage

SYNOPSIS

du [OPTION]... [FILE]...
du [OPTION]... --files0-from=F

DESCRIPTION

Summarize disk usage of the set of FILEs, recursively for directories.

Mandatory arguments to long options are mandatory for short options too.

-0, --null
end each output line with NUL, not newline

-a, --all
write counts for all files, not just directories

--apparent-size
print apparent sizes, rather than disk usage; although the ap-

Manual page du(1) line 1 (press h for help or q to quit)

Рис 12.1 и 12.2. Команда man du

13. Выполнила команды df и du:

Команда df показывает размер каждого смонтированного раздела диска.

df -vi (рис 13.1 и 13.2)

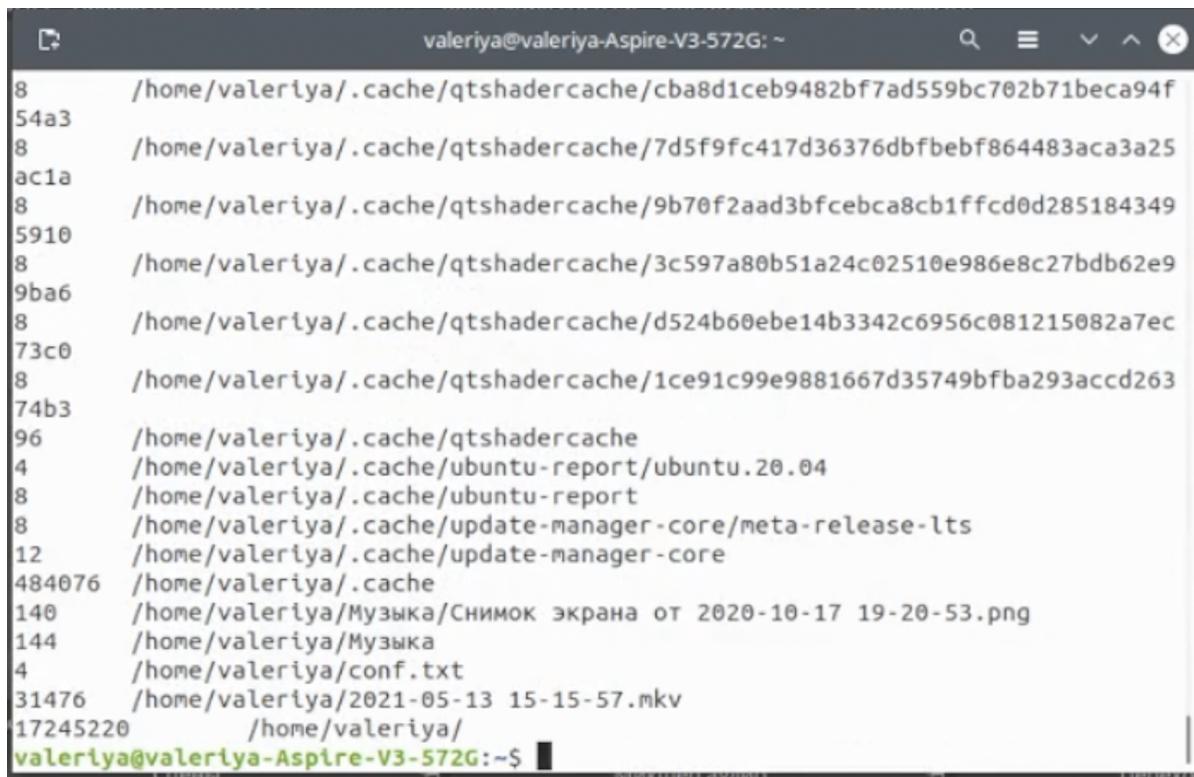
```
valeriya@valeriya-Aspire-V3-572G:~$ df -vi
```

		valeriya@valeriya-Aspire-V3-572G: ~			
/dev/loop7 78	29200	29200	0	100%	/snap/geany-gtk/3
/dev/loop6 68	29200	29200	0	100%	/snap/geany-gtk/3
/dev/loop8 1804/60	18513	18513	0	100%	/snap/gnome-3-34-
/dev/loop9 1804/66	18508	18508	0	100%	/snap/gnome-3-34-
/dev/loop11 themes/1515	64986	64986	0	100%	/snap/gtk-common-
/dev/loop12	5	5	0	100%	/snap/null/3
/dev/loop10 themes/1514	63978	63978	0	100%	/snap/gtk-common-
/dev/loop13 481	15843	15843	0	100%	/snap/snap-store/
/dev/loop15	474	474	0	100%	/snap/snapd/11588
/dev/loop14 518	15847	15847	0	100%	/snap/snap-store/
/dev/loop17 /143	13539	13539	0	100%	/snap/zoom-client
/dev/sda1	0	0	0	-	/boot/efi
tmpfs	1009086	97	1008989	1%	/run/user/1000
/dev/loop18	474	474	0	100%	/snap/snapd/11841
valeriya@valeriya-Aspire-V3-572G:~\$					

