

Лабораторная работа №6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Зарифбеков Амир Пайшанбиевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	14
5	Контрольные вопросы	15

Список иллюстраций

3.1	работа с файлом file.txt	7
3.2	Выведем имена всех файлов из file.txt	7
3.3	запишем их в новый текстовый файл conf.txt.	8
3.4	определяем имена всех файлов с символом с.	8
3.5	операции с файлом feathers	9
3.6	запустим процесс записи файла ~/logfile	9
3.7	удаляем файл	10
3.8	запустим gedit	10
3.9	Определим идентификатор процесса gedit	10
3.10	Читаем справку (man) команды kill	10
3.11	Выполню команды df и du	11
3.12	man df	11
3.13	man du	12
3.14	команда find	13

Список таблиц

1 Цель работы

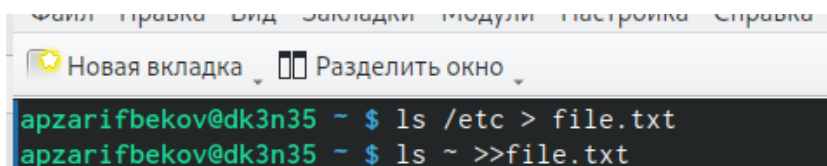
Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем

2 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, начинающиеся с символа `s`? Предложите несколько вариантов, как это сделать.
5. Выведите на экран (по странично) 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

3 Выполнение лабораторной работы

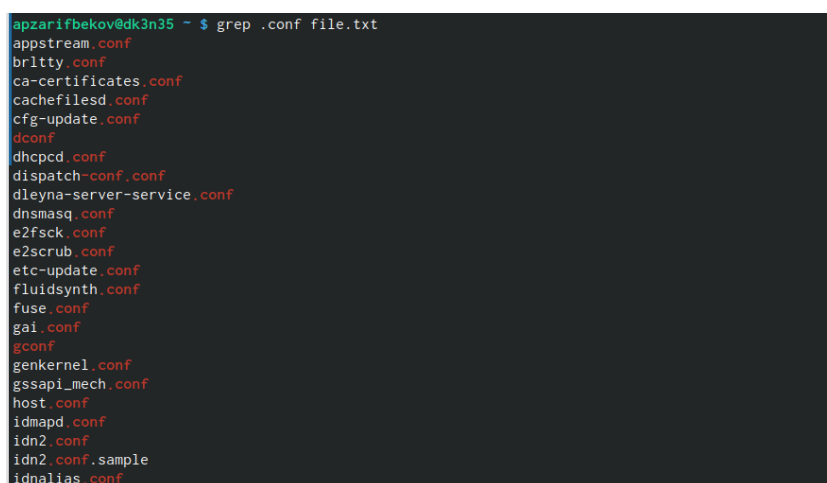
1. Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишем в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.



```
apzarifbekov@dk3n35 ~ $ ls /etc > file.txt
apzarifbekov@dk3n35 ~ $ ls ~ >> file.txt
```

Рис. 3.1: работа с файлом file.txt .

2. Выведем имена всех файлов из file.txt, имеющих расширение .conf, после чего запишем их в новый текстовый файл conf.txt.



```
apzarifbekov@dk3n35 ~ $ grep .conf file.txt
appstream.conf
brltty.conf
ca-certificates.conf
cachefilesd.conf
cfg-update.conf
dconf
dhcpcd.conf
dispatch.conf.conf
dleyna-server-service.conf
dnsmasq.conf
e2fsck.conf
e2scrub.conf
etc-update.conf
fluidsynth.conf
fuse.conf
gai.conf
gconf
genkernel.conf
gssapi_mech.conf
host.conf
idmapd.conf
idn2.conf
idn2.conf.sample
idnalias.conf
```

Рис. 3.2: Выведем имена всех файлов из file.txt

```
whois.conf
xattr.conf
xinetd.conf
xtables.conf
apzarifbekov@dk3n35 ~ $ grep .conf file.txt > conf.txt
apzarifbekov@dk3n35 ~ $
```

Рис. 3.3: запишем их в новый текстовый файл conf.txt.

3. Определим, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с.

```
apzarifbekov@dk3n35 ~ $ ls -l | grep c*
apzarifbekov@dk3n35 ~ $ ls -l | grep c*
apzarifbekov@dk3n35 ~ $ find ~/c* -name "c*" -print
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/conf.txt
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/cpp.5
apzarifbekov@dk3n35 ~ $
```

Рис. 3.4: определяем имена всех файлов с символом с.

4. Выведем на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.


```

Итого 108
-rw----- 1 apzarifbekov studsci 3389 фев 16 17:39
-rw-rw-r-- 1 apzarifbekov studsci 0 мар 9 15:07 abc1
-rw-r--r-- 1 apzarifbekov studsci 0 мар 9 14:46 apri
drwxr--r-- 2 apzarifbekov studsci 2048 мар 9 16:11 australia
drwxr-xr-x 2 apzarifbekov studsci 2048 фев 22 18:33 bin
-rw-r--r-- 1 apzarifbekov studsci 1191 мар 15 14:49 conf.txt
-rw-r--r-- 1 apzarifbekov studsci 1 ноя 30 10:45 cpp.5
-rw-r--r-- 1 apzarifbekov studsci 0 мар 9 16:21 file.old
-rw-r--r-- 1 apzarifbekov studsci 4116 мар 15 14:40 file.txt
drwxr-xr-x 3 apzarifbekov studsci 2048 сен 21 10:48 GNUstep
-rw-r--r-- 1 apzarifbekov studsci 173 дек 23 16:23 lab06-1.asm
-rwxr-xr-x 1 apzarifbekov studsci 15968 ноя 2 11:16 main2
-rw-r--r-- 1 apzarifbekov studsci 132 ноя 2 11:16 main2.cpp
-rwxr-xr-x 1 apzarifbekov studsci 15968 ноя 2 11:39 main3
-rw-r--r-- 1 apzarifbekov studsci 188 ноя 2 11:39 main3.cpp
-rwxr-xr-x 1 apzarifbekov studsci 16184 ноя 2 11:44 main4
-rw-r--r-- 1 apzarifbekov studsci 201 ноя 2 11:44 main4.cpp
-rw-r--r-- 1 apzarifbekov studsci 32 ноя 30 10:48 main5.cpp
-rw-r--r-- 1 apzarifbekov studsci 0 мар 9 14:58 may
drwxr-xr-x 2 apzarifbekov studsci 2048 мар 9 14:50 monthly
dr-xr--r-x 2 apzarifbekov studsci 2048 мар 9 16:19 my_os
d-w---x--x 3 apzarifbekov studsci 2048 мар 9 16:24 play
-rw-r--r-- 1 apzarifbekov studsci 745 фев 16 17:39 .pub
drwxr-xr-x 3 apzarifbekov root 2048 мар 2 13:34 public
lrwxr-xr-x 1 apzarifbekov root 18 мар 2 21:41 public_html -> public/public_html
drwxr-xr-x 3 apzarifbekov studsci 2048 мар 9 14:56 reports
drwxr-xr-x 4 apzarifbekov studsci 2048 мар 9 16:07 ski.places
drwxr-xr-x 2 apzarifbekov studsci 2048 окт 28 15:55 tmp
drwxr-xr-x 5 apzarifbekov studsci 2048 фев 22 18:41 work
drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Видео
drwxr-xr-x 2 apzarifbekov studsci 2048 фев 4 19:25 Документы
drwxr-xr-x 3 apzarifbekov studsci 4096 мар 2 19:52 Загрузки
-rw-r--r-- 1 apzarifbekov studsci 180 ноя 3 16:54 задал.cpp
drwxr-xr-x 3 apzarifbekov studsci 2048 фев 16 18:12 Изображения
drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Музыка
drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Общедоступные
drwxr-xr-x 2 apzarifbekov studsci 2048 фев 4 19:25 Рабочий стол
drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Шаблоны
~
~
~
lines 1-39/39 (END)

```

Рис. 3.5: операции с файлом feathers

5. Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.

```

drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Общедоступные
drwxr-xr-x 2 apzarifbekov studsci 2048 фев 4 19:25 Рабочий стол
drwxr-xr-x 2 apzarifbekov studsci 2048 сен 14 10:39 Шаблоны
apzarifbekov@dk3n35 ~ $ ls -l | grep h* |less
apzarifbekov@dk3n35 ~ $ ls -l | grep log* |less
apzarifbekov@dk3n35 ~ $ find ~ -name "log*" -print > ~/logfile &
[4] 6052