Рис 13.1 и 13.2. Размер каждого смонтированного раздела диска

Команда du показывает число килобайт, используемое каждым файлом или каталогом.

```
du -a ~/
valeriya@valeriya-Aspire-V3-572G:~$ du -a ~/
```



valeriya@valeriya-Aspire-V3-572G: ~

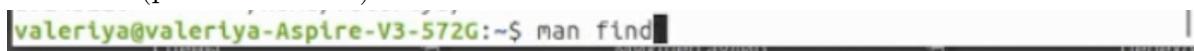
File or Directory	Size (kB)
/home/valeriya/.cache/qtshadercache/cba8d1ceb9482bf7ad559bc702b71beca94f	8
/home/valeriya/.cache/qtshadercache/7d5f9fc417d36376dbfbef864483aca3a25	8
/home/valeriya/.cache/qtshadercache/9b70f2aad3bfcebc8cb1ffcd0d285184349	8
/home/valeriya/.cache/qtshadercache/3c597a80b51a24c02510e986e8c27bdb62e9	5910
/home/valeriya/.cache/qtshadercache/d524b60ebe14b3342c6956c081215082a7ec	9bab
/home/valeriya/.cache/qtshadercache/1ce91c99e9881667d35749bfa293accd263	73c0
/home/valeriya/.cache/qtshadercache	74b3
/home/valeriya/.cache/ubuntu-report/ubuntu.20.04	96
/home/valeriya/.cache/ubuntu-report	4
/home/valeriya/.cache/update-manager-core/meta-release-lts	8
/home/valeriya/.cache/update-manager-core	12
/home/valeriya/.cache	484076
/home/valeriya/Музыка/Снимок экрана от 2020-10-17 19-20-53.png	140
/home/valeriya/Музыка	144
/home/valeriya/conf.txt	4
/home/valeriya/2021-05-13 15-15-57.mkv	31476
/home/valeriya/	17245220

valeriya@valeriya-Aspire-V3-572G:~\$

Рис 13.3 и 13.4. Число килобайт, используемое домашним каталогом

14. С помощью команды man определила опции команды find:

man find (рис 14.1 и 14.2)



valeriya@valeriya-Aspire-V3-572G:~\$ man find

```
FIND(1)          General Commands Manual          FIND(1)

NAME
    find - search for files in a directory hierarchy

SYNOPSIS
    find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [ex-
        pression]

DESCRIPTION
    This manual page documents the GNU version of find. GNU find searches
    the directory tree rooted at each given starting-point by evaluating
    the given expression from left to right, according to the rules of
    precedence (see section OPERATORS), until the outcome is known (the
    left hand side is false for and operations, true for or), at which
    point find moves on to the next file name. If no starting-point is
    specified, `.' is assumed.

    If you are using find in an environment where security is important
    (for example if you are using it to search directories that are
    writable by other users), you should read the 'Security Considerations'
    chapter of the findutils documentation, which is called Finding Files
    and comes with findutils. That document also includes a lot more de-
Manual page find(1) line 1 (press h for help or q to quit)
```