```

Рис. 3.6: запустим процесс записи файла ~/logfile

6. Удаляем файл ~/logfile.

```
[4] 6032
apzarifbekov@dk3n35 ~ $ rm -r logfile
[4] Завершён      find ~ -name "log*" -print > ~/logfile
apzarifbekov@dk3n35 ~ $
```

Рис. 3.7: удаляем файл

7. Запустим из консоли в фоновом режиме редактор gedit

```
apzarifbekov@dk3n35 ~ $ gedit &
262
apzarifbekov@dk3n35 ~ $
```

Рис. 3.8: запустим gedit

8. Определим идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep.

```
apzarifbekov@dk3n35 ~ $ gedit &
[8] 6543
apzarifbekov@dk3n35 ~ $ ps aux | grep -i gedit
apzarif+  6519  12.7  2.4 712924 145884 pts/0    Sl   15:10   0:02 gedit
apzarif+  6557   0.0  0.0   6904  2188 pts/0    R+   15:10   0:00 grep --colour=auto -i gedit
[8] Завершён      gedit
apzarifbekov@dk3n35 ~ $
```

Рис. 3.9: Определим идентификатор процесса gedit

9. Прочту справку (man) команды kill, после чего используйте её для завершения процесса gedit.

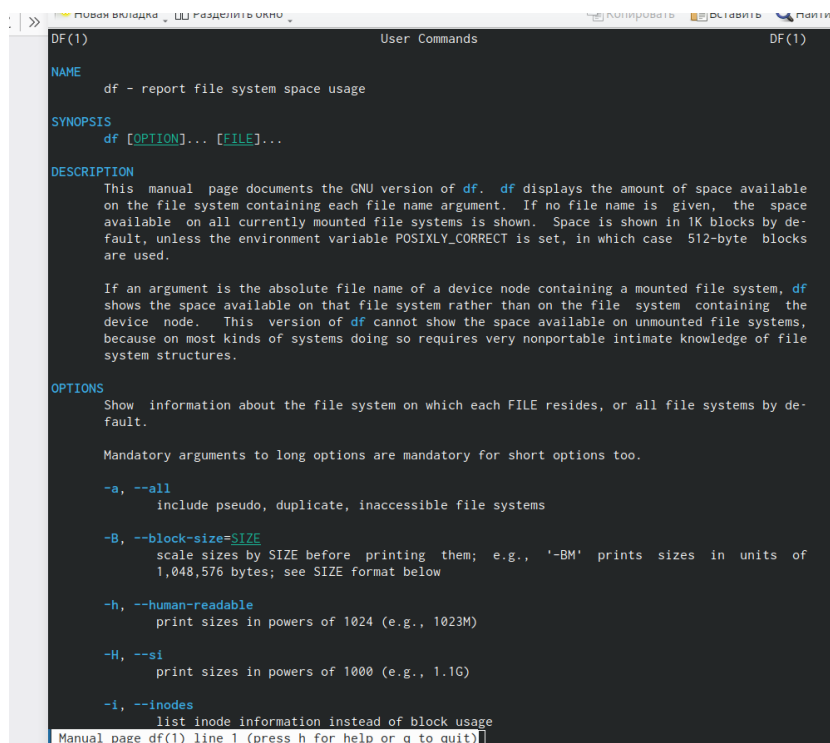
```
apzarif+  6557   0.0  0.0   6904  2188 pts/0    R+   15:10   0:00 grep --colour=auto -i gedit
[8] Завершён      gedit
apzarifbekov@dk3n35 ~ $ man kill
[8]+ Остановлен  man kill
apzarifbekov@dk3n35 ~ $ kill -9 3325
apzarifbekov@dk3n35 ~ $
```

Рис. 3.10: Читаем справку (man) команды kill

10. Выполню команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.

```
[9]+ Остановлен man df
apzarifbekov@dk3n35 ~ $ man du
[10]+ Остановлен man du
apzarifbekov@dk3n35 ~ $
```

Рис. 3.11: Выполню команды df и du



```
DF(1) User Commands DF(1)
NAME
df - report file system space usage

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

DESCRIPTION
This manual page documents the GNU version of df. df displays the amount of 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. 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 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 nonportable intimate knowledge of file system structures.

OPTIONS
Show information about the file system on which each FILE resides, or all file systems by default.

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

-a, --all
    include pseudo, duplicate, inaccessible file systems

-B, --block-size=SIZE
    scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of 1,048,576 bytes; see SIZE format below

-h, --human-readable
    print sizes in powers of 1024 (e.g., 1023M)

-H, --si
    print sizes in powers of 1000 (e.g., 1.1G)

-i, --inodes
    list inode information instead of block usage

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

Рис. 3.12: man df