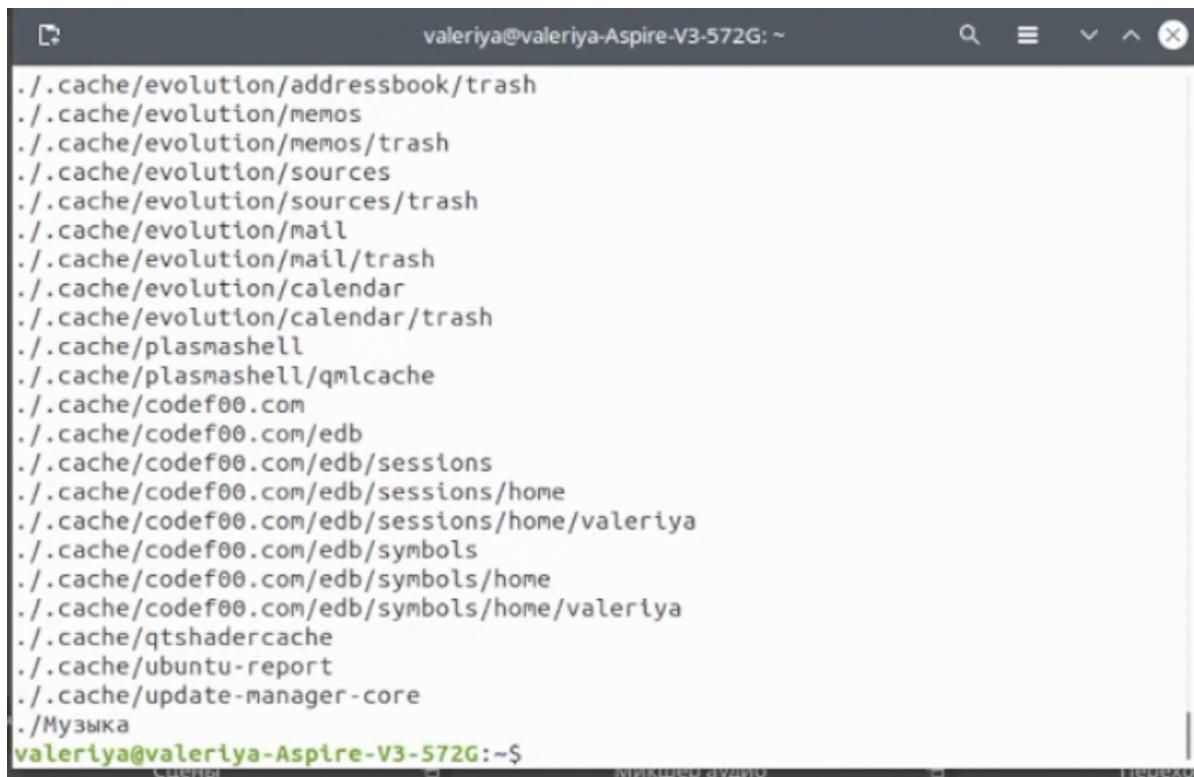
Рис 14.1 и 14.2. Команда man find

15. С помощью команды man find определила операцию, которая выводит имена всех директорий, имеющихся в домашнем каталоге:

-type - тип искомого, d - каталог;

find -type d -print (рис 15.1 и 15.2)

```
valeriya@valeriya-Aspire-V3-572G:~$ find -type d -print
```



```
valeriya@valeriya-Aspire-V3-572G: ~
./.cache/evolution/addressbook/trash
./.cache/evolution/memos
./.cache/evolution/memos/trash
./.cache/evolution/sources
./.cache/evolution/sources/trash
./.cache/evolution/mail
./.cache/evolution/mail/trash
./.cache/evolution/calendar
./.cache/evolution/calendar/trash
./.cache/plasmashell
./.cache/plasmashell/qmlcache
./.cache/codef00.com
./.cache/codef00.com/edb
./.cache/codef00.com/edb/sessions
./.cache/codef00.com/edb/sessions/home
./.cache/codef00.com/edb/sessions/home/valeriya
./.cache/codef00.com/edb/symbols
./.cache/codef00.com/edb/symbols/home
./.cache/codef00.com/edb/symbols/home/valeriya
./.cache/qtshadercache
./.cache/ubuntu-report
./.cache/update-manager-core
./Музыка
valeriya@valeriya-Aspire-V3-572G:~$
```

Рис 15.1 и 15.2. Вывод имен всех директорий, имеющихся в домашнем каталоге

5 Вывод:

В данной лабораторной работе, 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.

6 Контрольные вопросы (лабораторная работа №7)

1. Какие потоки ввода вывода вы знаете?

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.

2. Объясните разницу между операцией `>` и `>>`.

#Создаёт файл, содержащий список дерева каталогов.

`ls -lR > dir-tree.list`

`1>filename`

#Перенаправление вывода (stdout) в файл “filename”.

`1>>filename`

#Перенаправление вывода (stdout) в файл “filename”,

#файл открывается в режиме добавления.

`2>filename`

#Перенаправление stderr в файл “filename”.

`2>>filename`

#Перенаправление stderr в файл “filename”,

```
#файл открывается в режиме добавления.  
&>filename  
#Перенаправление stdout и stderr в файл “filename”.
```

3. Что такое конвейер?