```
DU(1)                                User Commands                                DU(1)

NAME
    du - estimate file space usage

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

DESCRIPTION
    Summarize device 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 device usage; although the apparent size is usually
        smaller, it may be larger due to holes in ('sparse') files, internal fragmentation, in-
        direct blocks, and the like

    -B, --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of
        1,048,576 bytes; see SIZE format below

    -b, --bytes
        equivalent to '--apparent-size --block-size=1'

    -c, --total
        produce a grand total

    -D, --dereference-args
        dereference only symlinks that are listed on the command line

    -d, --max-depth=N
        print the total for a directory (or file, with --all) only if it is N or fewer levels
        below the command line argument; --max-depth=0 is the same as --summarize

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

Рис. 3.13: man du

11. Воспользовавшись справкой команды `find`, выведите имена всех директо-
рий, имею-щихся в вашем домашнем каталоге

```
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/objects/info
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/objects/e6
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/objects/f9
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/objects/68
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/logs
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/logs/refs
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/logs/refs/heads
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/logs/refs/remotes
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/work/Amirchik-1.github.io/.git/logs/refs/remotes/origin
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var/luatex-cache
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var/luatex-cache/generic
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var/luatex-cache/generic/names
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var/luatex-cache/generic/fonts
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.texlive2022/texmf-var/luatex-cache/generic/fonts/otl
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/tmp
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.gphoto
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/monthly
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.pki
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.pki/nssdb
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.cache
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/.cache/samba
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/reports
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/reports/monthly
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/reports/monthly/monthly
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/australia
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/ski.places
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/ski.places/plans
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/ski.places/equipment
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/play
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/play/games
/afs/.dk.sci.pfu.edu.ru/home/a/p/apzarifbekov/my_os
apzarifbekov@dk3n35 ~ $
```

Рис. 3.14: команда find

4 Выводы

Ознакомился с инструментами поиска файлов и фильтрации текстовых данных. Приобрёл практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

5 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.
2. • перенаправление `stdout` (вывода) в файл. Если файл отсутствовал, то он создаётся, иначе - перезаписывается. • перенаправление `stdout` (вывода) в файл. Если файл отсутствовал, то он создаётся, иначе - добавляется.
3. Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.
4. Процесс это - совокупность программного кода и данных, загруженных в память ЭВМ. 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. Процессом называют выполняющуюся программу и все её элементы: адресное пространство, глобальные переменные, регистры, стек, открытые файлы и так далее.
5. `PID` — уникальный номер (идентификатор) процесса в многозадачной ОС. `GID` – идентификатор группы. 18
6. Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Запущенные фоном программы называют-

- ся 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. Команда top в Linux системах позволяет вывести в виде таблицы перечень запущенных процессов и оценить, какой объем ресурсов они потребляют, т.е., какую нагрузку создают на сервер и дисковую подсистему. Команда htop — продвинутый монитор процессов, показывает динамический список системных процессов, список обычно выравнивается по использованию ЦПУ. В отличие от top, htop показывает все процессы в системе. Также показывает время непрерывной работы, использование процессоров и памяти. htop часто применяется в тех случаях, когда информации даваемой утилитой top недостаточно, например при поиске утечек памяти в процессах.
 8. 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: • вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на f: find ~ -name "f" -print • вывести на экран имена файлов в каталоге /etc, начинающихся с символа p: find /etc -name "p" -print • найти в вашем домашнем каталоге файлы, имена которых заканчиваются символом и удалить их: find ~ -name "*~" -exec rm "{}" ;
 9. Найти файл по контексту (содержанию) позволяет команда grep. Формат команды: grep строка имя_файла Примеры: • показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на f, в которых есть слово begin: grep begin f* 19 • найти в текущем каталоге все файлы, в имени которых есть буквосочетание «лаб»: ls -l | grep лаб
 10. Определить объем свободной памяти на жёстком диске позволяет команда df.
 11. Определить объем домашнего каталога позволяет команда df /home/

12. Удалить зависший процесс можно командой `kill %номер задачи`