Конвейер в терминологии операционных систем семейства Unix — некоторое множество процессов, для которых выполнено следующее перенаправление ввода-вывода: то, что выводит на поток стандартного вывода предыдущий процесс, попадает в поток стандартного ввода следующего процесса.

Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

4. Что такое процесс? Чем это понятие отличается от программы?

Термин процесс впервые появился при разработке операционной системы Multix и имеет несколько определений, которые используются в зависимости от контекста. Процесс - это: программа на стадии выполнения “объект”, которому выделено процессорное время

Любой команде, выполняемой в системе, присваивается идентификатор процесса (process ID). Получить информацию о процессе и управлять им, пользуясь идентификатором процесса, можно из любого окна командного интерпретатора.

Компьютерная программа сама по себе — лишь пассивная последовательность инструкций. В то время как процесс — непосредственное выполнение этих инструкций. Также, процессом называют выполняющуюся программу и все её элементы: адресное пространство, глобальные переменные, регистры, стек, открытые файлы и так далее.

5. Что такое PID и GID?

Идентификатор процесса (англ. Process IDentifier, PID) — уникальный номер (идентификатор) процесса в многозадачной операционной системе (ОС). Например, в ОС Linux PID хранится в переменной целочисленного типа (int).

Идентификатор группы (GID)

Группы пользователей применяются для организации доступа нескольких пользо-

вателей к некоторым ресурсам. У группы, так же, как и у пользователя, есть имя и идентификационный номер — GID (Group ID).

6. Что такое задачи и какая команда позволяет ими управлять?

Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать знак амперсанда &.

Например:

```
gedit &
```

Будет запущен текстовой редактор gedit в фоновом режиме. Консоль при этом не будет заблокирована.

Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач.

Для завершения задачи необходимо выполнить команду

```
kill %номер задачи
```

7. Найдите информацию об утилитах top и htop. Каковы их функции?

top (table of processes) — консольная команда, которая выводит список работающих в системе процессов и информацию о них. По умолчанию она в реальном времени сортирует их по нагрузке на процессор. Программа написана для UNIX-совместимых операционных систем и опубликована под свободной лицензией GNU FDL.

htop — продвинутый монитор процессов, написанный для Linux. Он был задуман заменить стандартную программу top. В отличие от top, htop показывает все процессы в системе. Также показывает время непрерывной работы, использование процессоров и памяти.

8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.

Команда find используется для поиска и отображения имён файлов, соответствующих заданной строке символов.

Формат команды:

`find` путь [-опции]

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Примеры:

1. Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на `f`:

```
find ~ -name "f*" -print,
```

где `~` — обозначение вашего домашнего каталога, `-name` — после этой опции указывается имя файла, который нужно найти, `"f*"` — строка символов, определяющая имя файла, `-print` — опция, задающая вывод результатов поиска на экран.

2. Вывести на экран имена файлов в каталоге `/etc`, начинающихся с символа `p`:

```
find /etc -name "p*" -print
```

9. Можно ли по контексту (содержанию) найти файл? Если да, то как?

Фильтрация текста

Найти в текстовом файле указанную строку символов позволяет команда `grep`.

Формат команды:

```
grep строка имя_файла
```

Кроме того, команда `grep` способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом `grep`.

Примеры:

1. Показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на `f`, в которых есть слово `begin`:

```
grep begin f*
```

2. Найти в текущем каталоге все файлы, в имени которых есть буквосочетание «лаб»:

`ls -l | grep лаб`

10. Как определить объем свободной памяти на жёстком диске?

Команда `df` показывает размер каждого смонтированного раздела диска.

Формат команды:

`df [-опции] [файловая_система]`

Пример:

`df -vi`

11. Как определить объем вашего домашнего каталога?

Команда `du` показывает число килобайт, используемое каждым файлом или каталогом.

Формат команды:

`du [-опции] [имя_файла...]`

Пример.

`du -a ~ /`

На afs можно посмотреть использованное пространство командой

`fs quota`

12. Как удалить зависший процесс?

Находим PID зависшего процесса;

«Убиваем» процесс командой `kill`:

Когда известен PID процесса, мы можем убить его командой `kill`. Команда `kill` принимает в качестве параметра PID процесса.

Убиваем процессы командой `killall`:

Команда `killall` в Linux предназначена для «убийства» всех процессов, имеющих одно и то же имя. Это удобно, так как нам не нужно знать PID процесса.